

Галицький коледж імені В'ячеслава Чорновола  
відділення комп'ютерних та видавничих технологій  
циклова комісія інформатики та комп'ютерних дисциплін

ДОПУСТИТИ ДО ЗАХИСТУ

Завідувач відділенням  
комп'ютерних та видавничих  
технологій

Чубей О.О. / \_\_\_\_\_ /

підпис

« \_\_\_\_ » \_\_\_\_\_ 2021 р.

## ПОЯСНЮВАЛЬНА ЗАПИСКА

до дипломного проєкту  
освітньо-кваліфікаційного рівня «молодший спеціаліст»  
зі спеціальності 122 «Комп'ютерні науки»

на тему: «Вебконструктор професійного резюме»

Студент групи К-47

Кухарський Р.В.

\_\_\_\_\_  
(підпис)

Керівник проєкту

Чубей О.О.

\_\_\_\_\_  
(підпис)

Консультанти:

з техніко-економічного

обґрунтування

Меленчук Л.І.

\_\_\_\_\_  
(підпис)

нормоконтролер

Кульчинська Н.З.

\_\_\_\_\_  
(підпис)

Тернопіль - 2021

Галицький коледж імені В'ячеслава Чорновола  
відділення комп'ютерних та видавничих технологій  
циклова комісія інформатики та комп'ютерних дисциплін

**ЗАТВЕРДЖУЮ**

Завідувач відділенням  
комп'ютерних та видавничих  
технологій

Чубей О.О. / \_\_\_\_\_ /

підпис

« \_\_\_\_ » \_\_\_\_\_ 2020 р.

**ЗАВДАННЯ**

на дипломне проектування  
на здобуття освітньо-кваліфікаційного рівня «молодший спеціаліст»  
студенту Кухарському Роману Віталійовичу

---

(прізвище, ім'я та по-батькові студента)

1. Тема проєкту \_\_\_\_\_

\_\_\_\_\_

затверджено наказом по коледжу від “ \_\_\_\_ ” \_\_\_\_\_ 202\_ р., № \_\_\_\_

2. Термін здачі студентом завершеного проєкту “ \_\_\_\_ ” \_\_\_\_\_ 202\_ р.

3. Вихідні дані до проєкту \_\_\_\_\_

\_\_\_\_\_

4. Перелік питань, які повинні бути розроблені в проєкті:

а) основна частина \_\_\_\_\_

б) техніко-економічне обґрунтування \_\_\_\_\_

\_\_\_\_\_

5. Перелік графічного матеріалу \_\_\_\_\_

6. Консультанти проєкту: \_\_\_\_\_

Розділ	Консультанти	Підпис, дата	
		Завдання видано	Завдання прийнято
з техніко-економічного обґрунтування	_____ (вчена ступінь, звання П.І.Б. консультанта)		

### КАЛЕНДАРНИЙ ПЛАН дипломного проектування

№ п/п	Найменування етапу	Терміни	
		початку	завершення
1.	<i>Вибір теми, ознайомлення з вимогами до дипломного проектування</i>	16.11.20	30.11.20
2.	<i>Огляд типових рішень та написання відповідного розділу ПЗ</i>	01.12.20	26.01.21
3.	<i>Дослідження технологій реалізації та написання відповідного розділу ПЗ</i>	27.01.21	15.02.21
4.	<i>Розробка функціональних вимог до проекту та робота над структурою програмного продукту. Написання відповідного розділу ПЗ</i>	15.02.21	02.03.21
5.	<i>Встановлення та налаштування середовища реалізації та написання відповідного розділу ПЗ</i>	02.03.21	16.03.21
6.	<i>Проектування програмного засобу (функціоналу, інтерфейсу, бази даних) та написання відповідного розділу ПЗ</i>	16.03.21	16.04.21
7.	<i>Реалізація та налаштування програмного засобу та написання відповідного розділу ПЗ</i>	17.04.21	03.05.21
8.	<i>Допрацювання модулів</i>	03.05.21	17.05.21
9.	<i>Опрацювання економічного розділу дипломного проекту та оформлення спеціального розділу</i>	17.05.21	18.06.21
10.	<i>Тестування та налагодження програмного продукту та написання відповідного розділу ПЗ</i>	18.05.21	04.06.21
11.	<i>Робота над оформленням пояснювальної записки</i>	04.06.21	11.06.21
13.	<i>Попередній захист дипломного проекту, доопрацювання</i>	11.06.21	
14.	<i>Підготовка до захисту дипломного проекту</i>	18.06.21	22.06.21
15.	<i>Захист дипломного проекту</i>	23.06.21	23.06.21

7. Дата видачі “\_\_\_” \_\_\_\_\_ 2020р. Керівник \_\_\_\_\_/

Завдання прийняв до виконання \_\_\_\_\_/

## Реферат

Вебконструктор професійного резюме. Дипломний проєкт. Кухарський Роман. Галицький коледж імені В'ячеслава Чорновола, відділення комп'ютерних та видавничих технологій. Спеціальність 122 «Комп'ютерні науки». ГК, 2021. Сторінок – 60, рисунків – 27 додатків – 1.

Предметом дослідження є створення та збереження вебрезюме, а також його практичне застосування.

Метою проєкту є реалізація конструктора професійного резюме, який дозволяє скласти резюме онлайн

Система повинна бути реалізована у вигляді веб-сервісу, який містить відповідні вкладки, що дозволяють оперативно і легко подати HR-менеджеру усю необхідну інформацію. В результаті резюме виглядатиме професійно і естетично.

Середовищем для реалізації дипломного проєкту був обраний Visual Studio Code. Для розробки структури проєкту було використано мову програмування JavaScript. Також використано бібліотеку для JavaScript – React. Структуру сайту, інтерфейс та зовнішній вигляд реалізовано з допомогою мови розмітки HTML та мови опису зовнішнього вигляду CSS.

Результат – спроектований вебконструктор онлайн, який надає можливість швидко та якісно створити резюме користувача та завантажити його у PDF форматі.

ВЕБКОНСТРУКТОР, РЕЗЮМЕ, КОРИСТУВАЧ, PDF-ФОРМАТ, REACT, CSS, HTML, JAVASCRIPT, МОБА ПРОГРАМУВАННЯ.

## Abstract

Web designer of a professional resume. Diploma project. Kuharskiy Roman. Galytsky College named after Viacheslav Chornovil, Department of Computer and Publishing Technologies. Specialty 122 "Computer Science". GC, 2021. Pages – 60, figures – 27, appendixes – 1.

The subject of the study is the creation and preservation of a web resume, as well as its practical application.

The aim of the project is to implement a professional resume designer, which allows you to write a resume online

The system should be implemented in the form of a web service that contains the appropriate tabs that allow you to quickly and easily provide the HR-manager with all the necessary information. As a result, the resume will look professional and aesthetic.

Visual Studio Code was chosen as the environment for the implementation of the diploma project. JavaScript programming language was used to develop the project structure. Also used is a library for JavaScript - React. The site structure, interface and appearance are implemented using HTML markup language and CSS appearance description language.

The result is a designed online web designer that allows you to quickly and efficiently create a user resume and download it in PDF format.

WEB CONSTRUCTOR, RESUME, USER, PDF FORMAT, REACT, CSS, HTML, JAVASCRIPT, A PROGRAMMING LANGUAGE.

## ЗМІСТ

Вступ.....	7
1 Аналіз існуючих рішень та постановка завдання.....	9
1.1 Характеристика об'єкту інформації .....	9
1.2 Аналіз існуючих рішень .....	10
1.3 Обґрунтування доцільності створення веб-ресурсу.....	15
1.4 Постановка задачі.....	17
2 Проєктування сайту .....	18
2.1 Проєктування структури сайту.....	18
2.2 Проєктування інтерфейсу .....	20
3 Реалізація та тестування сайту .....	22
3.1 Опис засобів реалізації.....	22
3.1.1 Бібліотека React.....	23
3.1.2 Мова розмітки HTML .....	24
3.1.3 Розширена мова розмітки XML.....	26
3.1.4 Мова програмування javascript.....	27
3.1.5 Мова стилю CSS.....	28
3.1.6 Редактор вихідного коду – Visual Studio Code .....	28
3.2 Налаштування Visual Studio Code .....	30
3.3 Тестування сайту.....	34
4 Техніко-економічне обґрунтування .....	38
4.1 Аналіз ринку .....	38
4.2 Розрахунок витрат на проєктування .....	39
4.3 Обґрунтування необхідності розробки .....	43
Прелік джерел посилання .....	45
Додатки.....	46

					ДП. КН 21. 440.03.000 ПЗ		
Зм.	Арк.	№ докум.	Підпис	Дата	Вебконструктор професійного резюме		
Розроб.		Кухарський Р. В.					
Перев.		Чубей О.О.					
Рецензет.		Кузик В.М.					
Н. Контр.		Кульчинська Н.З.					
Зав. від.		Чубей О.О.			ГК.КВТ.К-47		
					Лім.	Арк.	Аркуші
						5	58

## СКОРОЧЕННЯ І УМОВНІ ПОЗНАКИ

ТЗ – технічне завдання;

VS – visual studio;

HTML – мова розмітки гіпертексту;

CSS – каскадні таблиці стилів;

JS – JavaScript;

ПК – персональний комп'ютер;

ОС – операційна система;

REACT – бібліотека JavaScript;

ПП – програмний продукт.

					<i>ДП. КН 21.440.03.000 ПЗ</i>	Арк.
						6
Змн.	Арк.	№ докум.	Підпис	Дата		

## ВСТУП

У всьому світі компанії намагаються по максимальному скористатися наявними можливостями Інтернету для того, щоб підвищити ефективність своїх бізнес-процесів. І стримуючою причиною тут є історично сформована інфраструктура міжнародної мережі, яка все ще не дозволяє задовільнити необхідний рівень обслуговування відвідувачів і безперешкодно інтегрувати корпоративні інформаційні системи для досягнення реальних переваг. Існуючі по сьогоднішній день корпоративні доповнення утворенні у вигляді ізольованих систем, що підтримують функціонування дискретних бізнес процесів, і не забезпечені знаряддями інтеграції з зовнішніми системами і процесами.

Концепція веб-сервісів народилася після кількох не зовсім вдалих спроб багатьох груп аналітиків, архітекторів і розробників по всьому світу створити середовище і механізм взаємодії того різноманіття інформаційних систем, які вони ж і створили. Така велика кількість людей рідко витрачають стільки енергії, інтелекту і часу просто так – зазвичай, це означає, що передбачуваний результат їх зусиль дуже затребуваний.

Наприклад, сучасне комерційне підприємство важко уявити без інформаційних систем різного призначення: бухгалтерських, фінансово-аналітичних, виробничих, складських та т. д. Велике підприємство використовує багатофункціональні інформаційні системи (можна згадати аббревіатури ERP, CRM, SCM і т. П.), часто одночасно. А є ще постачальники, клієнти, партнери, у яких свої, не менш складні і специфічні, інформаційні системи, і з ними інформаційних систем підприємства необхідно взаємодіяти.

Протягом останніх декількох років World Wide Web зазнає якісних змін. Якщо зовсім недавно "всесвітня павутина" представляла собою головним чином сукупність серверів, що містять статичні документи з посиланнями один на одного, то сучасний Web практично неможливо уявити без інтерактивних Web-

					<i>ДП. КН 21.440.03.000 ПЗ</i>	Арк.
						7
Змн.	Арк.	№ докум.	Підпис	Дата		



додатків, що обробляють різні запити і поміщають результати їх обробки як в бази даних, так і на Інтернет-сторінки.

Однак, розвиток WWW не зупинилася на Web-застосунках. Взаємна інтеграція бізнесів різноманітних компаній, що відбувається зараз по всій землі, безповоротно тягне за собою появу технологій і стандартів для інтеграції, обслуговуючи їхні додатки і корпоративні інформаційні системи. З'явився сервіс-орієнтований Web, в основі якого лежать дві відносно нові технології – SOAP і XML. Згідно з цим сценарієм Web складається з набору серверів додатків, які обмінюються інформацією в форматі XML за протоколом SOAP.

Основою сервіс-орієнтованого Web є Web-сервіс – набір логічно пов'язаних функцій, які можуть бути програмно викликані через Internet. Інформація про те, які функції надає даний Web-сервіс, міститься в документі WSDL, а для пошуку існуючих Web-сервісів передбачається використання спеціальних реєстрів, сумісних зі специфікацією UDDI.

					<i>ДП. КН 21.440.03.000 ПЗ</i>	Арк.
						8
Змн.	Арк.	№ докум.	Підпис	Дата		

## АНАЛІЗ ІСНУЮЧИХ РІШЕНЬ ТА ПОСТАНОВКА ЗАВДАННЯ

### 1.1 Характеристика об'єкту інформації

В умовах розвитку сучасних інформаційних технологій, невідомою частиною життя став інтернет. Сьогодні потрібну інформацію людина шукає з допомогою Інтернет-ресурсів (професійного, навчального, довідкового чи будь-якого іншого характеру), робить замовлення та придбання тих чи інших товарів, не виходячи з дому читає книжки, а використання Інтернет-реклами – це одна із складових успіху фірми.

Фактично кожна організація має власний веб-сайт, який являє собою сукупність веб-сторінок, об'єднаних між собою, як за змістом, так і навігаційно. Перший у світі сайт [info.cern.ch](http://info.cern.ch) з'явився у 1990 році. Його створив Тім Бернерс-Лі – Батько сучасного Інтернету. Опис нової технології WWW (World, Wide, Web), Автор опублікував на своїй веб-сторінці, заснованої на протоколі передачі даних HTTP, системі адресації URL і мові розмітки HTML. На ресурсі було описано принципи встановлення та роботи серверів і браузерів. Сайт став першим на планеті Інтернет-каталогом, на якому автор розмістив гіперпосилання на інші Інтернет-ресурси, що розпочали своє існування. Фізично сайт може розміщуватися не тільки на одному, а і на декількох серверах. При використанні сучасних інформаційних технологій – це необхідний фактор існування, що дозволяє збільшити поле рекламної діяльності і тим самим привернути додаткових відвідувачів.

В даний час у Всесвітній павутині розміщено декілька мільйонів веб-сервісів і їх число постійно росте. Це особисті сторінки, які містять інформацію про автора, його інтереси. Їх створюють для того, щоб знайти друзів по інтересах, розширити свій кругозір.

До інформаційних веб-сервісів відносяться сайти навчальних закладів, співтовариств, фірм та інше.

					<i>ДП. КН 21.440.03.000 ПЗ</i>	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		9

Сайт-портал – це веб-сервіси, які пропонують значну кількість сервісів, починаючи від поштових скриньок і форумів, закінчуючи каталогом зображень.

Веб-сервіси – програмні процедури, які ідентифікуються URL, публічні інтерфейси та сполучення які визначені та описані мовою-розмітки XML. Характеристика таких систем може бути задана іншими програмними системами, які за рахунок використання повідомлень, взаємодіють з ними, які базуються на XML та ними передаються за допомогою Інтернет-протоколів.

Термін "веб-сервіс" застосовується організацією W3C до багатьох систем, але в зазвичай він стосується серверів та клієнтів, що взаємодіють за допомогою повідомлень протоколу SOAP.

В обох випадках припускається, що є можливість опису доступних операцій у форматі WSDL.

Хоча присутність цієї характеристики не є вимогою SOAP, а скоріше передумовою для автоматичного генерування коду на платформах Java та .NET на стороні клієнта.

## 1.2 Аналіз існуючих рішень

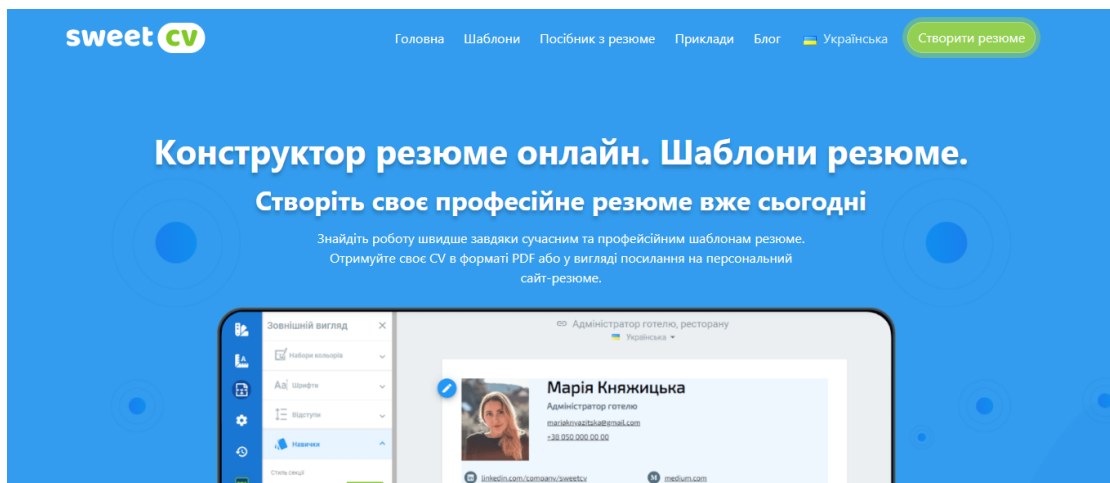
На сьогодні у мережі Інтернет розміщено доволі велика кількість веб-сервісів різної тематики. Кожен з них має як свої плюси, так і мінуси, які стосуються не лише дизайну, а ще й змістового наби́ття та структури.

Для аналізу існуючих рішень, було вибрано веб-сервіс «Sweet CV». Даний веб-ресурс почав свою діяльність лише у 2010 році, але розвивався дуже інтенсивно і встиг отримати значну популярність серед користувачів по всій мережі.

Головна його ціль – створення автоматизованого онлайн резюме. «Sweet CV» дає можливість клієнтам вносити особисті дані у професійні резюме та розміщувати їх на сайті, щоб допомогти працевлаштуватися.

					ДП. КН 21.440.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		10

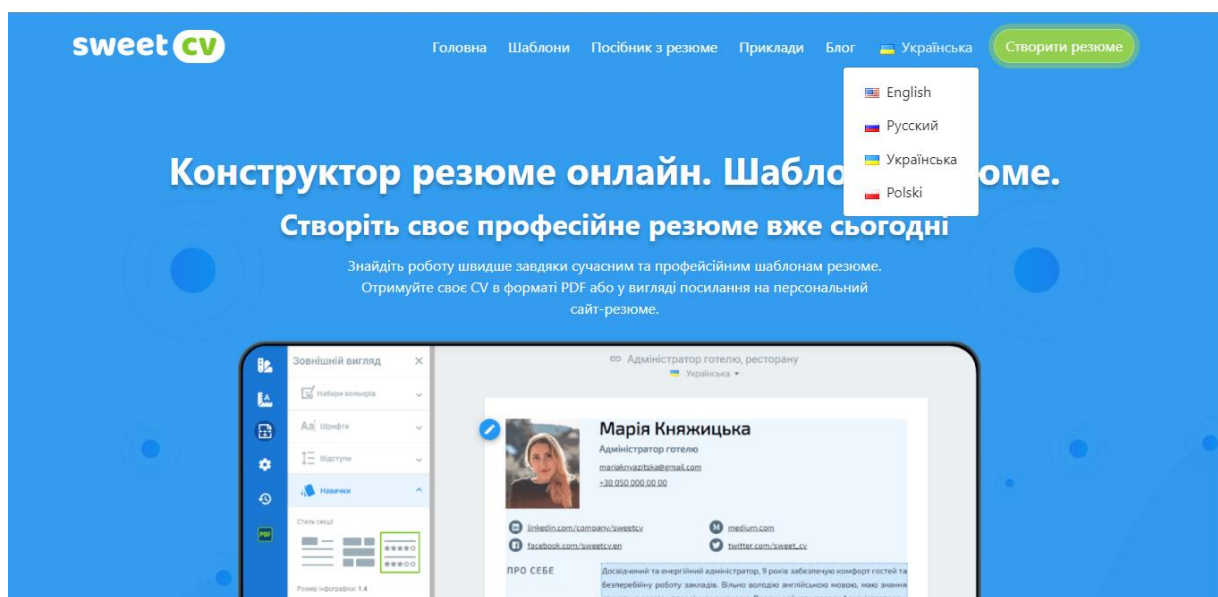
Адреса цього ресурсу – <https://sweetcv.com/ua/> . З лівої сторони розміщені приклади резюме (рисуюнок 1.1).



Рисуюнок 1.1 – Головна сторінка веб-сервісу «Sweet CV»

Проаналізувавши даний сайт, можна виділити такі його особливості:

- контент сайту можна представити чотирма мовами (рисуюнок 1.2);
- присутня вкладка «Шаблони», в якій користувач може обрати бажаний (рисуюнок 1.3);
- є можливість реєстрації для користувачів (рисуюнок 1.4)



Рисуюнок 1.2 – Вибір мови для перекладу контенту

					ДП. КН 21.440.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		11

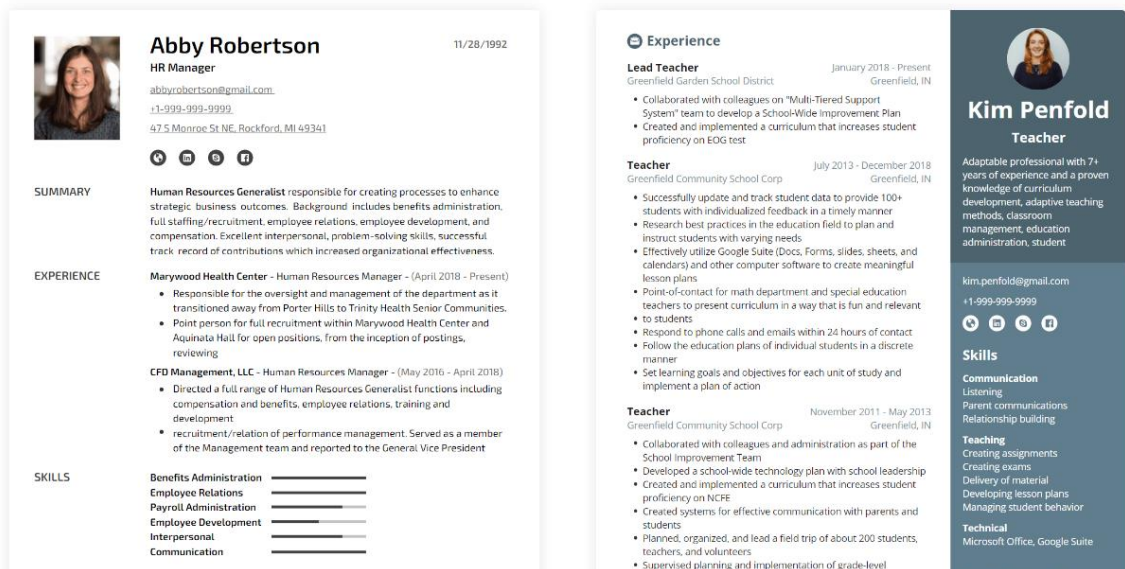


Рисунок 1.3 – Інтерфейс вкладки «Шаблони»

Проте, на мою думку, даний сервіс містить один недолік, а саме, обов'язкову реєстрацію користувача перед початком створення особистого резюме (рисунок 1.4).

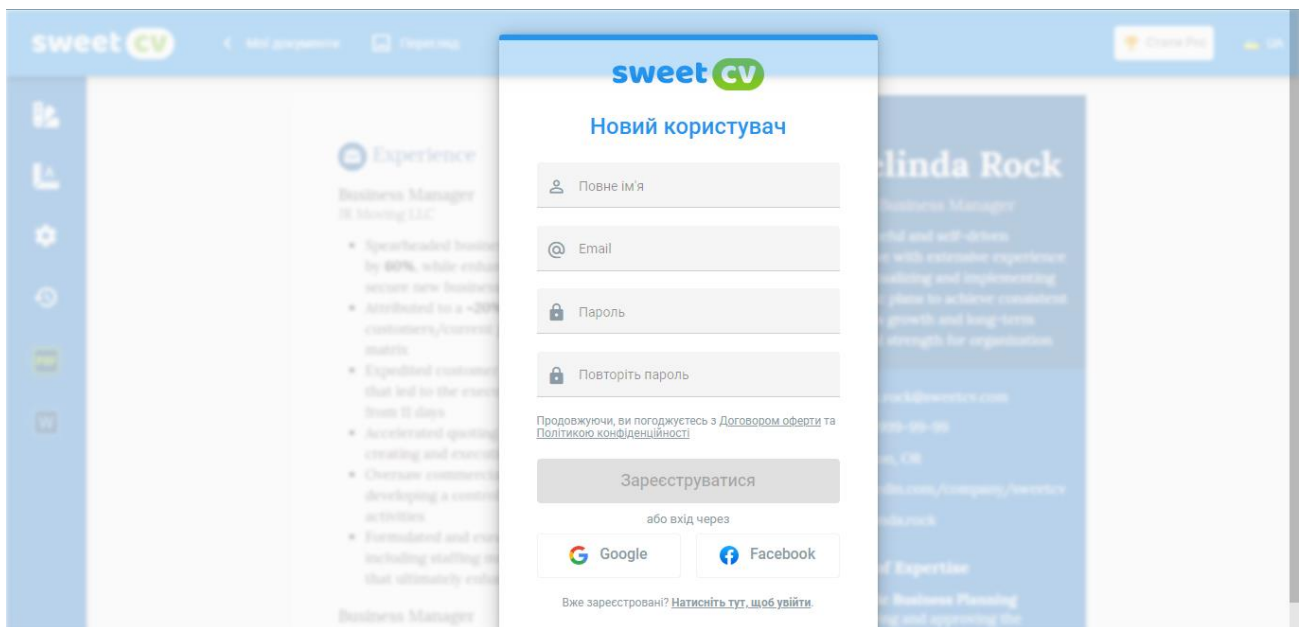


Рисунок 1.4 – Вікно для реєстрації користувача

Також, для аналізу існуючих рішень, було обрано сайт онлайн резюме «CV2you». Даний веб-ресурс набув своєї популярності через те, що містить все необхідне: шаблони з різними дизайнами, підказки, приклади. Вас буквально ведуть за руку по кроках.

Конструктор дозволяє змінювати розташування блоків та їх послідовність в середині шаблону, а також забезпечує автосортування навичок. З дизайнерських опцій – вибір шрифту і колірного стилю. При цьому все просто та інтуїтивно зрозуміло.

Заповнивши всю інформацію про себе, ви отримуєте готове резюме онлайн.

Адреса даного ресурсу <https://cv2you.com/uk-ua/>. З правої сторони розміщений «сайдбар», який відображає приклад резюме[1] (рисунк 1.5)

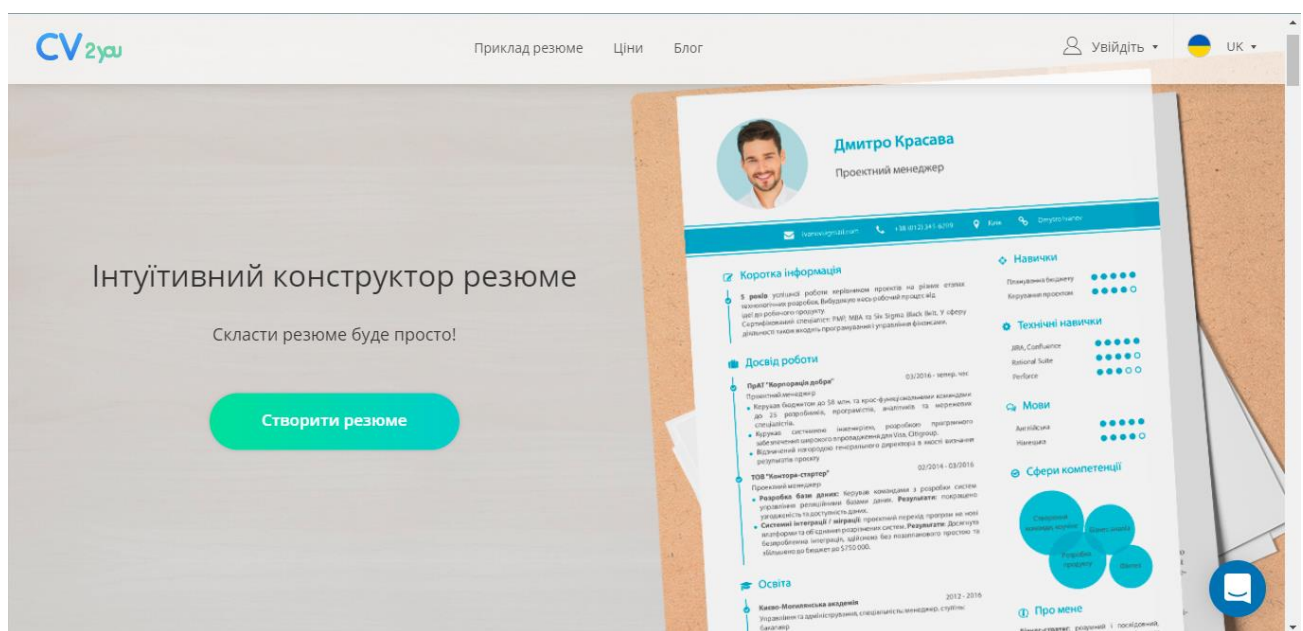


Рисунок 1.5 – Головна сторінка веб-ресурсу «CV2you»

Проаналізувавши даний сайт, можна виділити такі його особливості:

- контент сайту перекладається багатьма мовами (рисунк 1.6);
- присутній розділ «Блог», який містить покрокові поради по створенню резюме (рисунк 1.7).

					ДП. КН 21.440.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		13

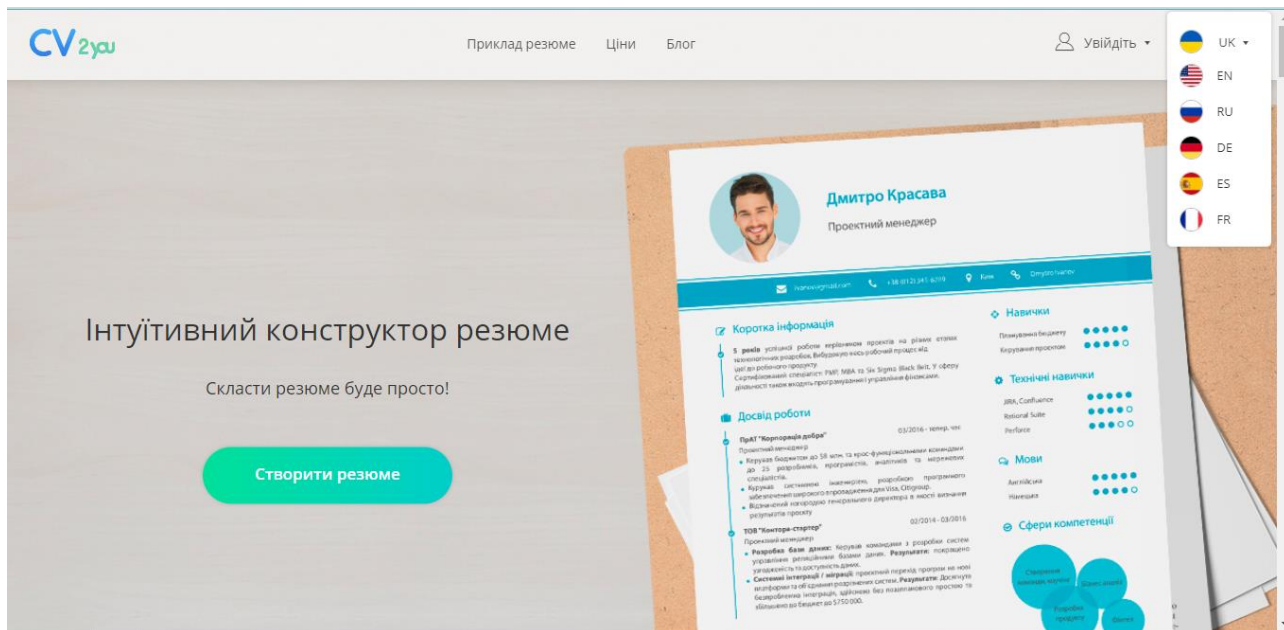


Рисунок 1.6 – Вибір мови для перекладу контенту

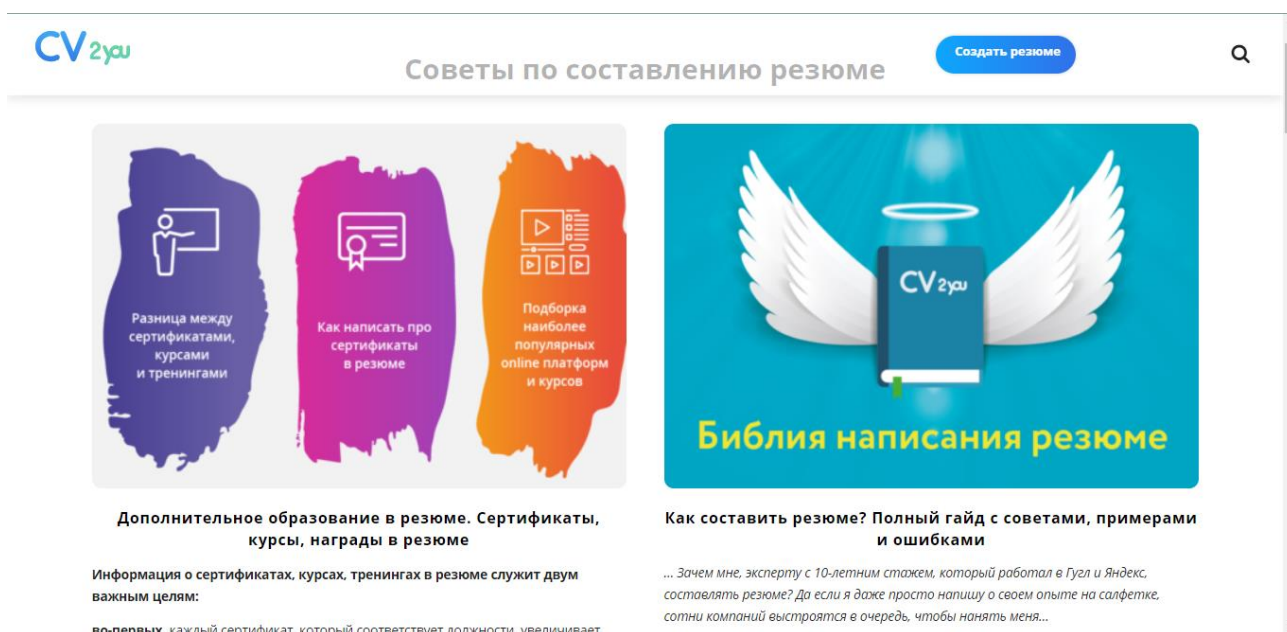


Рисунок 1.7 – Интерфейс розділу «Блог»

Недоліком даного сервісу є присутність розділу «Ціни», де усі шаблони професійного резюме є платними і лише один безплатний (рисунок 1.8).

Змн.	Арк.	№ докум.	Підпис	Дата

ДП. КН 21.440.03.000 ПЗ

Арк.

14



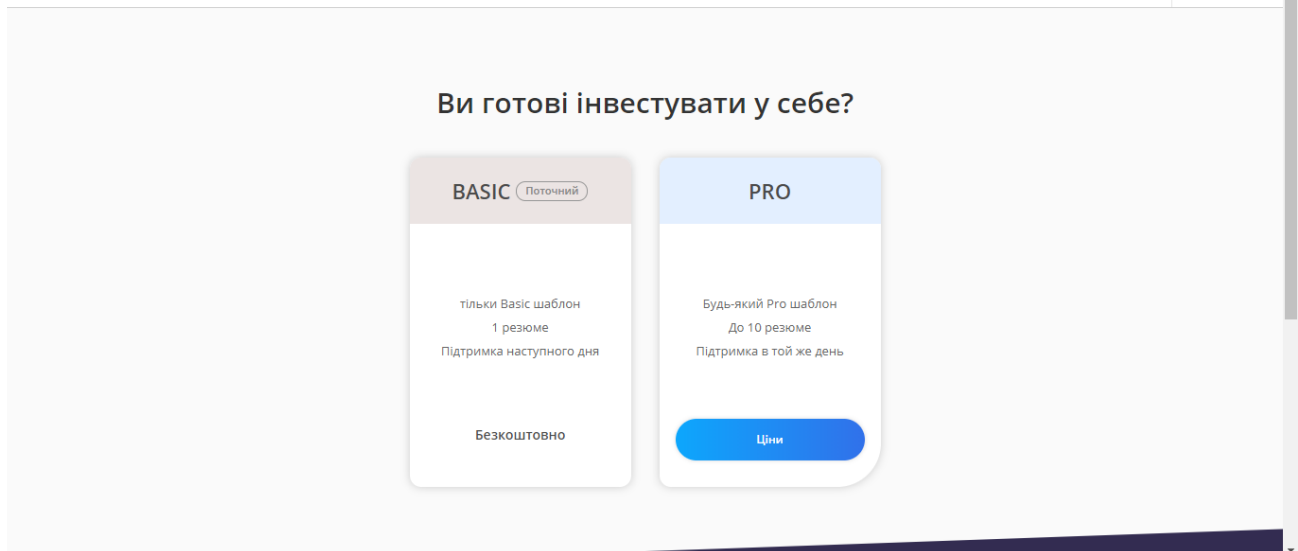


Рисунок 1.8 – Зовнішній вигляд розділу «Ціни»

### 1.3 Обґрунтування доцільності створення веб-ресурсу

Сьогодні людині в пошуках тієї чи іншої інформації варто просто звернутися до мережі Інтернет, де вона зможе отримати відповідь на будь-яке запитання. Саме тому багато компаній приходять до такого рішення, як створення сайту, щоб мати можливість швидко і доступно донести до людей інформацію своєї організації, послуги чи товар.

У свою чергу, в 21 столітті у фірми, яка не має сайту, просто немає майбутнього. Інтернет вже протягом довгого часу пов'язує покупців з продавцями, а замовників з виконавцями. Більш того, сайт не тільки приносить більший дохід, а й постійно залучає нових клієнтів. Витрати на сайт значно нижчі, ніж, наприклад, на оренду площі в бізнес-центрі.

На сьогоднішній день пошук роботодавцями кваліфікованих спеціалістів здійснюється теж за допомогою веб-ресурсів, а отже, для створення професійної анкети оптимальним варіантом є онлайн резюме. Цей спосіб дозволяє швидко і грамотно скласти повноцінний опис власних знань, навичок і досвіду для успішного майбутнього працевлаштування.

Онлайн резюме – це готовий шаблон, який включає кілька розділів, які ви заповнюєте самостійно.

					ДП. КН 21.440.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		15



Найчастіше до цього переліку включено такі дані:

- біографічні
- освіта;
- досвід роботи;
- специфічні навички (володіння ПК, водіння авто, знання іноземної мови);
- особистісні якості і т.д.

Такий тип резюме дає можливість отримати точну і доступну анкету. Головне, бути уважним при наборі тексту у відповідному вікні і не допускати граматичних помилок.

Зазвичай до основної частини резюме можна прикріплювати супровідний лист або більш детальну інформацію про досягнення. А також зразки робіт, якщо ви представник творчої або наукової сфери діяльності.

Після відправлення відомостей про вас, роботодавець отримує допуск до вашої інформації і, якщо вона його зацікавить, зв'язується з вами безпосередньо. Для цього використовується ваш контактний номер телефону або адреса електронної пошти.

Перш ніж говорити про доцільність, варто визначити саме поняття цього терміну і призначення такого Інтернет-ресурсу. Сайт даної категорії – це генератор набору даних, який дозволяє користувачеві швидко створити резюме.

Передумовами для створення сайту стали наступні фактори:

- зростання попиту на кваліфікованих фахівців, які шукають роботу;
- широкий спектр пропозицій від роботодавців;
- автоматизованість процесу створення резюме.

Тому створення даного веб-ресурсу є досить актуальним.

					<i>ДП. КН 21.440.03.000 ПЗ</i>	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		16

#### 1.4 Постановка задачі

Завданням дипломного проєкту – є створення нового веб-сервісу, який би дозволяв користувачам створювати власні резюме, з метою подальшого зацікавлення роботодавця.

Під час створення та розробки сайту потрібні певні знання в області програмування, досвід роботи з графічним пакетом програм, крім того, процес створення вимагає фантазії, естетичного смаку та вміння втілювати ідеї в життя. При цьому варто враховувати зручність і простоту користування веб-сервісом.

Проектування та розробка сайту включає:

- затвердження початкового технічного завдання на розробку сайту;
- визначення структурної схеми сайту;
- розташування контенту;
- проектування дизайну;
- створення графічних та стилів елементів сайту;
- тестування сайту.

Для реалізації завдання необхідно:

- обрати інструментальні засоби для реалізації сайту;
- побудувати структурну схему сайту на основі обраної платформи;
- створити шаблон сторінки сайту;
- встановити додаткові плагіни для функціональності сайту;
- розмістити необхідну інформацію на сайті.

Розроблений сайт повинен надавати користувачам змогу швидко та без проблем створювати резюме, а також завантажувати його у форматі PDF.

					<i>ДП. КН 21.440.03.000 ПЗ</i>	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		17

## 2 ПРОЄКТУВАННЯ САЙТУ

### 2.1 Проєктування структури сайту

Оскільки, онлайн-резюме, як документ із списком особистісних умінь і якостей, повинен зацікавити майбутнього роботодавця, то для зручності його краще спроектувати у вигляді веб-сервісу, який забезпечить оперативність та вдале візуальне оформлення. В останні роки спостерігається динамічне зростання веб-сервісів, розроблення яких стало носити поширений характер і загостило проблему їх якості. Складність створення сайту вимагає застосування певних методів роботи над ним, головним з яких є обдумане проєктування. Високоякісне проєктування – важливий момент створення веб-сервісу. Проєктування сприяє ясності, чіткості і легкості при проведенні робіт з програмуванням та формуванням сайту в цілому. Це зображено на рисунку 2.1

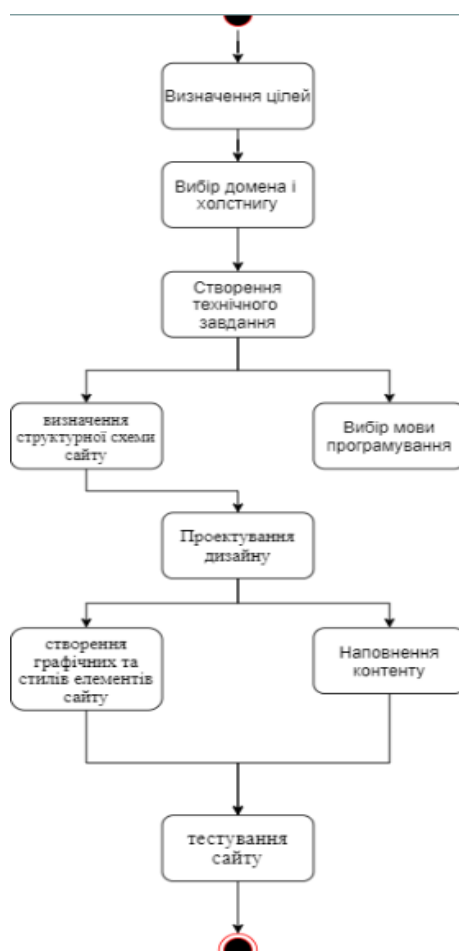


Рисунок 2.1 – Діаграма діяльності

Етапи розробки веб-сервісу було розподілено на такі складові:

1) попередній етап створення веб-сервісу (на цьому етапі розкриваються проблеми загального характеру. Обговорюється загальна структура веб-сервісу, утворюються та фіксуються цілі створення веб-сервісу);

2) період проєктування веб-сервісу (формулювання структури веб-сервісу: меню, концепцію, розміщення модулів, побудова списку компонентів, що підключаються, тощо);

3) етап розробки й тестування веб-сервісу;

4) розміщення веб-сервісу;

5) розвиток ресурсу.

Для того щоб веб-сервіс став доступним широкому колу відвідувачів, йому необхідно призначити доменне ім'я і розмістити в мережі Інтернет.

Розміщення сайту на сервері та подальше його адміністрування називають хостингом. Наданням такої послуги займаються спеціальні організації. Хостинг буває платним і безкоштовним.

Можливість створювати веб-сторінки та організовувати форуми й чати в автоматизованому режимі (крім власне розміщення веб-сервісу) часто надається на серверах, що забезпечують хостинг. Наприклад, ucoz.ru, mylivepage.com.

На головній сторінці веб-сервісу «Resumebuilder» відображається основна інформація про даний сайт. При відкритті сторінки «Ваше резюме» користувач потрапляє на сторінку з генератором резюме, де він може заповнити усі необхідні поля для створення резюме.

Особливої уваги заслуговує етап концептуального проєктування.

Правильне визначення мети, задач, цільової аудиторії та інтересів груп користувачів являється запорукою досягнення результату виходу в Інтернет-простір. Але саме даний етап на практиці зводиться до формального опису спрощеного завдання на розробку сайту.

Відсутність концепції є причиною того, що сайт залишається без уваги, нецікавий відвідувачеві.

					ДП. КН 21.440.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		19

## 2.2 Проектування інтерфейсу

Інтерфейс повинен мати чітку і зрозумілу структуру. Таку, щоб будь яка людина, яка зайшла на сайт, могла легко та без проблем створити власне резюме, яке надалі буде її візитівкою та запорукою успіху. Тексти повинні бути максимально прості і легкі до сприйняття.

Потрібно лише ввести запит в Google і отримуємо перелік веб-сервісів, які можуть надати потрібну послугу або товар, заходимо на сайт і вся інформація в наявності. Все просто і зручно.

Сайт – це не тільки красивий вигляд або картинка, перш за все це веб-інтерфейс з різноманітними вимогами. Дизайн сайту є одним із напрямів Web-розробки, в задачу якої входить проектування веб-інтерфейсів користувача та веб-додатків. Дизайн та створення сайту вимагають значного досвіду для того, щоб зробити щось насправді якісне.

Знати, які основні вимоги ставить веб-дизайн і що є запорукою гарного проектування буде корисно, як дизайнеру-початківцю так і клієнту-замовнику.

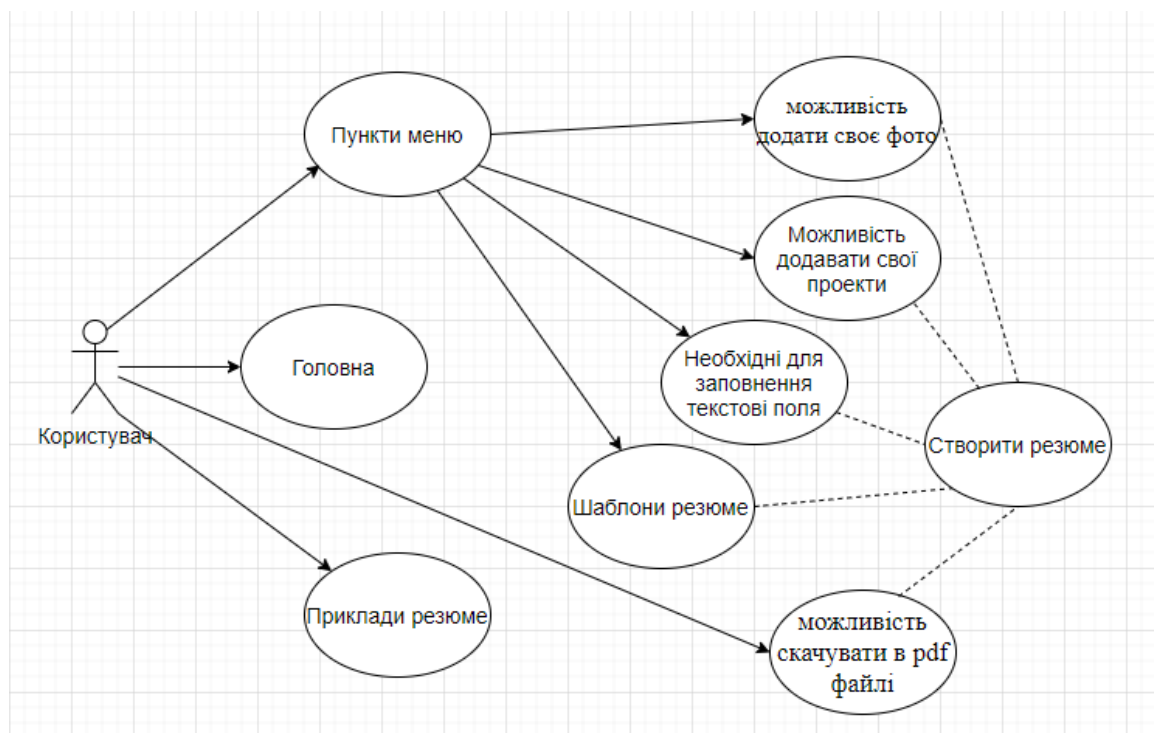


Рисунок 2.3 – Діаграма варіантів використання

Сайт повинен містити меню з розділами, де з кожної сторінки можна було б повернутися на головну. Необхідно, щоб меню було зручним і помітним,

розташовуватись воно повинно зверху сторінки або збоку, для того, щоб користувачеві не довелося його шукати. Якщо на сайті багато розділів із підрозділами, необхідно зробити підменю.

В текстах необхідно використовувати посилання, які повинні бути виділені одним шрифтом, відмінним від шрифту тексту, для того щоб користувач розумів що це.

Сайт необхідно створювати в єдиному стилі, а сторінки повинні мати приблизно однаковий вигляд. Допускається створення головної сторінки в іншій стилістиці, але повинні бути витримані єдині кольори та шрифти. Можна використовувати певні оформлення, проте, знову ж, в межах одного стилю. Необхідно також виділяти на сторінках головне. Для цього можна використовувати колір, шрифт, зображення і т.д., а другорядне – "приховувати", щоб воно не заважало користувачу, але при бажанні він міг дізнатися більше.

Щоб дизайн сайту був насправді якісним необхідно вміло оформити контент, тобто правильно розмістити тексти, таблиці, зображення тощо. Головним є те, що все це повинно гармонійно поєднуватись та мати свої функції. Проте дана структура є універсальною і зображена на рисунку 2.3.

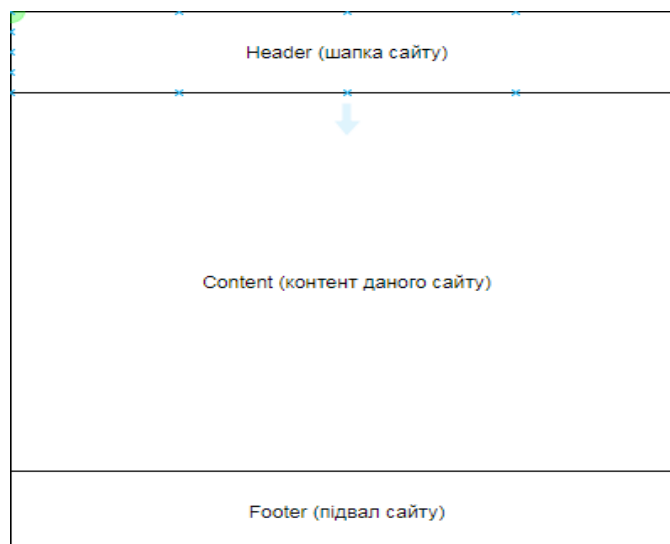


Рисунок 2.4 – Схема структури інтерфейсу сайту

## 3 РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ САЙТУ

### 3.1 Опис засобів реалізації

Для реалізації завдання дипломного проєкту необхідно використати таке інструментальне програмне забезпечення:

- система керування вмістом – для організації веб-сайтів чи інших інформаційних ресурсів в інтернеті;
- локальний веб-сервер, що забезпечить роботу над сайтом в період його розробки та тестування.
- середовище для написання коду
- усі необхідні плагіни для комфортного використання веб-сервісу

Загальний вигляд дизайну веб-сервісу зображений на рисунку 3.1

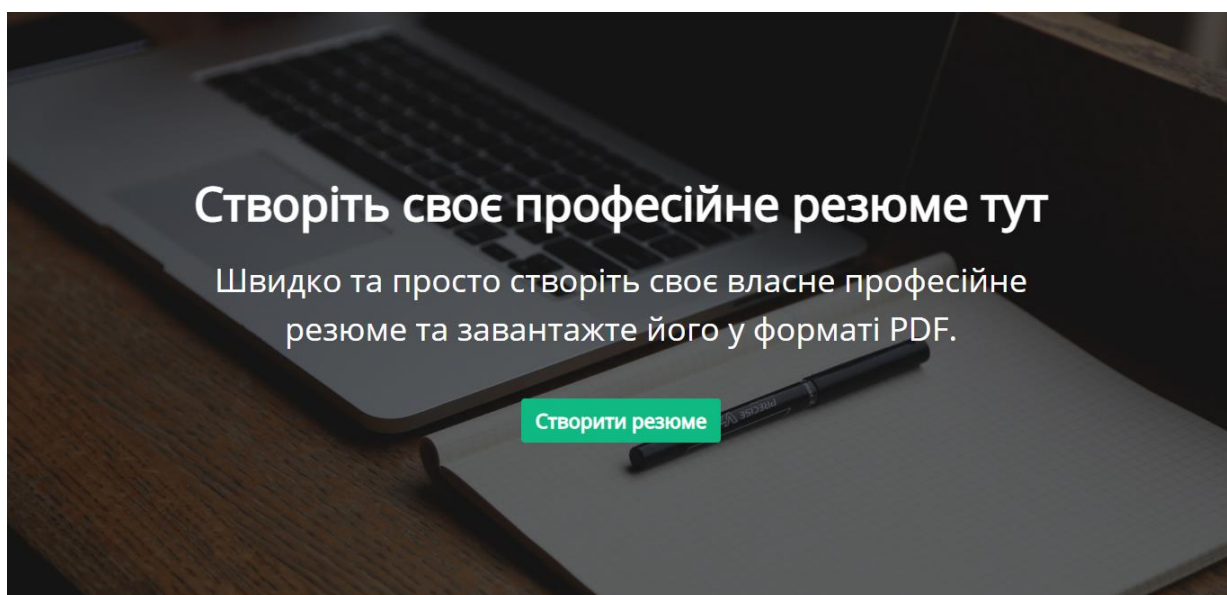


Рисунок 3.1 – Головна сторінка сайту

У структуру сайту входять наступні розділи: Головна, «Створити резюме».

Сайт є україномовним, містить загальні принципи дизайну, а зокрема: типовість усіх сторінок, структуру головного меню, розміщення елементів.

Всі мови програмування, які використовуються при розробці Web-сайтів, можна розділити на дві великі групи.

					ДП. КН 21.440.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		22

До першої належать ті з них, код яких виконується на комп'ютері користувача сайту – в його браузері. Це всім відомі JavaScript, VBScript, Java-аплети. Програми, написані на цих мовах, вбудовуються в код Web-сторінок (або виносяться в окремий файл з викликом з Web-сторінки).

До другої групи потрапляють ті мови, програми на яких виконуються на тому комп'ютері, де розташований Web-сервер. Ця група набагато ширше, саме до неї і відноситься PHP. Однак від своїх побратимів він вигідно відрізняється своєю простотою. Створюючи програми на цій мові, немає необхідності враховувати всі ті численні дрібниці, які отруюють життя програмістам на C++ або Perl: не треба піклуватися про права доступу до файлів сценарію або прописувати точні шляхи до різних модулів і т.д.

Для тестування PHP-програм на локальному комп'ютері, не обов'язково встановлювати на ньому ту ж операційну систему, що і у провайдера хостингу сайту – достатньо лише завантажити відповідний Web-сервер і підключити до нього модуль PHP, і повністю зосередитися на вирішенні задачі.

Програми, написані на PHP, виконуються на стороні сервера, тобто починають працювати до завантаження сторінки після подачі вашого запиту (наприклад, введення адреси). Вони обробляються програмою-сервером і як відповідь на ваш запит видають динамічно (тобто в залежності від запиту, інтерактивно) генеровані сторінки. Ці сторінки можуть містити як стандартні HTML-теги, так і «скрипти» (маленькі програми WEB-додатки, які виконуються на стороні клієнта), написані на JavaScript або VBScript, які почнуть виконуватися після повного завантаження браузером згенерованої на сервері сторінки.

### 3.1.1 Бібліотека React

React – бібліотека, написана на JavaScript, яка використовується для роботи з інтерфейсами. У 2011 році її почали використовувати для соціальної мережі Facebook, а вже в 2013 році бібліотеку виклали у відкритий доступ і ентузіасти з усього світу почали створювати інструменти для розширення її можливостей.

					<i>ДП. КН 21.440.03.000 ПЗ</i>	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		23



Розробники використовують React, щоб створювати інтерфейси, які здатні змінювати контент без перезавантаження сторінки. Завдяки цьому сайти або наявні додатки швидко відгукуються на дії користувачів. Можна додавати товари в кошик без перезавантаження сторінки або заповнювати форми без переадресації.

React вважається найпопулярнішою бібліотекою в світі, написаної на JS. Її популярність забезпечується тим, що за допомогою компонента можна вирішувати різні завдання. Особливо, якщо використовувати додаткові інструменти, які інтегруються в проєкти і відкривають доступ до нестандартних можливостей.

Одне з головних переваг бібліотеки в тому, що її можна використовувати де завгодно. Наприклад, на своєму сервері при створенні нестандартних веб-сайтів, які складно реалізувати на базі готових систем управління контентом. React часто використовується для розробки сервісів, де важлива швидкість роботи, а не красивий інтерфейс.

На мобільних платформах React теж можна використовувати. Існує React Native, який використовується для створення кроссплатформенних додатків. Він з'явився на 2 роки пізніше «старшого брата». Спочатку була реалізована підтримка iOS, але пізніше розробники додали сумісність з Android.

Виходить, що React можна використовувати всюди, якщо один раз витратити час на вивчення базових особливостей. У нативної бібліотеки є свої нюанси, але знання «ядра» дозволить працювати з додатками і сайтами[4].

### 3.1.2 Мова розмітки HTML

HTML – це аббревіатура, що розшифровується як Hyper Text Markup Language, яка використовується для створення веб-сторінок та веб-програм. Для більш ширшого представлення пропоную розглянути, що мається на увазі під мовою розмітки гіпертексту та веб-сторінкою.

Гіпертекст: HyperText означає просто "Текст у тексті", тобто текст, що має в собі посилання. Кожного разу, коли натискаємо на посилання, воно приводить

					ДП. КН 21.440.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		24

нас до нової веб-сторінки. HyperText – це спосіб зв’язування двох або більше веб-сторінок (HTML-документів) між собою.

Мова розмітки – це комп’ютерна мова, яка використовується для застосування конвенцій щодо розмітки та форматування текстового документа. Мова розмітки робить текст більш інтерактивним та динамічним. Вона може перетворювати текст на зображення, таблиці, посилання тощо.

Веб-сторінка – це документ, який зазвичай пишеться у форматі HTML і перекладається веб-браузером. Веб-сторінку можна ідентифікувати, ввівши URL-адресу, вона може бути статичного або динамічного типу. Лише за допомогою HTML ми можемо створювати статичні веб-сторінки[6].

Отже, HTML – це мова розмітки, яка використовується для створення привабливих веб-ресурсів за допомогою стилю і яка привабливо виглядає у форматі у веб-браузера. Документ HTML складається з безлічі тегів HTML.

Мова розмітки документів – це набір спеціальних вказівок, названих тегами, призначених для сформування в документах якоїсь структури і визначення зв’язку між різноманітними елементами цієї структури. Теги мови, або, як їх іноді називають, управляючі дескриптори, у таких документах якимось чином кодуються, виділяються щодо основного вмісту документа і служать у якості інструкцій для браузера.

Недосконалість HTML призвела до того, що мережа трансформувалася в мішанину тексту, повну багатогранних елементів і тегів, часто використовуваних, що називаються Pro Forma і нічого не значущих.

XML має величезне майбутнє для удосконалення гіпертексту. Наприклад у HTML для формування зв’язку використовується елемент A, приміром, не просто посилання, XML, що дозволяє створити двонаправлений зв’язок.

Перспектива XML полягає в тому, що він буде використовуватися для опису інших мов розмітки, наприклад, JavaScript, що використовується в HTML – документах.

					<i>ДП. КН 21.440.03.000 ПЗ</i>	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		25

XML розроблений для того, щоб спростити і полегшити використання SGML, при цьому зберігши його великі можливості по створенню, поширенню і публікації Web-документів мережі.

### 3.1.3 Розширена мова розмітки XML

XML означає розширювану мову розмітки. Мова розмітки - це набір кодів або тегів, що описують текст у цифровому документі. Найвідомішою мовою розмітки є мова розмітки гіпертексту (HTML), яка використовується для форматування веб-сторінок. XML, більш гнучкий двоюрідний брат HTML, дозволяє вести складний бізнес через Інтернет.

У той час як HTML повідомляє програмі браузера, як повинен виглядати документ, XML описує, що в документі. Іншими словами, XML стосується того, як організована інформація, а не як вона відображається. (Форматування XML здійснюється за допомогою окремих таблиць стилів.)

Кожен розробник сайту використовує однакові теги, щоб робити однакові дії. XML, навпаки, дозволяє створювати власні теги для позначення значення або використання даних.

Гнучкість XML має багато переваг. Це дозволяє передавати дані між корпоративними базами даних та веб-сайтами, не втрачаючи важливу описову інформацію. Це дозволяє автоматично налаштовувати подання даних, а не відображати одну і ту ж сторінку для всіх бажаючих. І це робить пошук більш ефективним, оскільки пошукові системи можуть сортувати точні теги, а не довгі сторінки тексту.

Оскільки XML привносить на веб-сайти складне кодування даних, це допомагає компаніям інтегрувати свої інформаційні потоки. Створюючи єдиний набір тегів XML для всіх корпоративних даних, інформація може безперешкодно обмінюватися веб-сайтами, базами даних та іншими внутрішніми системами. Але революційна сила XML полягає в підтримці транзакцій між бізнесом. Коли компанія продає товар або послугу іншій компанії, необхідно обмінятися великою кількістю інформації - про ціни, умови, технічні характеристики, графіки

					<i>ДП. КН 21.440.03.000 ПЗ</i>	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		26

доставки тощо. Універсальний характер HTML робить такі обміни важкими, а то й неможливими через Інтернет. За допомогою XML вся необхідна інформація може обмінюватися в електронній формі, що дозволяє укласти складні угоди без будь-якого втручання людини. Ось чому веб-ринки між бізнесами, такі як Ariba та Commerce One, вже покладаються на XML для автоматичного встановлення відповідності між покупцями та продавцями. У недалекому майбутньому ваша компанія може оцінюватися за вмістом її тегів XML.

### 3.1.4 Мова програмування JavaScript

JavaScript – це мова сценаріїв, яка використовується для створення та управління динамічним вмістом веб-сайту, тобто будь-що, що переміщується, оновлюється чи іншим чином змінюється на екрані, не вимагаючи перезавантаження веб-сторінки вручну. Такі функції, як:

- анімована графіка;
- слайд-шоу фото;
- пропозиції щодо автозаповнення тексту;
- інтерактивні форми.

Ще кращий спосіб зрозуміти, що робить JavaScript – це згадати певні веб-функції, якими ми користуємося щодня і, ймовірно, сприймаємо їх як належне. Наприклад, коли хронологія Facebook автоматично оновлюється на екрані або Google пропонує пошукові терміни на основі кількох запущених нами букв – в обох випадках це JavaScript у дії.

Мови сценаріїв – це мови кодування, які використовуються для автоматизації процесів, які користувачі повинні були б виконувати самостійно, поетапно. Якщо не брати сценаріїв, то будь-які зміни на веб-сторінках, вимагатимуть або перезавантаження вручну, або навігації по ряду статичних меню.

Тим чином, оскільки JavaScript є такою невід'ємною частиною веб-функціональності, усі основні веб-браузери мають вбудовані механізми, які

					<i>ДП. КН 21.440.03.000 ПЗ</i>	Арк.
						27
Змн.	Арк.	№ докум.	Підпис	Дата		

можуть відображати JavaScript. Це означає, що команди JS можна вводити безпосередньо в документ HTML, і веб-браузери зможуть їх зрозуміти. Іншими словами, використання JavaScript не вимагає завантаження додаткових програм чи компіляторів[8].

Насправді додавання коду JavaScript на веб-сторінку є досить простим процесом і звичним, якщо ми виконували кодування за допомогою HTML та CSS. JavaScript можна додавати безпосередньо до коду сторінки за допомогою тегів `<script>` і надаючи їм атрибут `type text / javascript`.

### 3.1.5 Мова стилю CSS

CSS – це елемент програмування, який так само важливий, і поряд з HTML та JavaScript, є одним із трьох основних наріжних каменів Інтернету.

CSS означає каскадні таблиці стилів, або CSS – це мова дизайну, завдяки якій веб-сайт виглядає більш привабливим. Якщо HTML значною мірою визначає текстовий вміст, CSS визначає візуальну структуру, макет та естетику. HTML – це мова розмітки, а CSS – мова таблиці стилів[7].

Якби HTML був компонентом двигуна автомобіля, CSS був би стилем кузова та роботою фарби. Веб-сайт може працювати без CSS, але це, звичайно, не красиво. CSS робить інтерфейс веб-сайту блискучим, і це створює чудові зручності для користувача. Без CSS веб-сайти були б менш приємними для ока і, ймовірно, набагато складнішими для навігації. Окрім верстки та формату, CSS відповідає за колір шрифту та багато іншого.

Як бачимо, CSS є критично важливим, коли йдеться про загальну презентацію веб-сторінки. І не тільки це, CSS значно полегшує життя розробників, коли справа стосується форматування. Освоєння CSS є життєво важливою частиною того, щоб стати цінним активом та надійним програмістом.

### 3.1.6 Редактор вихідного коду – Visual Studio Code

Visual Studio Code – розроблений Microsoft для Windows, Linux і macOS. Позиціонується як «легкий» редактор коду для кросплатформенної розробки веб-

					ДП. КН 21.440.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		28

хмарних додатків. Включає в себе відладчик, інструменти для роботи з Gi, підсвічування синтаксису, IntelliSense і засоби для рефакторинга. Має широкі можливості для кастомізації: призначені для користувача теми, поєднання клавіш і файли конфігурації. Розповсюджується безкоштовно, розробляється як програмне забезпечення з відкритим вихідним кодом, але готові збірки розповсюджуються під пропріетарною ліцензією[5].

Visual Studio Code був анонсований 29 квітня 2015 року компанією Microsoft на конференції Build, і незабаром була випущена бета-версія .

18 листопада 2015 року Visual Studio Code був випущений під ліцензією MIT, а вихідний код був опублікований на GitHub. Анонсована підтримка розширень.

14 квітня 2016 року Visual Studio Code вийшов зі стадії бета-тестування.

Visual Studio Code - це редактор вихідного коду. Він підтримує ряд мов програмування, підсвічування синтаксису, IntelliSense, рефакторинг, налагодження, навігацію по коду, підтримку Git та інші можливості. Багато можливостей Visual Studio Code недоступні через графічний інтерфейс, найчастіше вони використовуються через палітру команд або JSON-файли (наприклад, призначені для користувача настройки). Палітра команд представляє собою подобу командного рядка, яка викликається поєднанням клавіш.

Visual Studio також дозволяє замінювати кодову сторінку при збереженні документа, символи перекладу рядка і мову програмування поточного документа.

З 2018 року з'явилися розширення Python для Visual Studio Code з відкритим вихідним кодом. Воно надає розробникам широкі можливості для редагування, налагодження і тестування коду.

На березень 2019 року за допомогою вбудованого в продукт призначеного для користувача інтерфейсу можна завантажити і встановити кілька тисяч розширень тільки в категорії «programming languages» (мови програмування).

Також розширення дозволяють отримати більш зручний доступ до програм, таким як Docker, Git та інші. В розширеннях можна знайти лінтера коду, теми для редактора і підтримку синтаксису окремих мов.

					<i>ДП. КН 21.440.03.000 ПЗ</i>	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		29

### 3.2 Налаштування Visual Studio Code

При першому запуску перед нами відкриється вікно вітання, через яке можна отримати швидкий доступ до раніше запущеним проектам або створити нові. Весь інтерфейс англійською мовою (всього в кілька кліків він переводиться на російську, але про це трохи пізніше). Давайте поки розберемося з інтерфейсом: відкриємо для прикладу файл з кодом, подивимося, як його ідентифікує програма без попередніх налаштувань, і розберемо кожен блок утиліти.

Запустити файл з кодом ми можемо через правий клік миші -> Відкрити за допомогою Code. Файл можна відкрити і в програмі: для цього у верхньому правому кутку вибираємо «File» і тиснемо «Open File ...». Після цього переходимо в потрібну директорію і вибираємо файл.

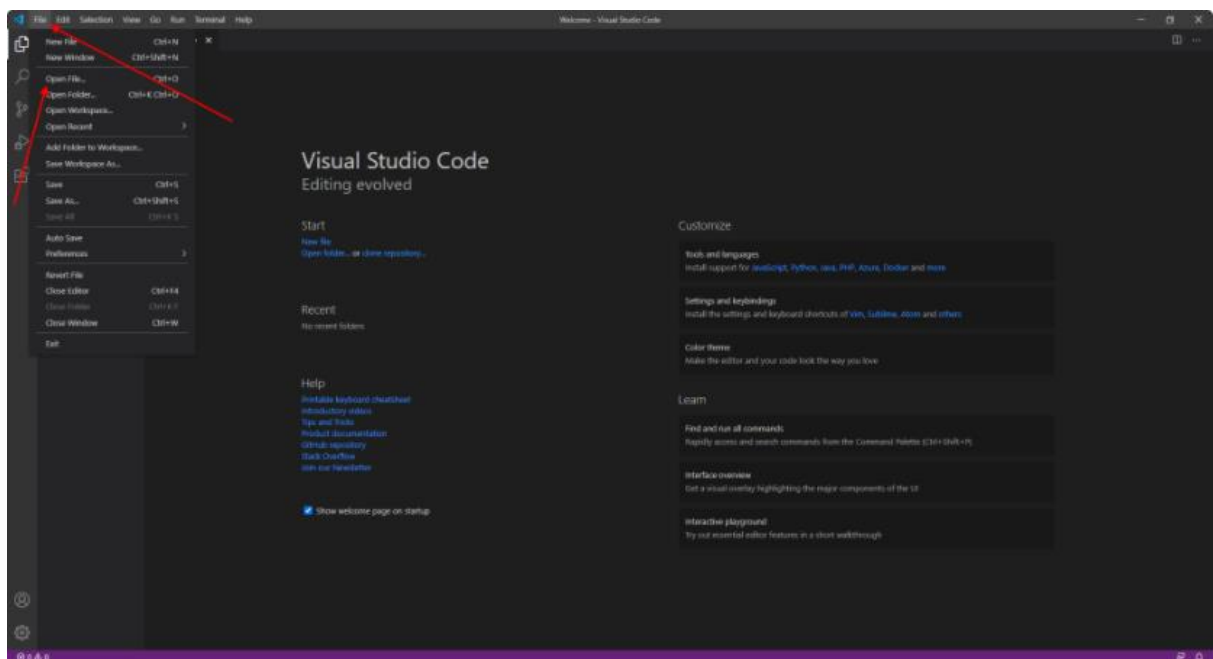


Рисунок 3.2 – Головна сторінка Visual Studio Code

В результаті перед нами відкриється вікно з кодом з файлу. Давайте візуально розділимо інтерфейс на блоки і розглянемо кожен з них більш детально.

Перше, що кидається в очі – це блок коду, для якого відведено найбільше місця. Назва функцій та інше підсвічено різними кольорами. Тут же вказано шлях до файлу, а трохи вище розташовані вкладки-файли, за якими можна переміщатися. За допомогою них ми можемо додавати не просто один файл з кодом, а цілий проект.

					ДП. КН 21.440.03.000 ПЗ	Арк.
						30
Змн.	Арк.	№ докум.	Підпис	Дата		

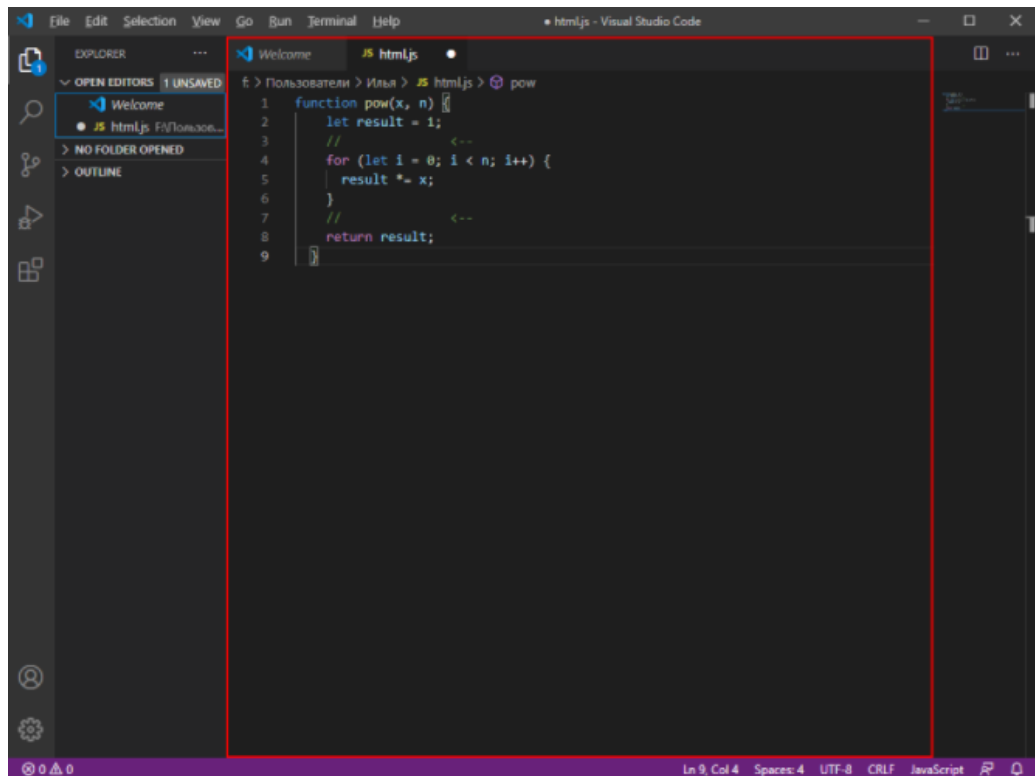


Рисунок 3.3 – Блок коду

Наступний блок – ліва панель управління, що включає в себе 5 основних вкладок: «Провідник», «Пошук», «Система управління версіями», «Запуск коду» і «Розширення». Перша надає користувачеві можливість переглядати вміст проекту і швидко переміщатися всередині нього.

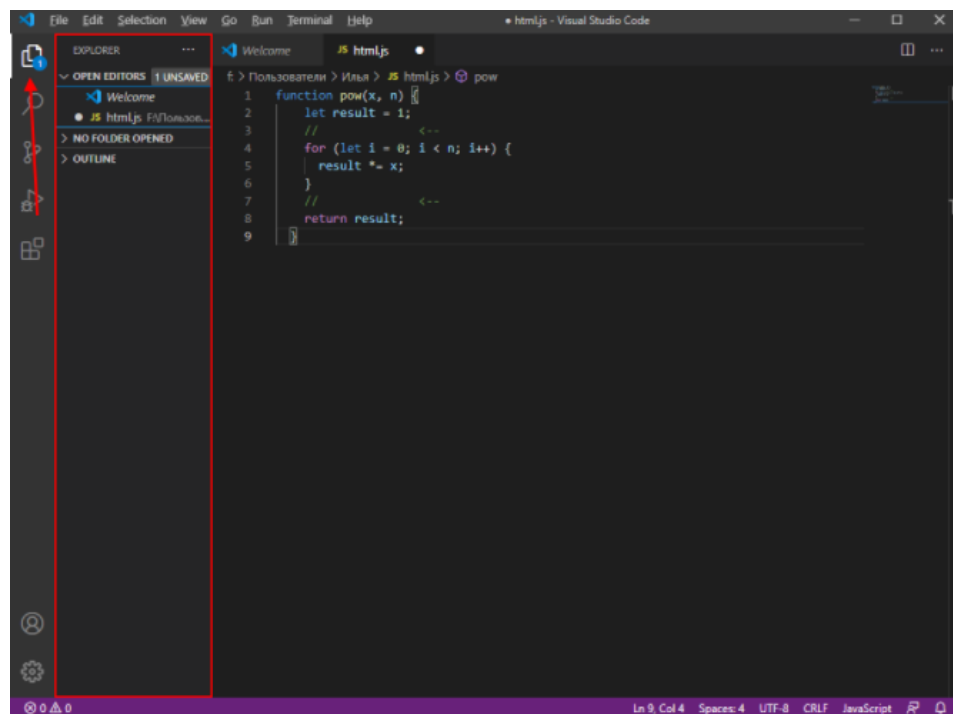


Рисунок 3.4 – Панель управління

					ДП. КН 21.440.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		31



Друга вкладка – пошук. Використовується для знаходження слів в документі. Він також може бути запущений за допомогою комбінації клавіш «CTRL + SHIFT + F». Алгоритм використання досить простий – вводим назву в пошук і дивимось результат.

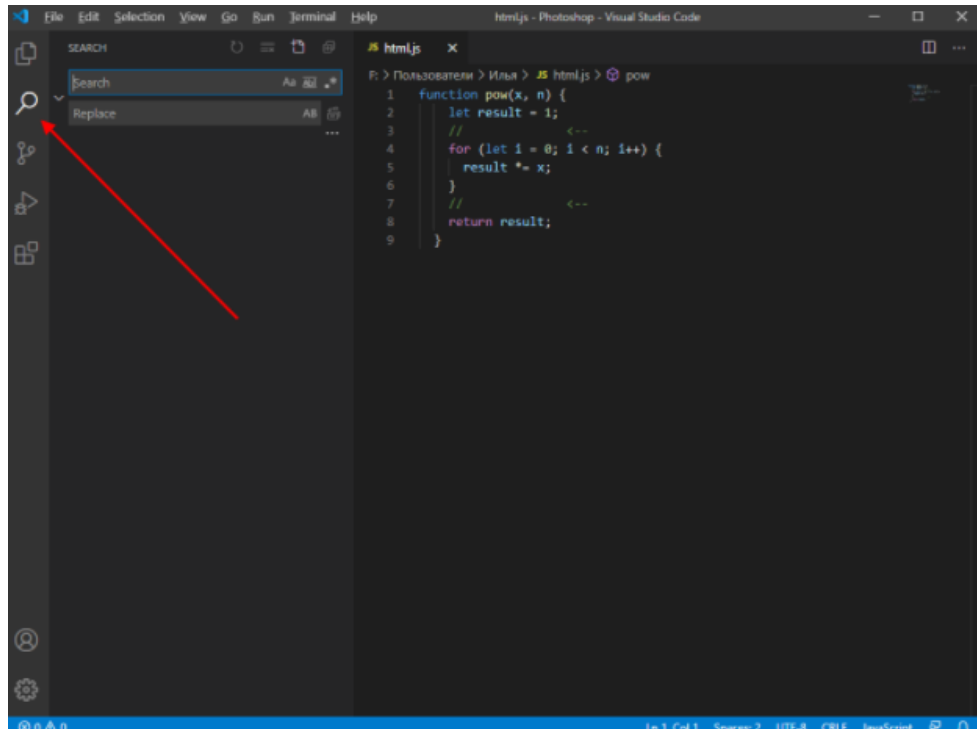


Рисунок 3.5 – Вкладка пошук

Система управління версіями - призначена для взаємодії з git.

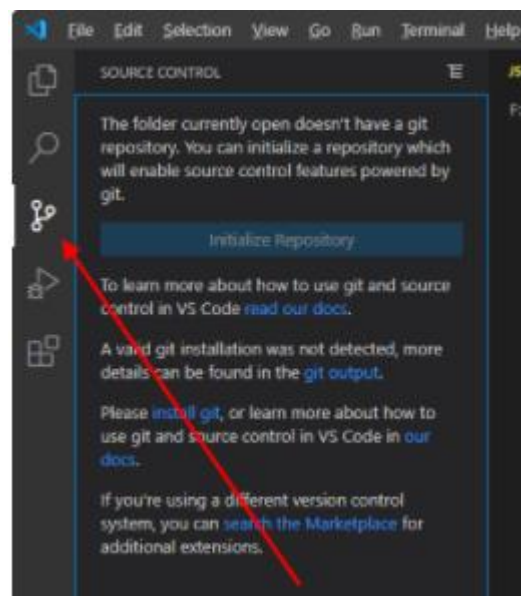


Рисунок 3.6 – Управління версіями

					ДП. КН 21.440.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		32

Запустити і перевірити працездатність коду ми можемо в наступній вкладці:

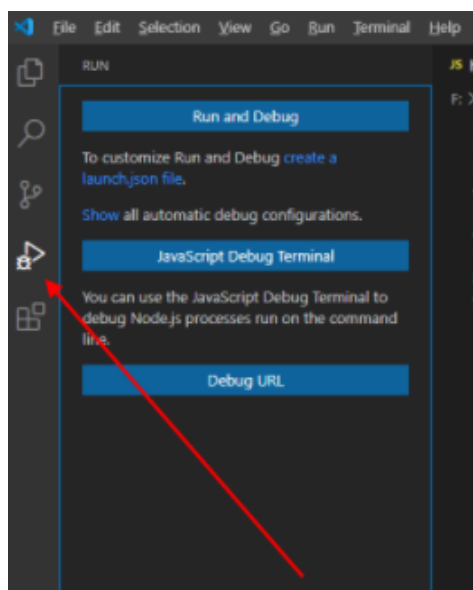


Рисунок 3.7 – Вкладка «Run»

Ще одна важлива вкладка, за допомогою якої з Visual Studio Code де можна зробити функціональну утиліту, це «Розширення». У ній знаходяться установочнікі для всіх популярних інструментів, використовуваних розробниками.

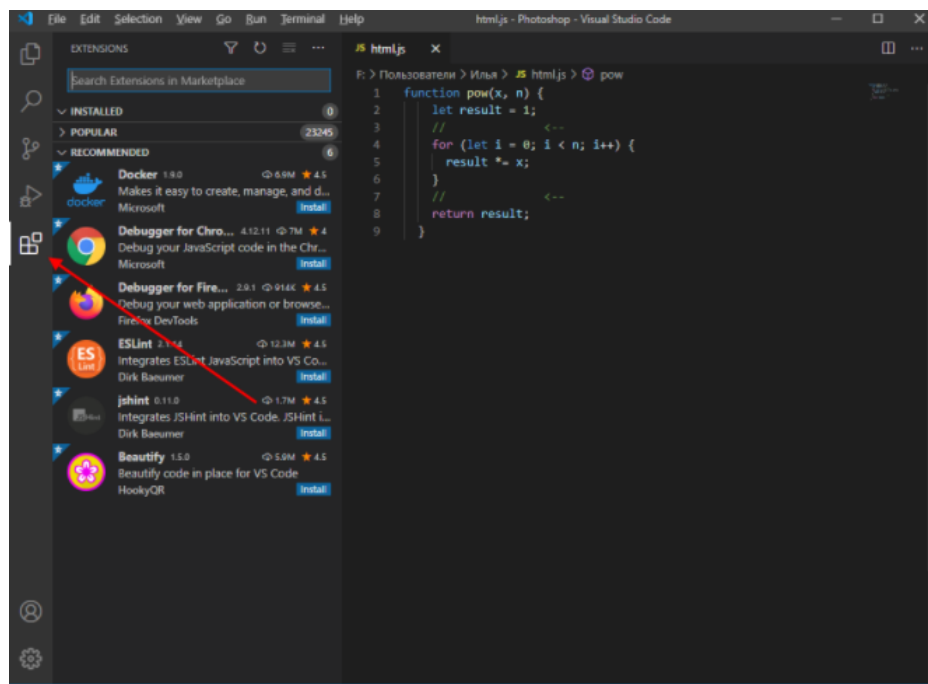


Рисунок 3.8 – «Розширення»

					ДП. КН 21.440.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		33

### 3.3 Тестування сайту

Проведемо тестування сайту згідно наступного плану:

- перевірка сайту різними веб-браузерами;
- працездатність меню
- перевірка роботи додаткових модулів.

Перевіримо підтримку сайту різними веб-браузерами Google Chrome (рисунок 3.9), Safari (рисунок 3.10), Internet Explorer (рисунок 3.11).

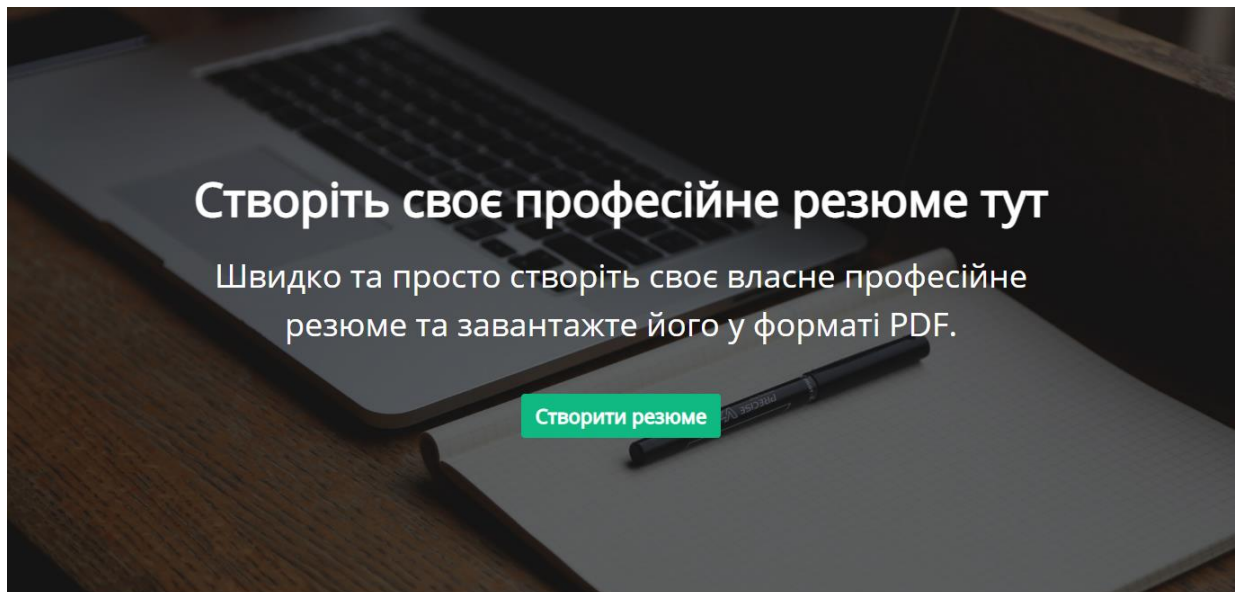


Рисунок 3.9 – Зовнішній вигляд сайту у браузері Google Chrome

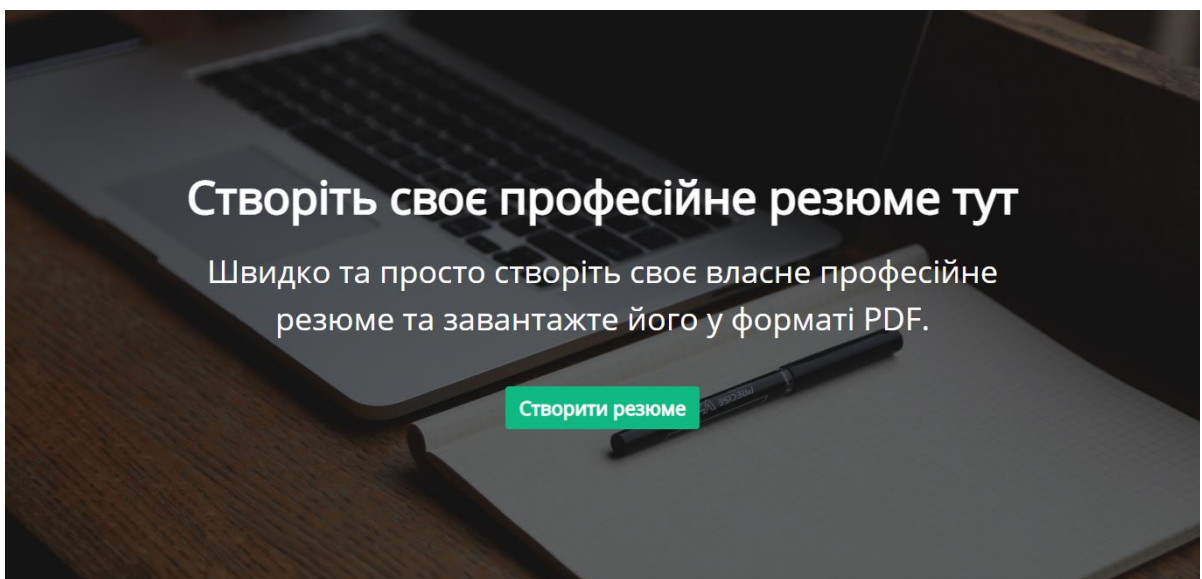


Рисунок 3.10 – Зовнішній вигляд сайту у браузері Safari

					ДП. КН 21.440.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		34

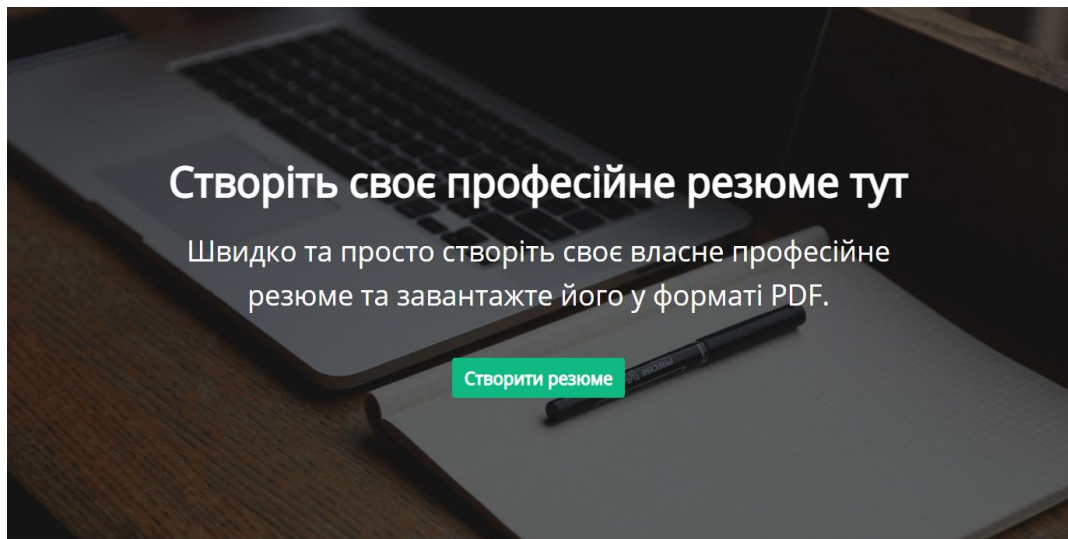


Рисунок 3.11 – Зовнішній вигляд сайту у браузері Internet Explorer

У всіх перевірених браузерах елементи інтерфейсу сайту відображені конкретно.

Також сайт для зручності було модифіковано пі мобільний пристрій (рисунок 3.12)

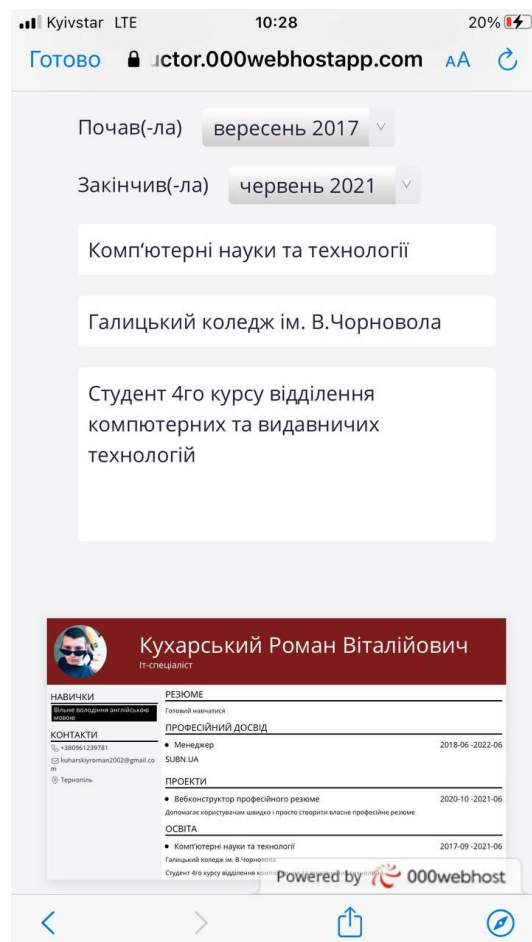


Рисунок 3.12 – вигляд резюме з мобільного пристрою

					ДП. КН 21.440.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		35

В процесі тестування було виявлено, що всі посилання теж працюють коректно. При натисненні на посилання будь якого розділу меню, відбувається перехід за вказаною адресою (рисунки 3.13)

Рисунок 3.13 – Перехід до генератора резюме

При тестуванні сайту в різних браузерах помилок в роботі виявлено не було. Сайт успішно виконує усі передбачені завданням функції. Для того щоб завантажити резюме користувач повинен заповнити усі поля форми (рисунки 3.14).

Рисунок 3.14 – Поля форми для заповнення

Після заповнення форми, користувач має змогу завантажити своє резюме у форматі PDF (рисунок 3.15)

12.06.2021

Резюме



Нікітюк Павло Сергійович

**НАВИЧКИ**  
Вільне володіння англійською мовою

**РЕЗЮМЕ**  
Молодий, енергійний, швидко вчуся готовий до будь якої праці

**ПРОФЕСІЙНИЙ ДОСВІД**  

- Бухгалтер "SEBN.UA" 2019-01 - 2021-02

**ПРОЕКТИ**  

- Вебконструктор професійного резюме 2017-02 - 2019-03


**КОНТАКТИ**  
+380961239781  
kuharskiyroman2002@gmail.com  
Тернопіль, Україна

До спектра моїх обов'язків входить: складати й подавати статичну звітність, вести облік договорів, звіряти бюджетні рахунки  
В даній роботі була поставлена ціль: розробити Web-сервіс по створенню вебконструктора професійного онлайн резюме.

ОСВІТА

Рисунок 3.15 – Формат PDF

Для того щоб завантажити потрібно натиснути на кнопку «Завантажити в PDF»(Рисунок 3.16)

 Завантажити в PDF

Вибір шаблону

Рисунок 3.16 – Кнопка завантаження

## 4 ТЕХНІКО-ЕКОНОМІЧНЕ ОБҐРУНТУВАННЯ

### 4.1 Аналіз ринку

Створення контенту та його подання, справа відповідальна і нелегка. Але до всього потрібно ставитися творчо, особливо, коли мова йде про трансляцію великого обсягу інформації. Чим складніше і різноманітніше те, про що потрібно розповісти, тим більше необхідність у спрощенні.

Увага людини дуже нестійка і не завжди піддається швидкому прочитанню, сприйняттю та усвідомленню, не говорячи вже про наукові та аналітичні дані. Але є методи, що дозволяють спростити процес засвоєння інформації, наприклад візуалізація.

Візуалізація даних – це графічна презентація інформації, завдяки якій можна лаконічно представити в зображенні те, що в текстовому еквіваленті займе не один абзац. Найпоширенішим інструментом для візуалізації даних є інфографіка.

Як показує досвід компаній, які активно її використовують, такими матеріалами набагато частіше діляться в соціальних мережах порівняно з іншими типами контенту. Згідно з даними Google Trends, інтерес до такого формату подачі інформації різко зріс в останні роки, адже інфографіка – ефективний спосіб об'єднати в єдине ціле текст, зображення й елементи дизайну для демонстрації складних даних у форматі послідовної історії.

Розроблена програма є аналогом вже наявних рішень. Розробити та реалізувати абсолютно новий та унікальний продукт на теперішній час є досить складним завданням, тому що сучасне наповнення інформацією в мережі інтернет є дуже урізноманітнене, але кожна є унікальною в своєму виконанні та тематиці.

Інтерактивна інфографіка відкриває багато можливостей взаємодії користувача з програмою у порівнянні зі статичною інфографікою. У ній можна вводити дані по яких здійснювати обрахунки та переглядати отриманий аналіз у вигляді графіків, діаграм, відсотків.

					<i>ДП. КН 21.440.03.000 ПЗ</i>	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		38

Потенційним замовником продукту вважається простий користувач. Основним ринком реалізації продукту буде сайт відділення ВКТ Галицького коледжу. Продукт реалізовано на некомерційній основі і буде знаходитись у вільному доступі абсолютно безкоштовно. У подальшому продукт буде вдосконалюватися та наповнюватися новими елементами.

#### 4.2 Розрахунок витрат на проєктування

Оплата праці – це грошове вираження вартості та ціни робочої сили, яка виступає в ролі заробітку, виплаченого власником підприємства працівнику за виконану роботу.

Розмір заробітної плати залежить від складності та умов виконуваної роботи, кількості виконавців, професійно-ділових якостей працівника, результатів його праці.

Витрати на розробку апаратних та програмних засобів включають:

- заробітну плату розробників;
- відрахування у спеціальні державні фонди;
- накладні витрати;
- інші витрати;
- витрати на використання комп’ютерної техніки.

На стадії розробки проєкту перш за все потрібно визначити працівники якої кваліфікації потрібні для реалізації даного проєктного рішення.

Отже, до розробки даного проєктного рішення були залучені такі спеціалісти, як: проєктувальник, графічний дизайнер, веб-розробник (таблиця 4.1).

Таблиця 4.1 – Розрахунок заробітної плати персоналу

N п/п	Посада виконавця	Оклад грн/міс
1	Проєктувальник	7500



Продовження таблиці 4.1.

2	Графічний дизайнер	8900
3	Веб-розробник	9200
	Всього:	25600

Витрати підприємства на соціальні заходи виникають внаслідок нарахувань сум в залежності від розміру фактичних витрат на оплату праці працівників, які включають витрати на основну і додаткову заробітну плату та здійснення інших видів заохочень та виплат.

Обов'язкові відрахування здійснюються на:

- податок на доходи фізичних осіб;
- військовий збір;
- єдиний внесок.

Проектувальник:

– Рахуємо податок на доходи фізичних осіб:  $7500 \times 18\%$  (ставка податку на доходи фізичних осіб) = 1350 грн.

– Рахуємо військовий збір:  $7500 \times 1,5\%$  (ставка військового збору) = 112,5 грн.

– Рахуємо єдиний внесок:  $7500 \times 22\%$  (ставка ЄСВ) = 1650 грн.

– Утримання становлять – 1462,5 грн. (1350 грн. + 112,5 грн.)

– До виплати працівникові – 6037,5 грн. ( $1 \times (7500 \text{ грн.} - 1462,5 \text{ грн.})$ )

Графічний дизайнер:

1) Рахуємо податок на доходи фізичних осіб:  $8900 \times 18\%$  (ставка податку на доходи фізичних осіб) = 1602 грн.

2) Рахуємо військовий збір:  $8900 \times 1,5\%$  (ставка військового збору) = 133,5 грн.

3) Рахуємо єдиний внесок:  $8900 \times 22\%$  (ставка ЄСВ) = 1958 грн.

4) Утримання – 1735,5 грн. (1602 грн. + 133,5 грн.)

					ДП. КН 21.440.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		40

5) До виплати працівникові – 7164,5 грн. (8900 грн. – 1735,5 грн.)

Веб-розробник:

1) Рахуємо податок на доходи фізичних осіб:  $9200 \times 18\%$  (ставка податку на доходи фізичних осіб) = 1656грн.

2) Рахуємо військовий збір:  $9200 \times 1,5\%$  (ставка військового збору) = 138 грн.

3) Рахуємо єдиний внесок:  $9200 \times 22\%$  (ставка ЄСВ) = 2024 грн.

4) Утримання – 1794грн. (1656 грн. + 138 грн.)

5) До виплати працівникові – 14812 грн. ( $2 \times (9200 \text{ грн.} - 1794 \text{ грн.})$ )

Відрахування на соціальні потреби становлять – 7656 грн.

$((1650+1958+(2024 \times 2))$ (таблиця 4.2)

Таблиця 4.2 – Розрахунок витрат на соціальні відрахування

N п/п	Посада Виконавця	Оклад,гр н/міс	Відрахування грн/міс	Кількість чол. місяців		Сума з/п, грн.
1	Проектувальник	7500	1462,5	1	1	6037,5
2	Графічний дизайнер	8900	1735,5	1	1	7164,5
3	Веб-розробник	9200	1794	1	2	14812
	Всього:	25600	6786	3	4	28014

Загальна сума на інші витрати на матеріальні ресурси визначається у такий спосіб (таблиця 4.3).

Перш за все потрібно визначити які матеріали потрібні на виконання даного завдання, скільки одиниць цього товару потрібно, яка ціна за одиницю товару та обрахувати загальну суму витрат.

Таблиця 4.3. – Розрахунки матеріальних витрат

№ п/п	Найменування ресурсів	Од. виміру	Факт. витрачено матеріалів	Ціна за одиницю, грн.	Сума,грн	Транспортні витрати (10% від суми)	Загальна сума, грн
1	Папір	пачка	1	100	100	2	102
2	Ручка	шт	2	8	16	1,6	17,6

Продовження таблиці 4.3.

3	Олівець	шт	2	10	20	2	22
4	Лінійка	шт	2	15	30	3	33
5	Зошит	шт	1	20	20	2	22
6	Маркери	шт	3	25	75	7,5	82,5
	Всього:						257,1

Витрати на використання комп'ютерної техніки включають витрати на амортизацію комп'ютерної техніки, витрати на користування програмним забезпеченням, витрати на електроенергію, що споживається комп'ютером.

Таблиця 4.4 – Розрахунок витрат на використання комп'ютерної техніки

№ п/п	Назва етапів робіт, при виконанні яких використовується комп'ютер	Час використання комп'ютера, год.	Витрати на використання комп'ютера грн.
1	Проведення досліджень та оформлення їх результатів	100	600
2	Оформлення техніко-економічного розділу	10	60
3	Оформлення ДП	24	144
Всього		134	804

Таблиця 4.5 – Кошторис витрат на проектування

Найменування статей витрат	Сума, грн
1 Зарплата працівників.	25600
2. Обов'язкові відрахування	2414
3. Відрахування на соціальні потреби.	7656
4. Розрахунки матеріальних витрат	257
5. Витрати на використання комп'ютерної техніки.	804
6. Витрати на відлагодження і дослідну експлуатацію програмного продукту	673
7. Накладні витрати експлуатацію	2123
8. Усього, кошторисна вартість проєкту.	37404

#### 4.3 Обґрунтування необхідності розробки

Інфографіка є відмінним доказом вашої компетентності та володіння актуальною інформацією. Вона може використовуватися мультимедіально, неодноразово і варіативно. Майже усім людям запам'ятовується лише 20% прочитаного тексту, а 90% інформації трансформується в візуальні образи. Кольорові зображення підвищують рівень сприйняття даних на 90%.

62 млн. Результатів видавав Google на запит «інфографіка» ще в квітні 2015 року. Інфографіка в 30 більше читана, ніж звичайний текст. 45% компаній використовують інфографіку в маркетингу (більше на 12%, ніж 2 роки тому).

З кожним роком сучасні технології розвиваються. Технологія реалізації даної роботи і безпосередньо сама робота є дуже популярною та розповсюдженою. Дана реалізована продукція сприятиме легкій взаємодії користувача з комп'ютером для заповнення, створення та збереження онлайн резюме.

					<i>ДП. КН 21.440.03.000 ПЗ</i>	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		43

## ВИСНОВКИ

Отже, в наш час Інтернету та гаджетів, Web-сервіс є дуже актуальним, в першу чергу для введення бізнесу і організацій в сучасному суспільстві. Використання онлайн Web-сервісу для того, щоб влаштуватися на роботу значно економить час.

В даній роботі була поставлена ціль: розробити Web-сервіс для створення професійного резюме онлайн. Даний веб-сервіс дозволяє зібрати швидко усю необхідну інформацію про себе, вибрати бажаний шаблон та завантажити готове резюме у pdf-форматі.

В процесі розробки Web-сервісу були виконані наступні завдання:

- вивчення основних понять розробки Web-сервісу;
- розглянуті популярні мови для створення Web-сервісу;
- вивчена вибрана система для створення Web-сервісу для створення онлайн резюме;
- розроблений Web-сервіс для створення професійного резюме онлайн;
- розроблена інструкція по користуванні Web-сервісом для користувача.

Таким чином, можна вважати, що результат розробки відповідає всім потребам технічного завдання, поставлена ціль досягнута, робота несе закономірний характер.

					<i>ДП. КН 21.440.03.000 ПЗ</i>	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		44

## ПРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Скласти резюме онлайн. *Конструктор резюме CV2you*: веб-сайт. URL: <https://cv2you.com/uk-ua/> (дата звернення: 09.02.2021).
2. *MyResume.ua* веб-сайт URL: <https://myresume.ru/> (дата звернення: 18.02.2021).
3. Резюме. *uk.wikipedia*: вебсайт. URL: <https://uk.wikipedia.org/wiki/Резюме> (дата звернення: 27.04.2021).
4. Посібник: *знайомство з React*. Веб-сайт URL : <https://uk.reactjs.org/tutorial/tutorial.html> (дата звернення: 21.05.2021).
5. Why did we build Visual Studio Code? - *Overview. Visual Studio*: вебсайт. URL: <https://code.visualstudio.com/docs/editor/whyvscode> (дата звернення: 11.05.2021).
6. What is HTML?. *w3schools.com*: вебсайт. URL: [https://www.w3schools.com/whatis/whatis\\_html.asp](https://www.w3schools.com/whatis/whatis_html.asp) (дата звернення: 11.05.2021).
7. CSS first steps. *MDN Web docs*: вебсайт. URL: [https://developer.mozilla.org/en-US/docs/Learn/CSS/First\\_steps](https://developer.mozilla.org/en-US/docs/Learn/CSS/First_steps) (дата звернення: 13.05.2021).
8. JavaScript - Overview. *Tutorials point*: вебсайт. URL: [https://www.tutorialspoint.com/javascript/javascript\\_overview.htm](https://www.tutorialspoint.com/javascript/javascript_overview.htm) (дата звернення: 16.05.2021).
9. XML Введение - XML: *Extensible Markup Language | MDN*: вебсайт. URL: [https://developer.mozilla.org/ru/docs/Web/XML/XML\\_introduction](https://developer.mozilla.org/ru/docs/Web/XML/XML_introduction) (дата звернення: 20.05.2021).
10. *What is Software Testing?*. *Guru99*: вебсайт. URL: <https://www.guru99.com/defect-management-process.html> (дата звернення: 23.05.2021).

					ДП. КН 21.440.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		45

ДОДАТКИ

Додаток А

Програмний код

Лістинг А1 – Код функції home

```
class Home extends Component {
  render() {
    return (
      <div className="home-content">
        <Helmet>
          <title>Головна</title>
          <link          rel="icon"          type="image/svg+xml"
href="data:image/svg+xml,<svg  xmlns=%22http://www.w3.org/2000/svg%22
width=%22256%22 height=%22256%22 viewBox=%220 0 100 100%22><text
x=%2250%22    y=%2250%22    dominant-baseline=%22central%22    text-
anchor=%22middle%22 font-size=%2290%22>□</text></svg>" />
        </Helmet>
        <div className="bg-veil"></div>
        <div className="wrapper">
          <h1 className="title">Створіть своє професійне резюме
тут</h1>
          <h3 className="subtitle">Швидко та просто створіть
своє власне професійне резюме та завантажте його у форматі PDF.</h3>
          <Link className="bg-green-500 hover:bg-green-700 text-
white font-bold py-2 px-4 rounded create-btn" to="/resume">Створити
резюме</Link>
        </div>
      </div>
    );
  }
}

export default Home;
```

					ДП. КН 21.440.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		46

## Лістинг А2 – Код функції Nav

```
class Nav extends Component {
  render() {
    return (
      <div className="flex justify-between bg-white mx-auto px-4
py-2 print:hidden navbar">
        <Link className="text-white font-bold py-2 px-2 chev-
left" to="/">На Головні</Link>
        {this.props.showDownloadBtn?<button
          className="z-10 rounded bg-blue text-white m-2 p-2
text-center print:hidden"
          onClick={(e) => {
            e.preventDefault();
            window.print();
          }}
        >
          <svg
            className="w-5 inline-block"
            xmlns="http://www.w3.org/2000/svg"
            fill="none"
            viewBox="0 0 24 24"
            stroke="currentColor"
          >
            <path
              strokeLinecap="round"
              strokeLinejoin="round"
              strokeWidth={2}
              d="M4 16v1a3 3 0 03 3h10a3 3 0 03-3v-1m-4-4l-4
4m0 0l-4-4m4 4v4"
            />
          </svg>{" "}
          Завантажити в PDF
        </button>:""}
      </div>
    );
  }
}

export default Nav;
```

## Лістинг А3 – Код функції Resume

```
class Resume extends React.Component {
  render() {
    return (
      <React.Fragment>
        {this.props.type === "minimalist" ? (
          <Minimalist
            photo={this.props.userPhoto}
            user={this.props.userData}
          />
        ) : null}
      </React.Fragment>
    );
  }
}
```

					ДП. КН 21.440.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		47



### Продовження лістингу А3

```
        employment={this.props.empData}
        empCount={this.props.empCount}
        education={this.props.eduData}
        eduCount={this.props.eduCount}
        project={this.props.projectData}
        projectCount={this.props.projectCount}
        headerColor={this.props.headerColor}
        headerTextColor={this.props.headerTextColor}
      />
    ) : (
      <TwoColumn
        photo={this.props.userPhoto}
        user={this.props.userData}
        employment={this.props.empData}
        empCount={this.props.empCount}
        education={this.props.eduData}
        eduCount={this.props.eduCount}
        project={this.props.projectData}
        projectCount={this.props.projectCount}
        headerColor={this.props.headerColor}
        headerTextColor={this.props.headerTextColor}
      />
    )}
  </React.Fragment>
);
}
}

export default Resume;
```

### Лістинг А4 – Код функції TemplateData

```
class TemplateData extends Component {
  constructor(props) {
    super(props);
    let employment = {
      jobTitles: {},
      emp: {},
      jobDesc: {},
      jobStartDate: {},
      jobEndDate: {},
    };
    let education = {
      qual: {},
      edu: {},
      eduDesc: {},
      eduStartDate: {},
      eduEndDate: {},
    };
    let project = {
```

					ДП. КН 21.440.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		48

## Продовження лістингу А4

```

    projectTitles: {},
    projectDesc: {},
    projectStartDate: {},
    projectEndDate: {},
  };

  this.state = {
    photo: null,
    data: {},
    type: "two-column",
    headerColor: "#7F1D1D",
    headerTextColor: "#ffffff",
    empTemplate: [],
    eduTemplate: [],
    projectTemplate: [],
    empCount: 0,
    eduCount: 0,
    projectCount: 0,
    employment: employment,
    education: education,
    project: project,
  };
  this.handleChange = this.handleChange.bind(this);
  this.handleEmpClick = this.handleEmpClick.bind(this);
  this.handleProjectClick
this.handleChange.bind(this);
  this.handleEduClick = this.handleEduClick.bind(this);
  this.handleAdd = this.handleAdd.bind(this);
}

handleEmpClick(e) {
  e.preventDefault();
  let i = this.state.empCount;
  ++i;
  const template = (
    <div className="w-4/5 p-2 m-2" key={`empKey${i}`}>
      <div className="w-full flex-col justify-center items-
center">
        <label className="p-2" htmlFor="startDate1">
          Почав(-ла)
        </label>
        <input
          className="w-full pl-0.5 m-2 rounded"
          type="month"
          id={`jobStartDate${i}`}
          name={`jobStartDate${i}`}
          onChange={this.handleChange}
        />
        <label className="p-2" htmlFor="endDate1">
          Закінчив(-ла)

```

## Продовження лістингу А4

```

        </label>
        <input
          className="w-full pl-0.5 m-2 rounded"
          type="month"
          id={`jobEndDate${i}`}
          name={`jobEndDate${i}`}
          onChange={this.handleChange}
        />
      </div>
      <input
        className="w-full p-2 m-2 rounded"
        type="text"
        id={`jobTitle${i}`}
        name={`jobTitle${i}`}
        placeholder="Посада"
        onChange={this.handleChange}
      />
      <input
        className="w-full p-2 m-2 rounded"
        type="text"
        id={`emp${i}`}
        name={`emp${i}`}
        placeholder="Роботодавець"
        onChange={this.handleChange}
      />
      <textarea
        className="w-full p-2 m-2 rounded"
        name={`jobDesc${i}`}
        placeholder="Опис"
        rows="5"
        cols="15"
        onChange={this.handleChange}
      />
    </div>
  );

  const empArray = this.state.empTemplate;
  empArray.push(template);
  this.setState({ empTemplate: empArray, empCount: i });
}

handleProjectClick(e) {
  e.preventDefault();
  let i = this.state.projectCount;
  ++i;
  const template = (
    <div className="w-4/5 p-2 m-2" key={`projKey${i}`}>
      <div className="w-full flex-col justify-center items-center">
        <label className="p-2" htmlFor="startDate1">

```

					ДП. КН 21.440.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		50

## Продовження лістингу А4

```

        Почав (-ла)
    </label>
    <input
      className="w-full pl-0.5 m-2 rounded"
      type="month"
      id={`projectStartDate${i}`}
      name={`projectStartDate${i}`}
      onChange={this.handleChange}
    />
    <label className="p-2" htmlFor="endDate1">
      Закічив (-ла)
    </label>
    <input
      className="w-full pl-0.5 m-2 rounded"
      type="month"
      id={`projectEndDate${i}`}
      name={`projectEndDate${i}`}
      onChange={this.handleChange}
    />
  </div>
  <input
    className="w-full p-2 m-2 rounded"
    type="text"
    id={`projectTitle${i}`}
    name={`projectTitle${i}`}
    placeholder="Назва проекту"
    onChange={this.handleChange}
  />
  <textarea
    className="w-full p-2 m-2 rounded"
    name={`projectDesc${i}`}
    placeholder="Опис"
    rows="5"
    cols="15"
    onChange={this.handleChange}
  />
</div>
);

const projectArray = this.state.projectTemplate;
projectArray.push(template);
this.setState({
  projectTemplate: projectArray,
  projectCount: i });
}

handleEduClick(e) {
  e.preventDefault();
  let i = this.state.eduCount;
  ++i;
  const template = (

```

## Продовження лістингу А4

```

<div className="w-4/5 p-2 m-2" key={`eduKey${i}`}>
  <div className="w-full flex-col justify-center items-
center">
    <label className="p-2" htmlFor="startDate1">
      Почав (-ла)
    </label>
    <input
      className="w-full pl-0.5 m-2 rounded"
      type="month"
      id={`eduStartDate${i}`}
      name={`eduStartDate${i}`}
      onChange={this.handleChange}
    />
    <label className="p-2" htmlFor="endDate1">
      Закінчив (-ла)
    </label>
    <input
      className="w-full pl-0.5 m-2 rounded"
      type="month"
      id={`eduEndDate${i}`}
      name={`eduEndDate${i}`}
      onChange={this.handleChange}
    />
  </div>
  <input
    className="w-full p-2 m-2 rounded"
    type="text"
    id={`qual${i}`}
    name={`qual${i}`}
    placeholder="Спеціальність"
    onChange={this.handleChange}
  />
  <input
    className="w-full p-2 m-2 rounded"
    type="text"
    id={`educ${i}`}
    name={`educ${i}`}
    placeholder="Школа/Коледж/Університет"
    onChange={this.handleChange}
  />
  <textarea
    className="w-full p-2 m-2 rounded"
    name={`eduDesc${i}`}
    placeholder="Опис"
    rows="5"
    cols="15"
    onChange={this.handleChange}
  />
</div>
);

```

## Продовження лістингу А4

```

    const eduArray = this.state.eduTemplate;
    eduArray.push(template);
    this.setState({ eduTemplate: eduArray, eduCount: i });
  }

  handleChange(e) {
    const userData = this.state.data;
    const empData = this.state.employment;
    const eduData = this.state.education;
    const projectData = this.state.project;

    if (e.target.name === "templates") {
      this.setState({ type: e.target.value });
      e.target.value === "minimalist"
        ? this.setState({ headerColor: "#F3F4F6",
headerTextColor: "#1F2937" })
        : this.setState({ headerColor: "#7F1D1D",
headerTextColor: "#ffffff" });
    } else if (e.target.name === "headerColor") {
      this.setState({ headerColor: e.target.value });
    } else if (e.target.name === "headerTextColor") {
      this.setState({ headerTextColor: e.target.value });
    } else if (e.target.name.includes("jobTitle")) {
      empData["jobTitles"][`${e.target.name}`] =
e.target.value;
    } else if (e.target.name.includes("emp")) {
      empData["emp"][`${e.target.name}`] = e.target.value;
    } else if (e.target.name.includes("jobDesc")) {
      empData["jobDesc"][`${e.target.name}`] = e.target.value;
    } else if (e.target.name.includes("jobStartDate")) {
      empData["jobStartDate"][`${e.target.name}`] =
e.target.value;
    } else if (e.target.name.includes("jobEndDate")) {
      empData["jobEndDate"][`${e.target.name}`] =
e.target.value;
    } else if (e.target.name.includes("qual")) {
      eduData["qual"][`${e.target.name}`] = e.target.value;
    } else if (e.target.name.includes("educ")) {
      eduData["edu"][`${e.target.name}`] = e.target.value;
    } else if (e.target.name.includes("eduDesc")) {
      eduData["eduDesc"][`${e.target.name}`] = e.target.value;
    } else if (e.target.name.includes("eduStartDate")) {
      eduData["eduStartDate"][`${e.target.name}`] =
e.target.value;
    } else if (e.target.name.includes("eduEndDate")) {
      eduData["eduEndDate"][`${e.target.name}`] =
e.target.value;
    } else if (e.target.name.includes("projectTitle")) {

```

## Продовження лістингу А4

```

        projectData["projectTitles"][`${e.target.name}`] =
e.target.value;
        } else if (e.target.name.includes("projectDesc")) {
        projectData["projectDesc"][`${e.target.name}`] =
e.target.value;
        } else if (e.target.name.includes("projectStartDate")) {
        projectData["projectStartDate"][`${e.target.name}`] =
e.target.value;
        } else if (e.target.name.includes("projectEndDate")) {
        projectData["projectEndDate"][`${e.target.name}`] =
e.target.value;
        } else {
        userData[`${e.target.name}`] = e.target.value;
        }
        this.setState({
        data: userData,
        employment: empData,
        education: eduData,
        project: projectData,
        });
    }

    handleAdd(e) {
        let file = e.target.files[0];
        console.log("e.target.files[0]: ", e.target.files[0]);
        let reader = new FileReader();
        if (file && file.type.match("image.*")) {
            reader.readAsDataURL(file);
        }
        reader.onloadend = () => {
            this.setState({ photo: reader.result });
        };
    }

    render() {
        return (
            <>
            <Helmet>
                <title>Резюме</title>
                <link
                    rel="icon"
                    type="image/svg+xml"
                    href="data:image/svg+xml,<svg
                    xmlns=%22http://www.w3.org/2000/svg%22
                    width=%22256%22
                    height=%22256%22
                    viewBox=%220 0 100 100%22><text
                    x=%2250%22
                    y=%2250%22
                    dominant-baseline=%22central%22
                    text-anchor=%22middle%22
                    font-size=%2290%22>□</text></svg>" />
                </Helmet>
                <Nav showDownloadBtn={true} />
                <div className="container mx-auto px-4">
                <div className="h-full text-gray-dark font-sans">
                <div className="flex p-3 print:p-0">

```

## Продовження лістингу А4

```
<form className="w-2/5 flex flex-col justify-evenly
print:hidden">
  <h3      className="w-4/5      m-2      text-xl">Вибір
шаблону</h3>
  <select
    className="w-4/5 p-2 m-2 rounded"
    name="templates"
    id="templates"
    onChange={this.handleChange}
    value={this.state.type}
  >
    <option value="two-column">Дві колони</option>
    <option
value="minimalist">Мінімалістичний</option>
  </select>
  <div className="w-4/5">
    <label      className="m-2      text-xl      block"
htmlFor="header">
      Копію верхньої частини
    </label>
    <input
      className="mx-2 border-2 border-white rounded"
      type="color"
      id="headerColor"
      name="headerColor"
      onChange={this.handleChange}
      value={this.state.headerColor}
    />
  </div>
  <div className="w-4/5">
    <label      className="m-2      text-xl      block"
htmlFor="header">
      Копію заголовку
    </label>
    <input
      className="mx-2 border-2 border-white rounded"
      type="color"
      id="headerTextColor"
      name="headerTextColor"
      onChange={this.handleChange}
      value={this.state.headerTextColor}
    />
  </div>
  <h3      className="w-4/5      m-2      text-xl">Персональні
деталі</h3>
  <div      className="w-4/5      p-2      m-2      rounded inline-
block"><span      className="inline-block      mr-5">Виберіть
фото</span><input
      type="file"
      onChange={this.handleAdd}
```

					ДП. КН 21.440.03.000 ПЗ	Арк.
						55
Змн.	Арк.	№ докум.	Підпис	Дата		



## Продовження лістингу А4

```

/></div>
<input
  className="w-4/5 p-2 m-2 rounded"
  type="text"
  id="name"
  name="name"
  value={this.state.data.name}
  onChange={this.handleChange}
  placeholder="ПІБ"
/>
<input
  className="w-4/5 p-2 m-2 rounded"
  type="email"
  id="email"
  name="email"
  value={this.state.data.email}
  onChange={this.handleChange}
  placeholder="E-mail"
/>
<input
  className="w-4/5 p-2 m-2 rounded"
  type="tel"
  id="phone"
  name="phone"
  value={this.state.data.phone}
  onChange={this.handleChange}
  placeholder="Телефон"
/>
<input
  className="w-4/5 p-2 m-2 rounded"
  type="text"
  id="location"
  name="location"
  value={this.state.data.location}
  onChange={this.handleChange}
  placeholder="Місто, Країна"
/>
<h3      className="w-4/5      m-2      text-xl">Професійні
деталі</h3>
<input
  className="w-4/5 p-2 m-2 rounded"
  type="text"
  id="wantedJobTitle"
  name="wantedJobTitle"
  value={this.state.data.wantedJobTitle}
  onChange={this.handleChange}
  placeholder="Назва шуканої вакансії"
/>
<textarea
  className="w-4/5 p-2 m-2 rounded"

```

					ДП. КН 21.440.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		56

## Продовження лістингу А4

```

        name="summary"
        placeholder="Професійне резюме"
        rows="5"
        cols="15"
        value={this.state.data.summary}
        onChange={this.handleChange}
      />
      <input
        className="w-4/5 p-2 m-2 rounded"
        type="text"
        id="skills"
        name="skills"
        value={this.state.data.skills}
        onChange={this.handleChange}
        placeholder="Навички"
      />
      <button
        className="w-4/5 rounded bg-blue text-white m-2
p-2 text-center"
        onClick={this.handleEmpClick}
      >
        Додати зайнятість
      </button>
      {this.state.empTemplate.map((el) => el)}
      <button
        className="w-4/5 rounded bg-blue text-white m-2
p-2 text-center"
        onClick={this.handleProjectClick}
      >
        Додати проект
      </button>
      {this.state.projectTemplate.map((el) => el)}
      <h3 className="w-4/5 m-2 text-xl">Освіта</h3>
      <button
        className="w-4/5 rounded bg-blue text-white m-2
p-2 text-center"
        onClick={this.handleEduClick}
      >
        Додати освіту
      </button>
      {this.state.eduTemplate.map((el) => el)}
    </form>
    <Resume
      userPhoto={this.state.photo}
      userData={this.state.data}
      empData={this.state.employment}
      empCount={this.state.empCount}
      eduData={this.state.education}
      eduCount={this.state.eduCount}
      projectData={this.state.project}
    />

```

					ДП. КН 21.440.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		57

```

        projectCount={this.state.projectCount}
        type={this.state.type}
        headerColor={this.state.headerColor}
        headerTextColor={this.state.headerTextColor}
      />
    </div>
  </div>
</div>
</>
);
}
}export default TemplateData;

```

					ДП. КН 21.440.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		58