

Вінницький національний технічний університет
(повне найменування вищого навчального закладу)

Факультет інтелектуальних інформаційних технологій та автоматизації
(повне найменування інституту, назва факультету (відділення))

Кафедра комп'ютерних наук
(повна назва кафедри (предметної, циклової комісії))

МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА

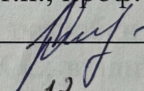
на тему:

«Інформаційна технологія для пошуку працівників»

Виконав: студент 2-го курсу, групи ЗКН-23м
спеціальності 122 «Комп'ютерні науки»
(шифр і назва напрямку підготовки, спеціальності)

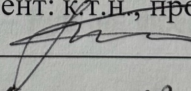

Цимборович Ю. В.
(прізвище та ініціали)

Керівник: к.т.н., проф. каф. КН


Колесницький О. К.
(прізвище та ініціали)

« 05 » 12 2024 р.

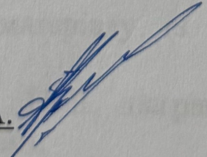
Опонент: к.т.н., професор каф. КСУ


Биков М. М.
(прізвище та ініціали)

« 05 » 12 2024 р.

Допущено до захисту

Завідувач кафедри КН


д.т.н., проф. Яровий А.А.
(прізвище та ініціали)

« 06 » 12 2024 р.

Вінницький національний технічний університет
Факультет інтелектуальних інформаційних технологій та
автоматизації

Кафедра комп'ютерних наук

Рівень вищої освіти II-й (магістерський)

Галузь знань – 12 «Інформаційні технології»

Спеціальність – 122 «Комп'ютерні науки»

Освітньо-професійна програма – «Системи штучного інтелекту»

ЗАТВЕРДЖУЮ

Завідувач кафедри КН

Д.т.н., проф. Яровий А.А.

11.10 2024 року

**ЗАВДАННЯ
НА МАГІСТЕРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ**

Цимборовичу Юрію Вікторовичу

(прізвище, ім'я, по батькові)

1. Тема роботи Інформаційна технологія для пошуку працівників.

керівник роботи к.т.н., проф. кафедри КН Колесницький О. К.

затверджені наказом вищого навчального закладу від "17" 09 2024 року № 310

2. Строк подання студентом роботи 11.11 2024 року

3. Вихідні дані до роботи:

тип програми – онлайн платформа; модель розробки – ітеративна; засоби організації даних програми – база даних MySQL; вхідні дані: інформація про вакансії та працівників; вихідні дані: відображення профілів користувачів, роботи користувачів; кількість функцій – не менше 6; середовище розробки – Visual Studio Code; мова програмування – Python, фреймворк React.

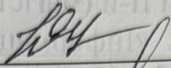
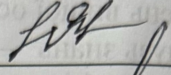
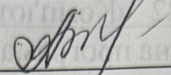
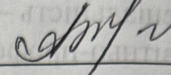
4. Зміст текстової частини:

Вступ, обґрунтування вибору методу розробки та постановка задач дослідження; розробка структури та алгоритмів роботи програмного продукту; розробка програмних компонент онлайн платформи; тестування програми; економічна частина, висновки, перелік використаних джерел, додатки

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

Алгоритм роботи програми, UML діаграма класів, робочі вікна програми, результати тестування програми.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	виконання прийняв
1-4	Колесницький О.К., к.т.н., проф. каф. КН		
5	Адлер О.О. к.т.н., доц. каф. ЕПВМ		

7. Дата видачі завдання 02.09 2024 року

КАЛЕНДАРНИЙ ПЛАН

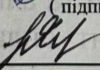
№ з/п	Назва етапів магістерської кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Аналіз сучасного рівня інформаційних технологій для пошуку працівників. Постановка задач дослідження	02.09.24 - 14.09.24	
2	Побудова моделей для пошуку працівників	15.09.24 - 25.09.24	
3	Практичне застосування та оцінка ефективності розроблених моделей	26.09.24 - 02.10.24	
4	Підготовка економічної частини	03.10.24 - 18.10.24	
5	Апробація та/або впровадження результатів дослідження	19.10.24 - 28.10.24	
6	Оформлення пояснювальної записки, графічного матеріалу та презентації	29.10.24 - 04.11.24	
		05.11.24 - 11.11.24	

Студент


(підпис)

Цимборович Ю. В.

Керівник роботи


(підпис)

Колесницький О. К.

АНОТАЦІЯ

УДК 004.8

Цимборович Ю. В. Інформаційна технологія для пошуку працівників. Магістерська кваліфікаційна робота зі спеціальності 122 – «Комп'ютерні науки», освітня програма – «Системи штучного інтелекту». Вінниця: ВНТУ, 2024. 105 с.

На укр. мові. Бібліогр.: 26 назв; рис.: 30; табл. 9.

У даній магістерській кваліфікаційній роботі розглянуто завдання створення інформаційної технології для пошуку працівників сфери мистецтва. В ході аналізу предметної області розглянуто основні методи подачі вакансій для працівників на платформах для пошуку роботи. Обґрунтовано доцільність використання для даної задачі технології React, розроблено структуру й алгоритми роботи програмного забезпечення, а саме удосконалено алгоритм пошуку та сортування працівників відповідно до потреб роботодавця, у якому, на відміну від інших алгоритмів пошуку, враховано специфіку сфери мистецтва та додано характеристики, які відповідають сучасним вимогам роботодавців та працівників, також удосконалено алгоритм пошуку працівників та порівняння необхідних критеріїв відповідно до вакансій та потреб роботодавця. Інформаційну технологію програмно реалізовано у середовищі розробки Visual Studio Code, на мові програмування Python. Розроблена онлайн платформа допомагає спростити процес пошуку та найму працівників у сфері мистецтва.

Графічна частина складається із 6 плакатів.

У економічному розділі доведено доцільність проведення науково-дослідної роботи за даною темою. Термін окупності становить 2,4 р., що свідчить про комерційну привабливість науково-технічної розробки.

Ключові слова: онлайн платформа, пошук працівників, працевлаштування, мистецтво.

ABSTRACT

Tsymborovych Y. V. Information technology for finding employees. Master's thesis in the specialty 122 - «Computer sciences», educational program – «Artificial intelligence systems». Vinnytsia: VNTU, 2024. 105 p.

In Ukrainian language. Bibliogr .: 26 titles; fig . 30; table 9.

This master's qualification work considers the task of creating information technology for searching for workers in the field of art. During the analysis of the subject area, the main methods of submitting vacancies for workers on job search platforms were considered. The feasibility of using React technology for this task was substantiated, the structure and algorithms of the software were developed, namely, the algorithm for searching and sorting workers according to the needs of the employer was improved, which, unlike other search algorithms, took into account the specifics of the field of art and added characteristics that meet the modern requirements of employers and employees, and the algorithm for searching for workers and comparing the necessary criteria according to vacancies and the needs of the employer was also improved. The information technology was programmatically implemented in the Visual Studio Code development environment, in the Python programming language. The developed online platform helps to simplify the process of searching and hiring workers in the field of art..

The graphic part consists of 6 posters.

In the economic section, the expediency of conducting research work on this topic is proven. The payback period is 2.4 years, which indicates the commercial attractiveness of scientific and technical development.

Key words: online platform, employee search, employment, art

ЗМІСТ

ВСТУП.....	4
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ДЛЯ ПОШУКУ ПРАЦІВНИКІВ.....	8
1.1 Аналіз стану платформ для пошуку працівників	8
1.2 Порівняльний аналіз аналогів	10
1.3 Аналіз методів розв’язання задачі	16
1.4 Постановка задач для розробки інформаційної технології пошуку працівників сфери мистецтва	18
1.5 Висновок до розділу 1	19
2 ПРОЕКТУВАННЯ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ДЛЯ ПОШУКУ ПРАЦІВНИКІВ	20
2.1 Аналіз вхідних даних інформаційної технології для пошуку працівників сфери мистецтва.....	20
2.2 Розробка моделі системи	22
2.3 Розробка алгоритмів роботи складових інформаційної технології для пошуку працівників.....	24
2.4 Розробка структури інтерфейсу програмного застосунку	28
2.5 Висновок до розділу 2	35
3 ПРОГРАМНА РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ДЛЯ ПОШУКУ ПРАЦІВНИКІВ.....	36
3.1 Варіантний аналіз і обґрунтування вибору мови програмування	36
3.2 Вибір середовища розробки	38
3.3 Програмна реалізація інформаційної технології для пошуку працівників у вигляді онлайн платформи	41
3.4 Висновок до розділу 3.....	48
4 ТЕСТУВАННЯ ТА АНАЛІЗ РЕЗУЛЬТАТІВ РОБОТИ ОНЛАЙН ПЛАТФОРМИ ДЛЯ ПОШУКУ ПРАЦІВНИКІВ.....	49
4.1 Аналіз методів тестування онлайн платформ.....	49
4.2 Тестування розробленої онлайн платформи.....	51
4.3 Розробка інструкції користувача	55

4.4 Системні вимоги	59
4.5 Аналіз результатів роботи онлайн платформи для пошуку працівників ..	62
4.6 Висновок до розділу 4	63
5 ЕКОНОМІЧНА ЧАСТИНА	64
5.1 Проведення комерційного та технологічного аудиту інформаційної технології для пошуку працівників.	64
5.2 Розрахунок витрат на здійснення науково-дослідної роботи	65
5.3 Розрахунок економічної ефективності науково-технічної розробки інформаційної технології для пошуку працівників за її можливої комерціалізації потенційним інвестором	71
5.4 Висновок до розділу 5	76
ВИСНОВКИ	77
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	80
Додаток А (обов’язковий) ПРОТОКОЛ ПЕРЕВІРКИ КВАЛІФІКАЦІЙНОЇ РОБОТИ НА НАЯВНІСТЬ ТЕКСТОВИХ ЗАПОЗИЧЕНЬ	82
Додаток Б (обов’язковий) Лістинг програми	83
Додаток В (обов’язковий) ІЛЮСТРАТИВНА ЧАСТИНА	99

ВСТУП

Актуальність. У сучасному світі, розвиток інформаційних технологій сприяє змінам у різних сферах життя, включаючи ринок праці. Швидкий розвиток інтернету та зростання популярності онлайн платформ створюють нові можливості для пошуку та найму працівників. Однак, в сфері мистецтва, де талант і креативність є ключовими факторами, необхідно мати спеціалізовану платформу, яка забезпечує зручний та ефективний пошук талановитих працівників.

Актуальність теми дослідження базується на зростаючій потребі у зручних та ефективних інструментах для пошуку талановитих працівників у сфері мистецтва. Сучасний ринок праці в сфері мистецтва вимагає швидкості, точності та зручності при відборі кандидатів для різноманітних творчих проєктів. Традиційні методи пошуку працівників, такі як оголошення в газетах або розміщення вакансій на спеціалізованих дошках оголошень, вже не забезпечують оптимальний вибір працівників, а також не забезпечують ефективного спілкування між роботодавцями та працівниками.

Застосування React технології в розробці програмного забезпечення для пошуку працівників має великий потенціал для поліпшення цього процесу. React є однією з найпопулярніших бібліотек JavaScript, яка дозволяє розробляти ефективні, швидкі та взаємодіючі веб-інтерфейси. Використання React у поєднанні зі зручними інструментами для пошуку та фільтрації працівників може значно полегшити процес пошуку та відбору кандидатів, а також покращити комунікацію між роботодавцями та працівниками.

Таким чином, розробка програмного забезпечення онлайн платформи для пошуку працівників сфери мистецтва з використанням React технології є актуальною та перспективною задачею, яка може значно спростити та покращити процес підбору працівників у цій галузі.

Значимість обраної теми полягає в тому, що розробка програмного забезпечення для пошуку працівників сфери мистецтва має потенціал змінити

підхід до рекрутингу та відбору працівників у цій галузі. Традиційні методи, які використовуються на сьогоднішній день, не завжди ефективні і не забезпечують швидкого та точного відбору кандидатів. Використання новітніх технологій, таких як React, в поєднанні зі зручним та інтуїтивно зрозумілим інтерфейсом, може допомогти прискорити процес пошуку талановитих працівників і забезпечити більш точний відбір відповідно до вимог проекту чи вакансії.

Також, значимість цієї теми проявляється у контексті розвитку ринку праці сфери мистецтва. Тому, розробка програмного забезпечення, яке допоможе залучити та відібрати найкращих кандидатів для мистецьких проектів, є актуальною задачею та важливим кроком у покращенні ефективності цього процесу та задоволенні потреб роботодавців у сфері мистецтва.

Зв'язок роботи з науковими програмами, планами, темами. Робота виконувалася згідно плану виконання наукових досліджень кафедри комп'ютерних наук Вінницького національного технічного університету 22 К1 «Розробка прикладних інтелектуальних інформаційних технологій та систем» та плану наукової навчально-медичної роботи кафедри.

Мета та завдання дослідження. Метою дослідження є розширення функціональних можливостей програмних засобів пошуку працівників сфери мистецтва.

Відповідно до поставленої мети в роботі потрібно вирішити такі **завдання:**

- проаналізувати вимоги користувачів до функціональних можливостей інформаційної технології;
- спроектувати модель інформаційної технології;
- розробити алгоритми пошуку та фільтрації працівників у сфері мистецтва;

- розробити користувацький інтерфейс з використанням React технології;
- інтегрувати платформу з базою даних для зберігання інформації про працівників та роботодавців;
- програмно реалізувати функціонал для зареєстрованих користувачів, включаючи можливість розміщення оголошень про вакансії та відгуки;
- провести тестування розробленого програмного забезпечення;
- розробити інструкцію користувача.

Об’єкт дослідження – процес пошуку працівників сфери мистецтва роботодавцями.

Предмет дослідження – методи та програмні засоби для пошуку працівників сфери мистецтва та їх функціональні можливості.

Методи дослідження. Під час дослідження використовувались наступні методи: порівняльний аналіз функціональних характеристик програм-аналогів для виявлення їх недоліків та постановки задач розробки; теорія алгоритмів для розробки та візуалізації алгоритмів роботи програмного забезпечення; елементи математичної статистики для генерування статистичних даних в онлайн платформі; комп’ютерне моделювання для аналізу та перевірки отриманих теоретичних положень.

Наукова новизна одержаних результатів.

1. Набула подальшого розвитку інформаційна технологія для пошуку працівників, яка відрізняється використанням React технологій та додаванням характеристик, які відповідають сучасним вимогам роботодавців та працівників, що дозволило розширити функціональні можливості програмних засобів для пошуку працівників.

Практичне значення одержаних результатів полягає в тому, що на основі проведених досліджень розроблено програмне забезпечення для пошуку працівників.

Запропонована інформаційна технологія сприяє розширенню функціональних можливостей програмних засобів для пошуку працівників, зокрема:

- розроблено алгоритм роботи програмного забезпечення для пошуку працівників;
- розроблено програмні засоби для пошуку працівників.

Достовірність теоретичних положень магістерської кваліфікаційної роботи підтверджується коректністю постановки завдання, коректністю використання математичного апарату методів дослідження, експериментальними дослідженнями тестування програмної реалізації інформаційної технології для пошуку працівників. Адекватність розроблених математичних моделей підтверджується результатами експериментальних досліджень.

Особистий внесок здобувача. Усі результати, наведені у магістерській кваліфікаційній роботі, отримані самостійно. У працях, написаних у співавторстві, здобувачу належать: аналіз процесу побудови інформаційної технології пошуку працівників [1].

Апробація результатів роботи. Результати роботи були апробовані на Міжнародній науково-практичній Інтернет-конференції студентів, аспірантів та молодих науковців «МОЛОДЬ В НАУЦІ: ДОСЛІДЖЕННЯ, ПРОБЛЕМИ, ПЕРСПЕКТИВИ (МН-2025)», Вінниця, 15 жовтня 2024 року - 15 червня 2025 року».

Публікації. За результатами досліджень опубліковано одні тези доповіді на науково-технічній конференції [1].

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ДЛЯ ПОШУКУ ПРАЦІВНИКІВ

1.1 Аналіз стану платформ для пошуку працівників

На сучасному етапі існує значна кількість платформ для пошуку роботи, орієнтованих на різні галузі та професії [2]. Однак, аналізуючи стан таких платформ у сфері мистецтва, можна виявити певні недоліки та обмеження.

По-перше, багато існуючих платформ недостатньо враховують специфіку сфери мистецтва. Вони часто призначені для загальних професій і не надають необхідних інструментів для презентації творчого потенціалу та розміщення резюме, портфоліо артистів, художників, дизайнерів тощо.

Крім того, існуючі платформи не завжди забезпечують ефективну комунікацію між роботодавцями та працівниками у сфері мистецтва. Вони можуть бути недостатньо інтуїтивними, не надають можливості для обміну повідомленнями, детального обговорення вакансій, та не забезпечують зручності відгуків та оцінювання працівників [3].

Крім того, інтеграція з соціальними мережами, яка має велике значення у сфері мистецтва, не завжди реалізована належним чином на існуючих платформах.

Таким чином, аналізуючи стан платформ для пошуку роботи в сфері мистецтва, стає очевидною потреба у розробці нового програмного забезпечення, яке враховуватиме специфіку цієї галузі та надаватиме зручні інструменти для пошуку талановитих працівників, ефективної взаємодії роботодавців та працівників, а також урахуватиме потреби інтеграції з соціальними мережами. Розробка нової програмної платформи, спеціально призначеної для сфери мистецтва, є необхідною для забезпечення ефективного та зручного пошуку працівників у цій галузі.

Аналізуючи існуючі платформи для пошуку роботи, можна зробити висновок, що вони недостатньо задовольняють потреби роботодавців та працівників у сфері мистецтва. Такі платформи не забезпечують належного рівня інтерактивності, можливості презентації творчих досягнень, а також ефективної комунікації між сторонами.

Для покращення ситуації варто розглянути розробку нових платформ, що враховуватимуть особливості сфери мистецтва та надаватимуть усі необхідні інструменти для ефективного та зручного пошуку роботи. Такі платформи повинні мати зручний інтерфейс, можливість презентації творчих досягнень, а також функціонал для зручної комунікації та обміну інформацією між роботодавцями та працівниками. Також важливо враховувати можливість інтеграції з популярними соціальними мережами, що дозволить розширити кількість можливих контактів та знайти найбільш підходящу вакансію або працівника.

Застосування нової платформи для пошуку роботи в сфері мистецтва має потенціал розширити можливості і поліпшити результативність процесу найму працівників. Одним із ключових аспектів розробки такої платформи є забезпечення спеціалізованих інструментів та функціоналу, які дозволять артистам, художникам, дизайнерам та іншим фахівцям у галузі мистецтва ефективно представити свої таланти та досягнення.

Одним зі способів покращення платформи може бути введення детальних профілів користувачів, де працівники зможуть додавати відео, зображення, аудіо або посилання на свої роботи, що дасть можливість роботодавцям отримати більш повну картину про таланти та стиль кандидатів. Також, варто розглянути впровадження розділу для портфоліо, де працівники зможуть виставляти свої найкращі роботи, що значно полегшить процес оцінки та відбору потенційних працівників.

Додатково, необхідно забезпечити зручний інтерфейс для навігації та взаємодії на платформі, щоб роботодавці могли швидко та легко знаходити необхідних працівників. Функціонал пошуку з можливістю фільтрування за

типом мистецької діяльності, рейтингом, досвідом, локацією та іншими параметрами дозволить знаходити найбільш відповідних кандидатів для конкретних вакансій.

Покращення комунікації між роботодавцями та працівниками також відіграє важливу роль у процесі найму. Реалізація системи повідомлень та можливості проведення віртуальних співбесід покращить ефективність комунікації та сприятиме взаєморозумінню між сторонами. Крім того, важливо передбачити можливість залишення відгуків та оцінок, що допоможе збільшити довіру та об'єктивність у відборі працівників.

Загалом, розробка нової програмної платформи, спеціально призначеної для сфери мистецтва, має на меті вирішити недоліки і обмеження існуючих платформ для пошуку роботи. Це створить сприятливі умови для знаходження талановитих працівників, забезпечить ефективну комунікацію між роботодавцями та працівниками, а також сприятиме інтеграції з соціальними мережами. Розглянутий аналіз стану платформ підтверджує потребу у створенні такої нової платформи, яка враховуватиме унікальні вимоги та потреби учасників мистецького середовища.

1.2 Порівняльний аналіз аналогів

Порівняльний аналіз існуючих аналогів є важливим етапом дослідження перед розробкою програмного забезпечення онлайн платформи для пошуку працівників сфери мистецтва з використанням React [4] технології. Цей аналіз дає можливість оцінити сильні та слабкі сторони існуючих платформ, виявити їхні обмеження та недоліки, а також визначити основні напрямки вдосконалення і розробки нової платформи.

До існуючих аналогів можна віднести наступні платформи: LinkedIn, Behance, ArtStation, Pinterest.

LinkedIn є одним з провідних професійних соціальних мереж і може бути розглянутий як аналог розроблюваної платформи для пошуку

працівників сфери мистецтва (рис. 1.1). LinkedIn забезпечує можливість створення профілів користувачів, де вони можуть представити свої професійні досягнення, навички, освіту та досвід роботи [5].

На LinkedIn роботодавці можуть розміщувати вакансії та шукати кандидатів для певних посад. Користувачі мають можливість спілкуватись, відправляти повідомлення, вступати в професійні групи та мережі для розширення своїх контактів та можливостей.

Однак, LinkedIn має специфічну спрямованість на всі галузі праці, і його функціональність та інтерфейс не завжди повністю відповідають потребам працівників і роботодавців сфери мистецтва. Розроблювана платформа спрямована саме на цю сферу та має враховувати специфіку і потреби її учасників. Вона буде надавати спеціалізовані інструменти для представлення творчих досягнень, зв'язку зі знавцями мистецтва, оцінки навичок та функціонал для ефективного пошуку та найму працівників.

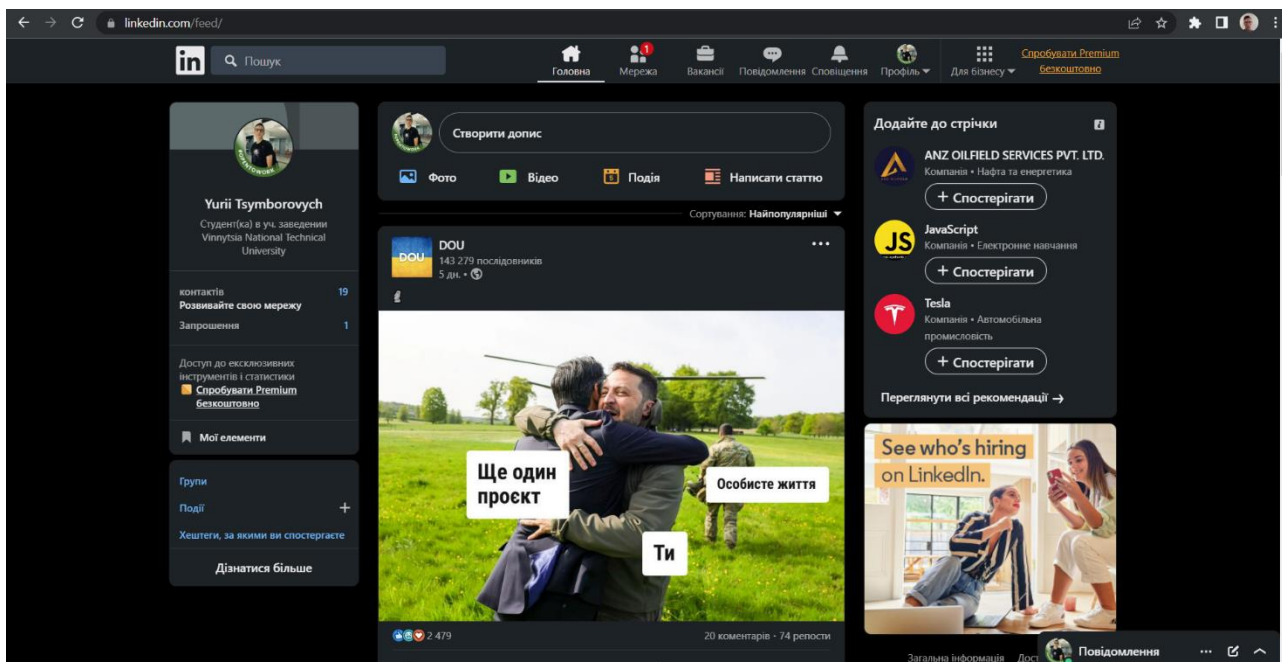


Рисунок 1.1 – Головна сторінка сайту LinkedIn

Behance є платформою для креативних професіоналів, де вони можуть виставляти свої творчі роботи, демонструвати свої навички та отримувати

відгуки від спільноти (рис. 1.2). Цей сервіс дає змогу художникам, дизайнерам, фотографам та іншим творчим людям показати свою роботу та встановити професійні контакти [6].

Behance надає можливість створення профілів, де користувачі можуть додавати свої роботи в галерею, описувати їх та вказувати на свої навички та досвід роботи. Користувачі можуть переглядати роботи інших учасників, залишати коментарі, вподобання та рекомендації.

Однак, Behance сконцентований головним чином на виставленні творчих робіт і побудові профілю художника чи дизайнера, а не на пошуку роботи або найму працівників. Розроблювана платформа буде забезпечувати спеціалізовані інструменти для пошуку працівників сфери мистецтва, включаючи функціонал для розміщення вакансій, пошуку талантів, зв'язку з потенційними кандидатами та оцінки їхнього професійного рівня.

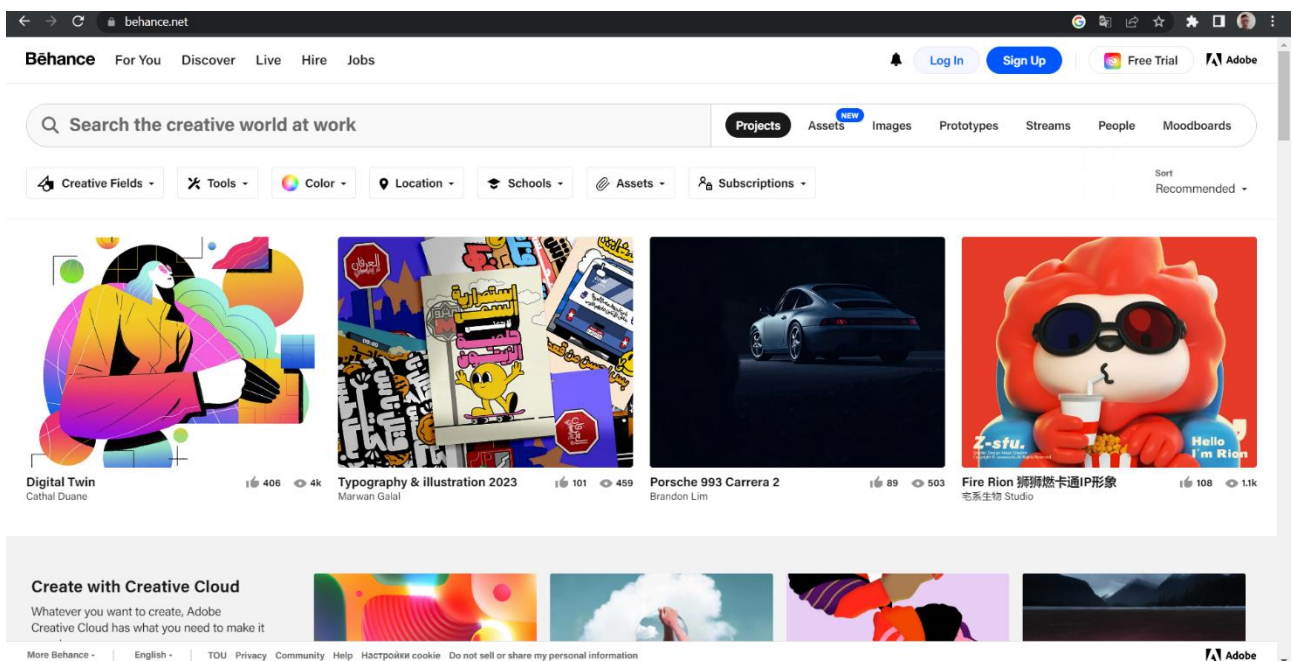


Рисунок 1.2 – Головна сторінка сайту Behance

ArtStation є онлайн-платформою, спрямованою на творчих професіоналів, зокрема художників, дизайнерів і 3D-моделерів, щоб вони могли демонструвати свої творчі роботи, отримувати відгуки та знаходити

професійні можливості (рис. 1.3). Ця платформа надає користувачам можливість створити профіль, де вони можуть показати свої творчі роботи, включаючи малюнки, ілюстрації, концепції, моделі тощо [7].

ArtStation зосереджується на зображеннях високої якості та спеціалізується на художніх роботах та візуальному мистецтві. Вона дає можливість користувачам виставляти свої роботи на показ та продавати їх через магазин, а також спілкуватися з іншими художниками та професіоналами через коментарі та повідомлення.

Однак, ArtStation не спеціалізується на пошуку роботи або наймі працівників. Розроблювана платформа з фокусом на React технології буде пропонувати спеціалізовані інструменти для пошуку та найму працівників у сфері мистецтва. Вона забезпечить функціонал для розміщення вакансій, пошуку талантів, оцінки професійного рівня та зручний спосіб зв'язку з потенційними кандидатами.

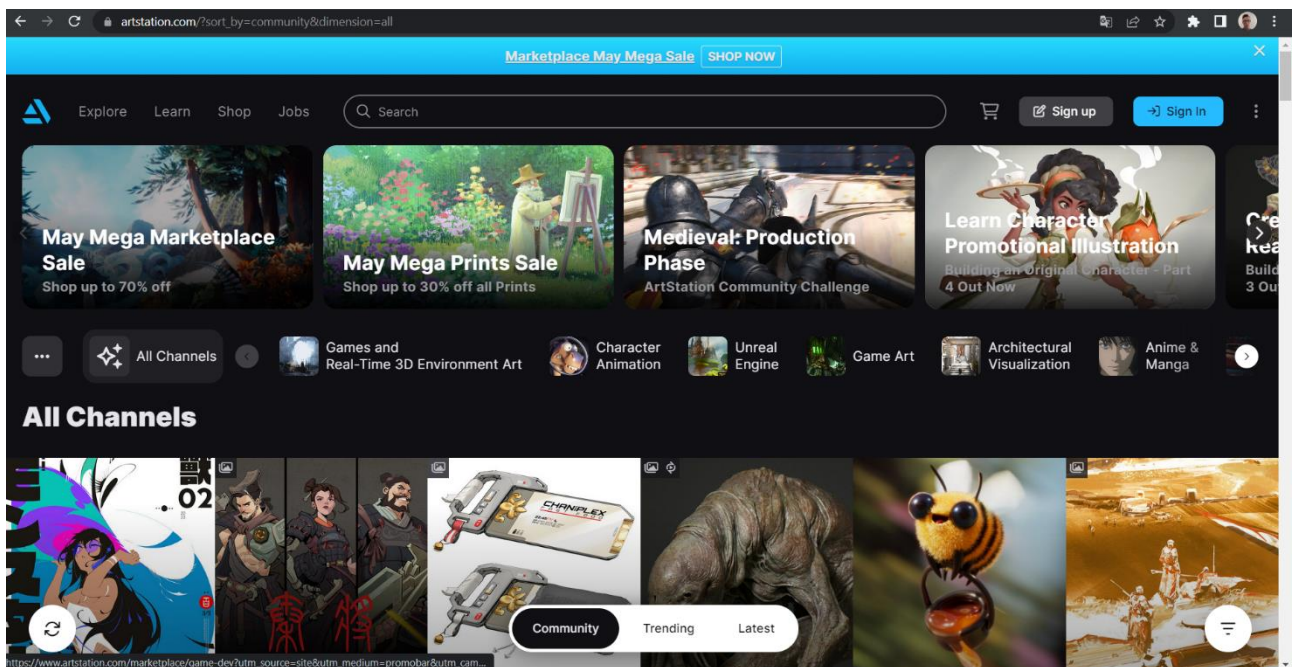


Рисунок 1.3 – Головна сторінка сайту ArtStation

Pinterest – це онлайн-платформа, яка служить як аналог розроблюваної платформи для пошуку роботи в сфері мистецтва (рис. 1.4). Вона пропонує

користувачам можливість зберігати та організовувати візуальні контент-дошки зображень та ідей, відповідно до їх інтересів та потреб.

Pinterest дозволяє користувачам створювати власні профілі, на яких вони можуть створювати та редагувати свої дошки, додавати зображення, відео та посилання з Інтернету. Користувачі також можуть переглядати та підписуватися на профілі інших користувачів, зберігати їх контент у власних колекціях та ділитися ним з іншими.

Аналогічно до розроблюваної платформи, Pinterest надає зручний інтерфейс для пошуку та фільтрування контенту. Користувачі можуть шукати зображення за ключовими словами, категоріями чи тематиками, що дозволяє знаходити відповідний контент для своїх потреб.

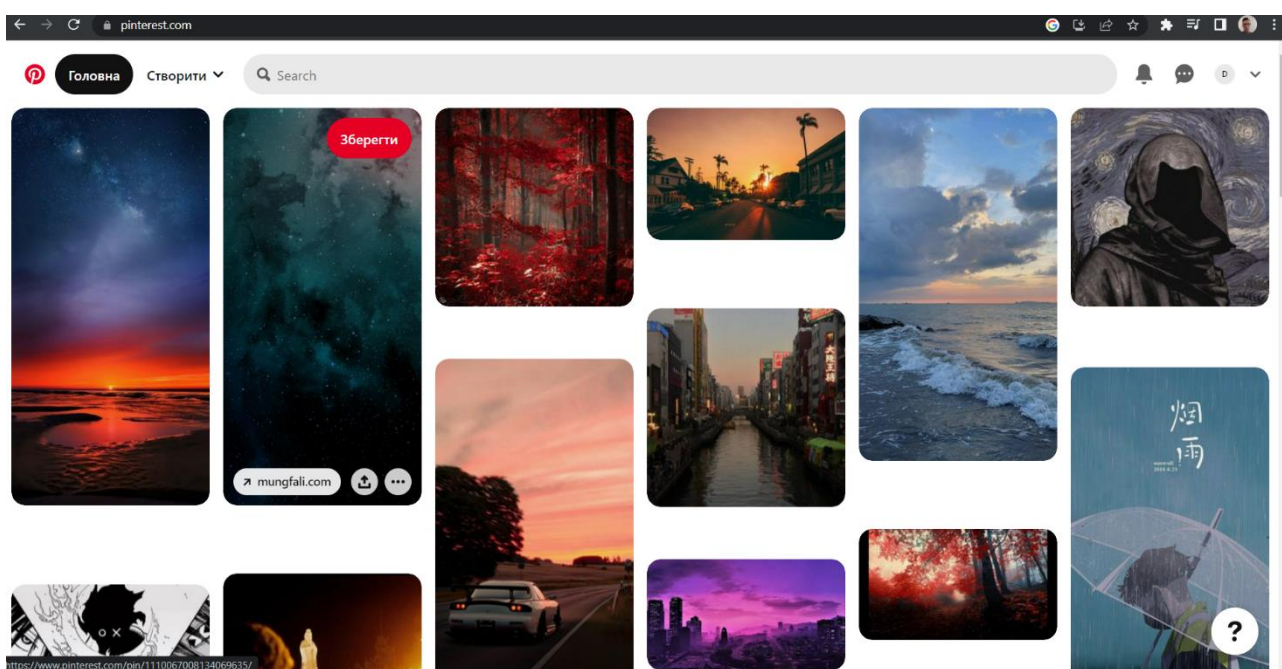


Рисунок 1.4 – Головна сторінка сайту Pinterest

Pinterest також забезпечує можливість взаємодії та спільної роботи між користувачами. Вони можуть коментувати та відзначати сподобання під зображеннями, ділитися своїми враженнями та рекомендаціями. Це створює простір для обміну ідеями та інспірації.

Узагальнюючи, Pinterest є платформою для зберігання, організації та пошуку візуального контенту, що надає користувачам можливість знаходити, зберігати та ділитися ідеями та інспірацією. Це може бути використано як аналог розроблюваної платформи для створення профілів мистецьких працівників, презентації їх творчого потенціалу та обміну ідеями у сфері мистецтва.

В ході порівняльного аналізу було вивчено кілька провідних платформ, що спеціалізуються на пошуку роботи в сфері мистецтва, результат наведено в таблиці 1.1.

Таблиця 1.1 – Порівняльний аналіз аналогів

Критерій	Linkedin	Behance	ArtStation	Pinterest
Функціональність	+	+	-	+
Інтерфейс та використання	-	+	+	+
Взаємодія та комунікація	+	-	+	-
Презентація та візуалізація	+	+	-	+
Налаштування та персоналізація	+	-	+	-

Оцінка проводилась на основі таких критеріїв:

1. Функціональність: аналіз можливостей, які надають платформи для роботодавців та працівників, такі як створення профілів, розміщення вакансій, пошук роботи, комунікація тощо.

2. Інтерфейс та використання: оцінка зручності та інтуїтивності інтерфейсу, легкості навігації та використання платформи.

3. Взаємодія та комунікація: аналіз інструментів, що дозволяють ефективну комунікацію між роботодавцями та працівниками, обмін повідомленнями та уточнення деталей вакансій.

4. Презентація та візуалізація: оцінка можливостей платформи для презентації творчих досягнень, включаючи резюме, портфоліо, зразки робіт та мультимедійний контент.

5. Налаштування та персоналізація: аналіз можливостей налаштування профілів, фільтрації результатів пошуку, сповіщень та інших параметрів.

Порівняльний аналіз аналогів показав, що багато існуючих платформ мають обмежені функціональні можливості або не повністю задовольняють потреби роботодавців та працівників сфери мистецтва. Були виявлені проблеми зі зручністю використання, недостатньою функціональністю, а також обмеженими можливостями для ефективної комунікації та презентації творчих досягнень.

Отже, на основі проведеного порівняльного аналізу стало очевидним, що розробка нової платформи для пошуку працівників сфери мистецтва з використанням React технології є актуальною та перспективною. Вона має потенціал усунути існуючі недоліки та надати користувачам зручний, функціональний та ефективний інструмент для пошуку та найму талантів в цій галузі.

1.3 Аналіз методів розв'язання задачі

У даному розділі буде проведено аналіз методів, які можуть бути використані для розв'язання поставленої задачі. Аналіз методів є важливим етапом в процесі розробки програмного продукту, оскільки він дозволяє оцінити доступні варіанти та обрати найбільш оптимальний підхід для досягнення поставленої мети.

Під час аналізу методів будуть враховані різні аспекти, такі як ефективність, складність реалізації, вимоги до апаратного та програмного забезпечення, можливість масштабування та інші фактори, які впливають на якість та продуктивність програмного продукту.

Основна мета аналізу методів полягає в тому, щоб вибрати такий метод розв'язання задачі, який забезпечить найкращі результати та відповідатиме потребам користувачів. При цьому важливо враховувати особливості самої задачі, а також можливості та обмеження, які виникають під час розробки програмного продукту.

Отриманий аналіз методів буде використаний для визначення стратегії розробки та вибору оптимального шляху розв'язання задачі. Детальні результати аналізу будуть надані у наступних підрозділах даного розділу, що дозволить зробити обґрунтований вибір та розробити ефективний програмний продукт, який задовольнить потреби користувачів та вимоги поставленої задачі.

Для розробки програмного забезпечення онлайн платформи для пошуку працівників сфери мистецтва з використанням React технології, було проведено аналіз різних методів розв'язання задачі, щоб знайти найефективніші та найбільш відповідні підходи. Досліджено наступні методи:

1. Метод веб-розробки з використанням React. Цей метод базується на використанні React технології, яка є однією з найпопулярніших і потужних бібліотек для створення користувацького інтерфейсу. Завдяки своїй модульності та гнучкості, React дозволяє швидко розробляти високоякісні інтерактивні веб-додатки.

2. Метод бази даних. Використання потужних баз даних для зберігання інформації про працівників сфери мистецтва та їхніх професійних навичок [8]. Цей метод дозволяє ефективно організувати та структурувати дані, забезпечуючи швидкий доступ та оптимальну обробку інформації.

3. Метод машинного навчання. Використання алгоритмів машинного навчання для аналізу профілів працівників, рекомендацій та підбору кандидатів [9]. Цей метод дозволяє автоматично виявляти та аналізувати певні характеристики профілів та забезпечувати персоналізовані рекомендації для кожного користувача.

4. Метод впровадження API. Використання сторонніх API для збору та обробки додаткової інформації, такої як дані про мистецтво, вакансії та спеціалізації [10]. Цей метод дозволяє розширити функціональність платформи та забезпечити більш широкий спектр інформації для користувачів.

Отже, було вирішено використати метод веб-розробки з використанням React для створення інтерфейсу, та метод впровадження власної пошукової програми адже вона дозволить надати повноцінний програмний продукт без необхідності в потужному апаратному забезпеченні.

Крім того, для реалізації функціональності пошукової програми було вирішено використати власне рішення, що дозволяє забезпечити повноцінну роботу програмного продукту без необхідності в потужному апаратному забезпеченні. Це дозволяє зберегти ресурси та забезпечити ефективне функціонування програми на різних платформах та пристроях.

1.4 Постановка задач для розробки інформаційної технології пошуку працівників сфери мистецтва

Для успішної розробки інформаційної технології та програмного забезпечення онлайн платформи для пошуку працівників сфери мистецтва з використанням React технології, були сформульовані наступні задачі:

- проаналізувати вимоги користувачів до функціональних можливостей платформи;
- спроектувати архітектуру платформи;
- розробити алгоритми пошуку та фільтрації працівників у сфері мистецтва;
- розробити користувацький інтерфейс з використанням React технології;

- інтегрувати платформу з базою даних для зберігання інформації про працівників та роботодавців;
- програмно реалізувати функціонал для зареєстрованих користувачів, включаючи можливість розміщення оголошень про вакансії та відгуки;
- провести тестування розробленого програмного забезпечення;
- розробити інструкцію користувача.

Постановка цих задач визначає напрямок роботи над розробкою програми пошуку працівників сфери мистецтва та встановлює конкретні цілі, які необхідно досягти у процесі розробки.

1.5 Висновок до розділу 1

У результаті аналізу стану платформ для пошуку роботи, порівняльного аналізу аналогів, аналізу методів розв'язання задачі та постановки задач для розробки інформаційної технології пошуку працівників сфери мистецтва, були з'ясовано, що в даний час існує певна кількість платформ для пошуку роботи в сфері мистецтва, таких як LinkedIn, Behance, ArtStation та Pinterest. Вони надають можливості для презентації творчої роботи та знаходження робочих можливостей, проте кожна з них має свої переваги та обмеження. Аналіз методів розв'язання задачі показав, що використання React технології дозволить створити ефективний та зручний інтерфейс користувача, забезпечити масштабованість та високу продуктивність системи. Поставлені задачі для розробки програми пошуку працівників сфери мистецтва включають розробку системи авторизації та профілю користувача, системи пошуку працівників, функціоналу оцінки та відгуків, системи спілкування та обміну повідомленнями, а також розробку інтерфейсу користувача.

2 ПРОЕКТУВАННЯ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ДЛЯ ПОШУКУ ПРАЦІВНИКІВ

2.1 Аналіз вхідних даних інформаційної технології для пошуку працівників сфери мистецтва

У даному пункті проводиться аналіз даних, який має на меті отримання розуміння структури та характеристик даних, що будуть використовуватись у програмному продукті. Аналіз даних є важливим етапом в процесі розробки, оскільки він допомагає визначити необхідні дані, їх взаємозв'язки та основні параметри, необхідні для ефективної роботи програмного додатку.

В першу чергу, здійснюється збір та аналіз вхідних даних, які будуть використовуватись для пошуку працівників сфери мистецтва. Це можуть бути дані про роботодавців, їх вакансії, професійні навички та освіту працівників, відгуки та оцінки користувачів тощо. Аналізу підлягають як структуровані дані, так і неструктуровані, які можуть бути отримані з різних джерел, таких як бази даних, API, веб-сторінки, соціальні мережі та інші джерела.

Далі, проводиться обробка та узагальнення отриманих даних з метою виявлення ключових параметрів та характеристик, які впливатимуть на ефективність пошуку та знаходження працівників сфери мистецтва. Це можуть бути такі параметри, як професійні навички, досвід роботи, географічне розташування, категорії та типи робітників тощо. Також проводиться аналіз потреб користувачів, їх вимог та очікувань щодо пошуку працівників.

Загальний аналіз даних допомагає визначити ключові параметри, які будуть враховані при розробці структури та алгоритмів програмного продукту. Він також допомагає зрозуміти особливості та вимоги сфери мистецтва, що дозволяє налаштувати програмний продукт для досягнення максимальної ефективності та задоволення потреб користувачів.

В процесі аналізу даних було враховано різноманітні фактори, такі як кількість доступних вакансій, типи робочих місць, навички та вимоги роботодавців, а також профілі користувачів і їхні переваги. Використовуючи статистичні методи, було виявлено популярні напрямки пошуку роботи у сфері мистецтва та з'ясовано, які критерії є важливими для роботодавців при відборі кандидатів.

Результати аналізу даних вказують на потребу в ефективній системі пошуку роботи та працівників у сфері мистецтва. Основні висновки включають важливість врахування професійних навичок, портфоліо та відгуків, а також можливість фільтрування результатів пошуку за різними критеріями. Дані висновки служать основою для розробки алгоритмів, які забезпечать точні й релевантні результати пошуку роботи та кваліфікованих працівників.

Дані з аналізу допоможуть у визначенні стратегії розвитку та покращення розробленої онлайн платформи. Зрозуміння потреб користувачів і тенденцій у сфері мистецтва дозволить нам пристосувати функціонал платформи для задоволення їхніх вимог та покращення їхнього досвіду користування. У наступних розділах будуть більш детально розглянуті результати аналізу даних і їхнє вплив на подальшу розробку системи.

Крім того, аналіз даних виявив потребу у зручному та інтуїтивно зрозумілому інтерфейсі для користувачів, а також важливість забезпечення безпеки та конфіденційності даних. З цільовою аудиторією було проведено опитування та співбесіди, що дозволило зібрати важливі відгуки та пропозиції для поліпшення платформи.

Загальний аналіз даних став основою для розробки функціональних вимог та особливостей платформи, а також визначення пріоритетних завдань для подальшої реалізації. Отже, розробка алгоритмів та функцій онлайн платформи базується на уважному аналізі даних, що дозволяє забезпечити ефективність, точність та задоволення потреб користувачів у пошуку роботи або працівників у сфері мистецтва.

У наступних розділах будуть детально розглянуті результати аналізу даних, що стануть основою для подальшої розробки та реалізації інформаційної технології. Зазначений аналіз є важливим кроком у процесі створення розробленої онлайн платформи, оскільки допомагає забезпечити високу якість та відповідність продукту потребам користувачів.

Отже, аналіз даних є важливим кроком у розробці програмного продукту для пошуку працівників сфери мистецтва, оскільки він надає необхідну інформацію для подальшого проектування структури та алгоритмів системи.

2.2 Розробка моделі системи

В даному пункті проведена розробка моделі системи для нашої онлайн платформи. Модель системи включає в себе опис структури та функціональності системи, її компонентів та взаємозв'язків між ними. Під час розробки моделі системи було враховано всі основні вимоги та функціональні можливості, які необхідні для ефективної роботи платформи.

Модель системи включає опис основних модулів та їх функціональності. Один з ключових модулів – модуль пошуку працівників, який забезпечує користувачам можливість знаходити відповідних працівників зі сфери мистецтва за допомогою різних критеріїв. Інший важливий модуль - модуль управління вакансіями, який дозволяє розміщувати нові вакансії та керувати ними. Крім того, в моделі системи передбачено модуль авторизації та реєстрації користувачів, що забезпечує безпечний доступ до платформи та збереження персональних даних користувачів. Також в моделі системи визначені модулі для рейтингування та відгуків, які дозволяють користувачам оцінювати та залишати відгуки про працівників. Для кращого розуміння роботи онлайн платформи для пошуку роботи було розроблено модель роботи системи на прикладі UML [11] діаграми діяльності (рис. 2.1).

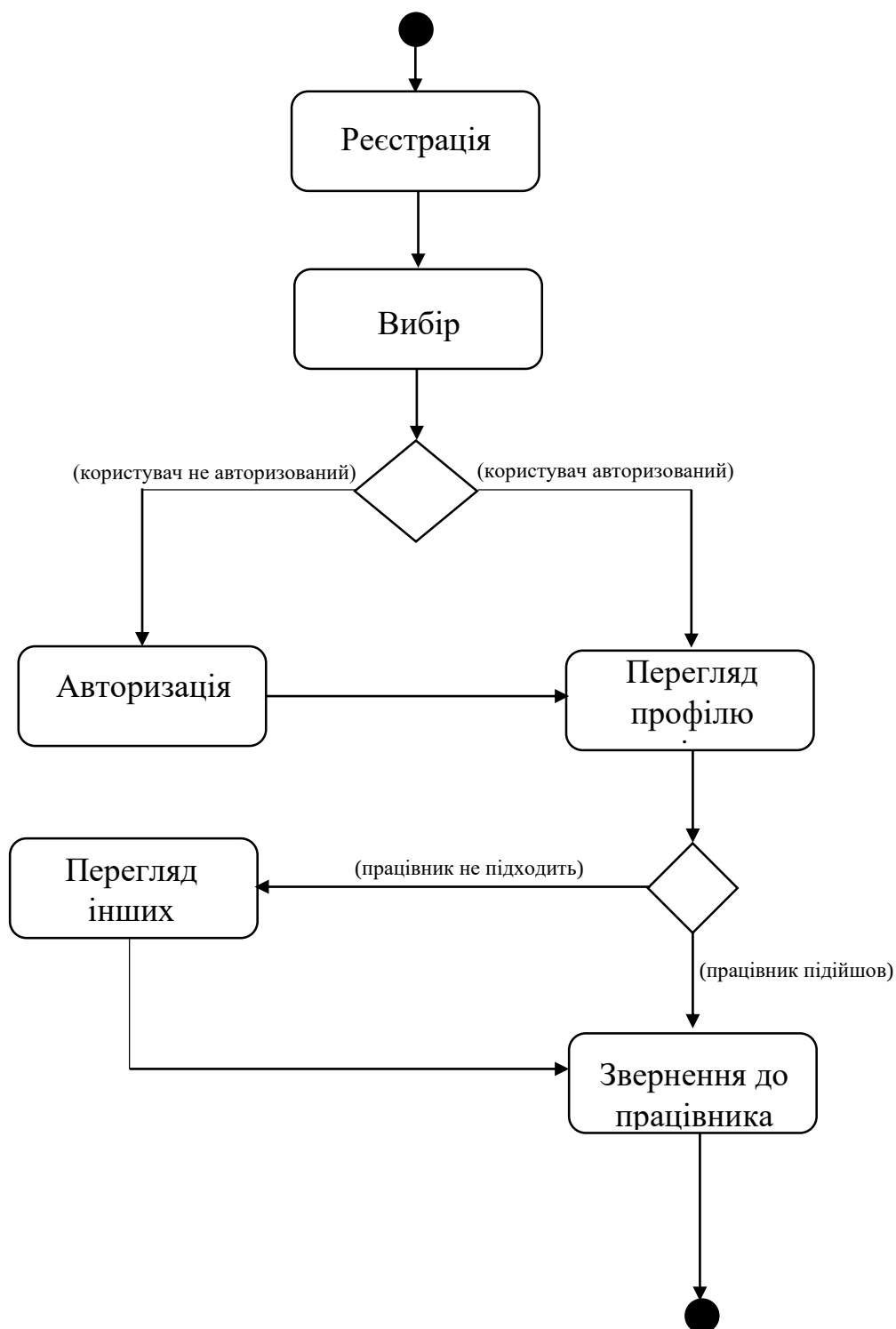


Рисунок 2.1 – Діаграма діяльності інформаційної технології для пошуку працівників

Згідно вказаної діаграми найперше користувачу необхідно зареєструватись, після чого обрати працівника який йому підходить для найму. Якщо користувач авторизований профіль працівника стає доступний

для перегляду та звернення до працівника з пропозицією працевлаштування, якщо ж ні – користувачу пропонується авторизуватись. У випадку якщо працівник сподобався користувачу – він звертається до нього завдяки контактам вказаним працівником в його профілі, в іншому випадку користувач переглядає інші доступні профілі працівників.

У результаті розробки моделі системи було створено загальну структуру програмного додатку, яка відповідає вимогам та функціональності онлайн платформи. Модель системи є основою для подальшої розробки та реалізації програмного додатку, дозволяючи забезпечити його ефективну та надійну роботу.

2.3 Розробка алгоритмів роботи складових інформаційної технології для пошуку працівників

У даному пункті розглядається процес розробки алгоритмів, що використовуються в інформаційній технології для пошуку працівників. Алгоритми відіграють важливу роль у забезпеченні ефективного та точного пошуку кандидатів для роботи. Основною метою розробки алгоритмів є створення системи, яка здатна швидко та ефективно обробляти великі обсяги даних і надавати релевантні результати користувачам.

При розробці алгоритмів роботи пошукової системи враховуються різноманітні фактори, такі як критерії відбору, ранжування та сортування результатів, а також врахування важливих параметрів, що визначають якість кандидатів. Комплексний підхід до розробки алгоритмів дозволяє досягти оптимального результату, забезпечуючи точність і швидкість пошуку працівників, що відповідають вимогам роботодавців.

У цьому розділі будуть детально розглянуті розроблені алгоритми роботи інформаційної технології для пошуку працівників, їх принципи функціонування та переваги, які вони надають в контексті відбору кандидатів для роботи.

Для розробки інформаційної технології для пошуку працівників сфери мистецтва необхідно розробити загальний алгоритм, згідно якого буде працювати система (рис. 2.2).

Після ретельного аналізу цього алгоритму стає очевидним, що запропонований підхід до відбору кандидатів дозволить роботодавцєві автоматизувати процес початкової перевірки їх знань та уникнути вибору користувачів з низьким рівнем кваліфікації. Більш того, цей алгоритм забезпечить можливість працівнику отримати роботу, що відповідає його вмінням та навичкам.

Розглянемо всі кроки алгоритму:

Крок 1. Початок

Крок 2. Користувач реєструється та заповнює всі необхідні дані про вакансію згідно яких хоче найняти собі працівника.

Крок 3. Користувач вказує пріоритетну характеристику вакансії якщо така необхідна.

Крок 4. Виповнюється порівняння критеріїв вакансії з даними вказаними працівниками про свої навички та вміння при реєстрації. Враховується пріоритетна характеристика якщо вона була вказана.

Крок 5. Виводиться відсортований список профілів працівників.

Крок 6. Користувач обирає та переглядає виведені профілі працівників та зв'язується з ними пропонуючи роботу.

Крок 7. Кінець.

Також, з метою забезпечення зручності та задоволення потреб користувачів, було прийнято рішення розробити алгоритм пошуку працівників для незареєстрованих користувачів. Цей алгоритм (рис. 2.3) базується на чотирьох основних критеріях, які дозволяють користувачам здійснювати точний та узгоджений пошук працівників.

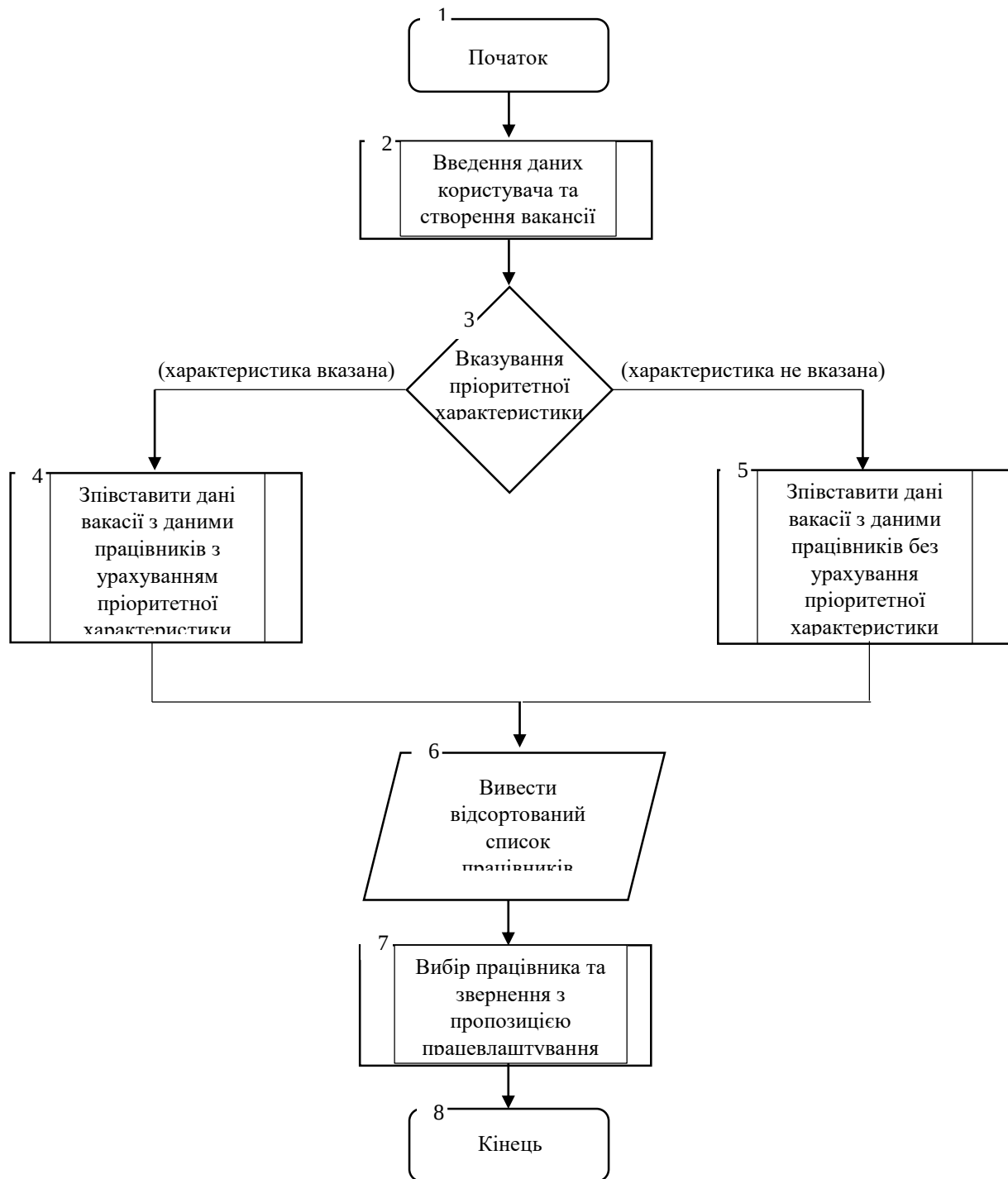


Рисунок 2.2 – Блок-схема алгоритму пошуку працівників за вакансією

Перший критерій – місцезнаходження – дозволяє користувачеві вказати бажане місцезнаходження працівника за допомогою відповідного фільтру. Алгоритм враховує цей критерій та здійснює пошук працівників, які знаходяться в заданому районі або місті.

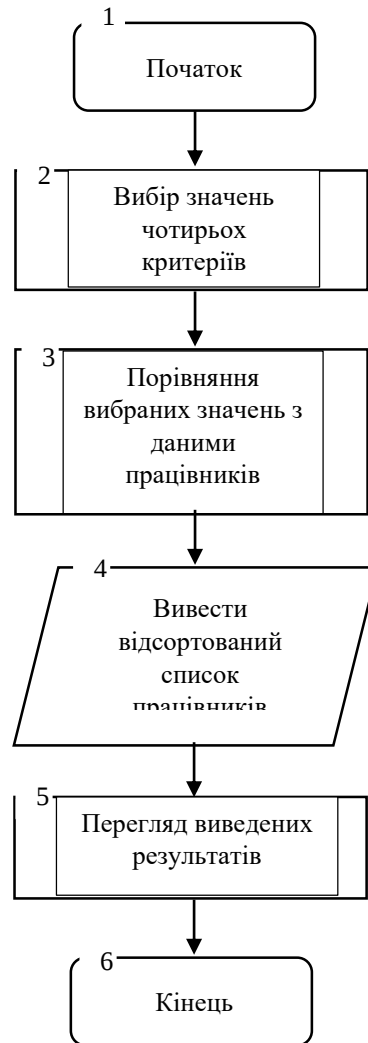


Рисунок 2.3 – Блок-схема алгоритму пошуку працівників по 4 критеріям

Другий критерій – вид діяльності – дозволяє користувачеві обрати певну сферу мистецтва або іншу галузь, для якої потрібен працівник. Алгоритм враховує цей критерій та шукає працівників, які мають відповідні навички та досвід у вибраній сфері.

Третій критерій – досвід – дозволяє користувачеві вказати бажаний рівень досвіду працівника, наприклад, початківець, середній або досвідчений. Алгоритм враховує цей критерій та шукає працівників, які мають відповідний рівень досвіду.

Четвертий критерій – рейтинг – дозволяє користувачеві враховувати рейтинг працівників при пошуку. Алгоритм здійснює сортування працівників

за їхнім рейтингом, щоб користувач міг знайти найкращих працівників у вибраних критеріях.

Цей алгоритм пошуку був створений з метою надання можливості людям, які не бажають створювати вакансію або просто хочуть ознайомитись з тенденціями ринку, знайти відповідних працівників за заданими критеріями. Його використання спрощує та прискорює процес пошуку та взаємодії з потенційними працівниками.

Розглянемо всі кроки алгоритму:

Крок 1. Початок

Крок 2. Користувач вибирає значення кожного з чотирьох основних критеріїв.

Крок 3. Виповнюється порівняння обраних критеріїв з даними вказаними працівниками про свої навички та вміння при реєстрації.

Крок 4. Виводиться відсортований список профілів працівників для вибору користувачем.

Крок 6. Користувач обирає та переглядає виведені профілі ознайомлюючись з тенденціями ринку.

Крок 7. Кінець.

Отже, було розроблено алгоритми пошуку працівників в сфері мистецтва з урахуванням критеріїв вказаних в вакансії. Також було удосконалено алгоритм пошуку працівників та порівняння необхідних критеріїв відповідно до вакансій та потреб роботодавця за допомогою вказування пріоритетної характеристики.

2.4 Розробка структури інтерфейсу програмного застосунку

Розробка ефективного та логічно структурованого інтерфейсу програмного додатку є важливим етапом процесу розробки. У цьому пункті описується процес розробки структури інтерфейсу нашого програмного

додатку, який дозволить користувачам зручно та ефективно взаємодіяти з програмою.

Під час розробки структури інтерфейсу було приділено особливу увагу зрозумілості та інтуїтивності. Починаючи з аналізу вимог та потреб користувачів, були визначені основні функціональні блоки та їх взаємозв'язки. Структура інтерфейсу була побудована таким чином, щоб забезпечити логічний потік роботи [12] та зручну навігацію для користувача.

Одним з ключових аспектів розробки структури інтерфейсу було використання візуальних елементів, таких як меню, панелі інструментів та кнопки, для організації функціональних модулів. Це дозволяє користувачам швидко зорієнтуватись у програмі та доступатись до потрібних функцій.

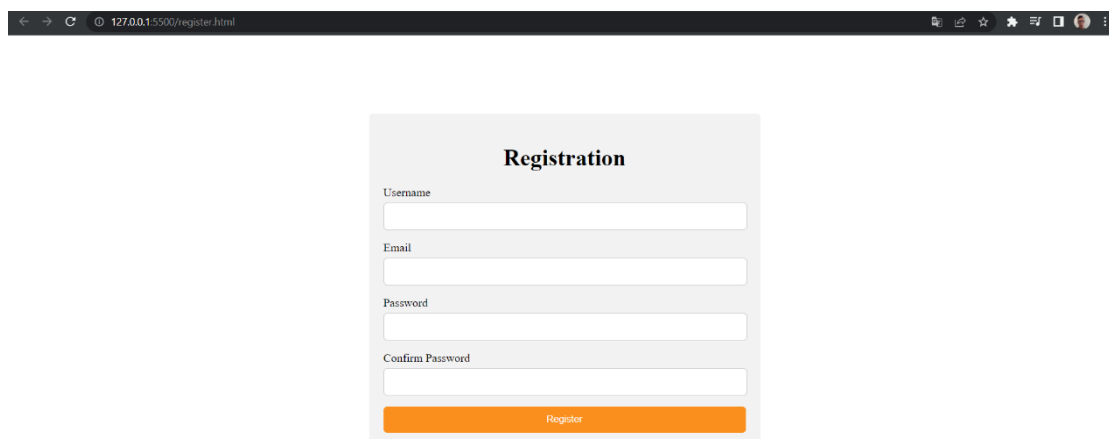
Крім того, важливим аспектом було врахування принципів дизайну інтерфейсу, таких як чіткість, консистентність та естетичний вигляд. Розробленої структури інтерфейсу приділено достатню увагу з погляду розміщення елементів, використання кольорів та шрифтів, щоб створити приємний та привабливий зовнішній вигляд програми.

В процесі розробки інтерфейсу онлайн платформи було прийнято рішення зробити його легким у використанні та зрозумілим для користувачів. Для полегшення навігації було включено головне меню (рис. 2.4), де користувач має можливість обрати бажану сторінку для перегляду.



Рисунок 2.4 – Головне меню

Використання онлайн платформи передбачає обов'язкову реєстрацію користувачів у системі, тому було розроблено окрему сторінку для реєстрації користувача (рис. 2.5). На цій сторінці користувач зобов'язаний ввести свої особисті дані та обрати пароль. Після успішної реєстрації, користувач автоматично авторизується в системі.



The image shows a web browser window with the address bar displaying '127.0.0.1:5500/register.html'. The main content is a registration form with the title 'Registration' in bold. Below the title are four input fields labeled 'Username', 'Email', 'Password', and 'Confirm Password'. At the bottom of the form is an orange button labeled 'Register'.

Рисунок 2.5 – Сторінка реєстрації користувача

На початковій сторінці (рис. 2.6) доступні різні фільтри для пошуку працівників, які є корисним інструментом для тих користувачів, які не бажають створювати конкретну вакансію. Це особливо корисно для незареєстрованих користувачів та працівників, які бажають створити профіль, що відповідає сучасним тенденціям на ринку праці.

Фільтри на початковій сторінці дозволяють користувачеві встановлювати певні параметри пошуку, такі як місцезнаходження, вид діяльності, рівень досвіду та рейтинг працівника. Користувач може вказати бажане місцезнаходження, вибрати конкретну сферу діяльності, вказати необхідний рівень досвіду та враховувати рейтинг працівників під час пошуку. Ці фільтри допомагають звузити результати пошуку та знайти працівників, що відповідають заданим критеріям.

Для незареєстрованих користувачів ці фільтри є особливо важливими, оскільки вони можуть використовувати їх для ознайомлення з потенційними працівниками та їхніми характеристиками без необхідності створення конкретної вакансії. Також це є зручним інструментом для працівників, які

бажають створити свій профіль та привернути увагу роботодавців, відповідаючи сучасним тенденціям на ринку праці.

Завдяки доступним фільтрам на початковій сторінці, користувачі можуть здійснювати зручний та спрощений пошук працівників, а також створювати профілі, які відповідають їхнім потребам та очікуванням. Цей інструмент допомагає підвищити ефективність та швидкість процесу пошуку працівників, сприяючи успішній взаємодії між роботодавцями та працівниками.

Jobfinder for artists

Monetize your hobby

Location	Activity
Experience	Rating
Find	



Рисунок 2.6 – Головна сторінка онлайн платформи для пошуку працівників

Наступна сторінка створена (рис. 2.7) для інформування користувачів про даний проєкт та його цілі а також містить в собі інструкцію користувача і контактні дані як розробників так і власників онлайн платформи для зв'язку у випадку виникнення питань або зауважень.



A new way to find a job

For decades, people in the arts have been looking for the most promising and high-paying jobs, but there hasn't been a platform aimed exclusively at this field, but now they can always turn to us in search of the perfect job.

[Learn more](#)

Рисунок 2.7 – Інформаційна сторінка

На наступній сторінці (рис. 2.8) можна побачити популярні напрямки пошуку працівників з різних сфер мистецтва. Ця інформація надає користувачам можливість ознайомитись з тенденціями на ринку праці та популярними роботами працівників з найвищим рейтингом на онлайн платформі.

Ці популярні напрямки допомагають користувачам швидко зорієнтуватись у сферах мистецтва, де попит на працівників є високим, а також дізнатись про популярні роботи та досягнення працівників з високим рейтингом. Це корисна інформація для тих, хто бажає отримати огляд поточних тенденцій і знати, які сфери мистецтва є особливо вимогливими та привабливими для роботодавців та аудиторії.

Враховуючи ці напрямки пошуку, користувачі можуть швидко знайти працівників з високою популярністю та рейтингом у вибраній сфері мистецтва, що сприяє покращенню ефективності пошуку та вибору кандидатів. Крім того, ця інформація може стати корисною для саморозвитку працівників, які бажають орієнтуватись у вимогах ринку праці та навчитись виконувати популярні та високооцінені роботи у своїй сфері мистецтва.

Крім того, популярні напрямки пошуку працівників з різних сфер мистецтва також можуть надати інсайти щодо змін у вимогах ринку праці та нових тенденцій у мистецькій сфері. Це може бути корисно для роботодавців, які бажають відстежувати розвиток сфери та адаптувати свої вимоги до поточних потреб. Крім того, ці дані можуть стати основою для подальших досліджень та аналізу ринку праці в сфері мистецтва, сприяючи розумінню його динаміки та прогнозуванню майбутніх змін. Таким чином, популярні напрямки пошуку працівників є цінним ресурсом, який допомагає створити зв'язок між роботодавцями та працівниками, сприяючи ефективнішому та успішнішому залученню кваліфікованих фахівців у сфері мистецтва.

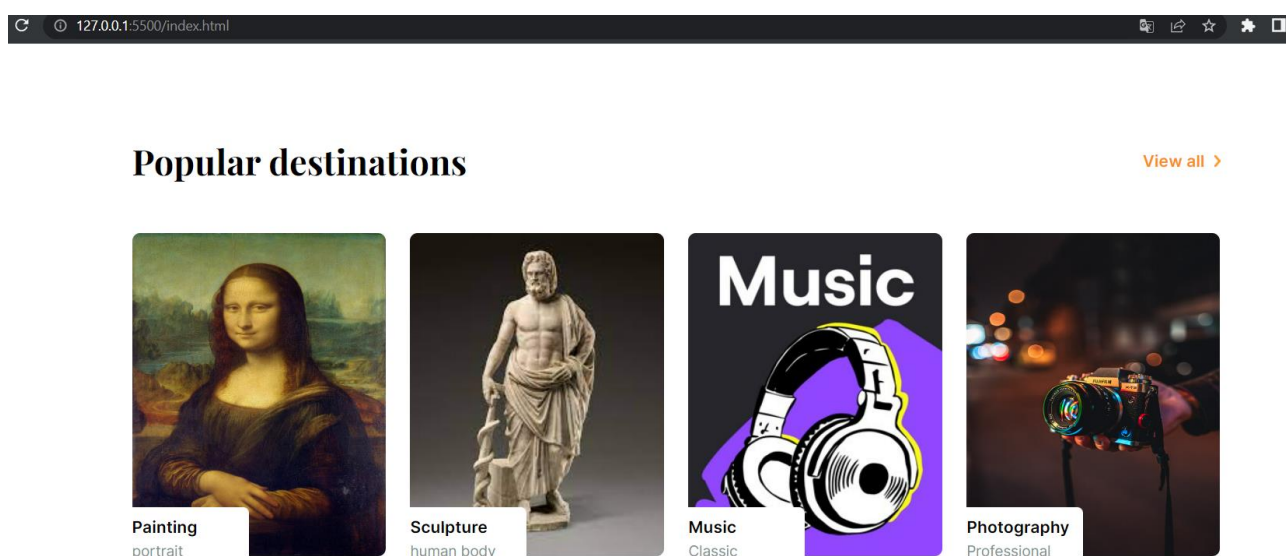


Рисунок 2.8 – Сторінка «Destinations»

Ще одна корисна сторінка для перегляду – це сторінка з відгуками про найбільш популярних працівників онлайн платформи (рис. 2.9) та їхній рейтинг.

На сторінці з відгуками про найбільш популярних працівників онлайн платформи користувачі мають можливість ознайомитись з думками та враженнями інших роботодавців, які вже співпрацювали з цими працівниками. Відгуки можуть містити інформацію про якість роботи,

професіоналізм, комунікативність та інші аспекти, що є важливими при виборі працівника. Крім того, наявність високого рейтингу таких працівників підтверджує їхню успішність та популярність серед роботодавців. Завдяки цій сторінці, користувачі можуть отримати більш об'єктивну оцінку працівників і прийняти більш обгрунтоване рішення при виборі потрібного фахівця для своїх проектів. Відгуки є важливим елементом взаємодії між користувачами платформи, допомагаючи побудувати довіру та забезпечити якість виконання робіт в мистецькій сфері.

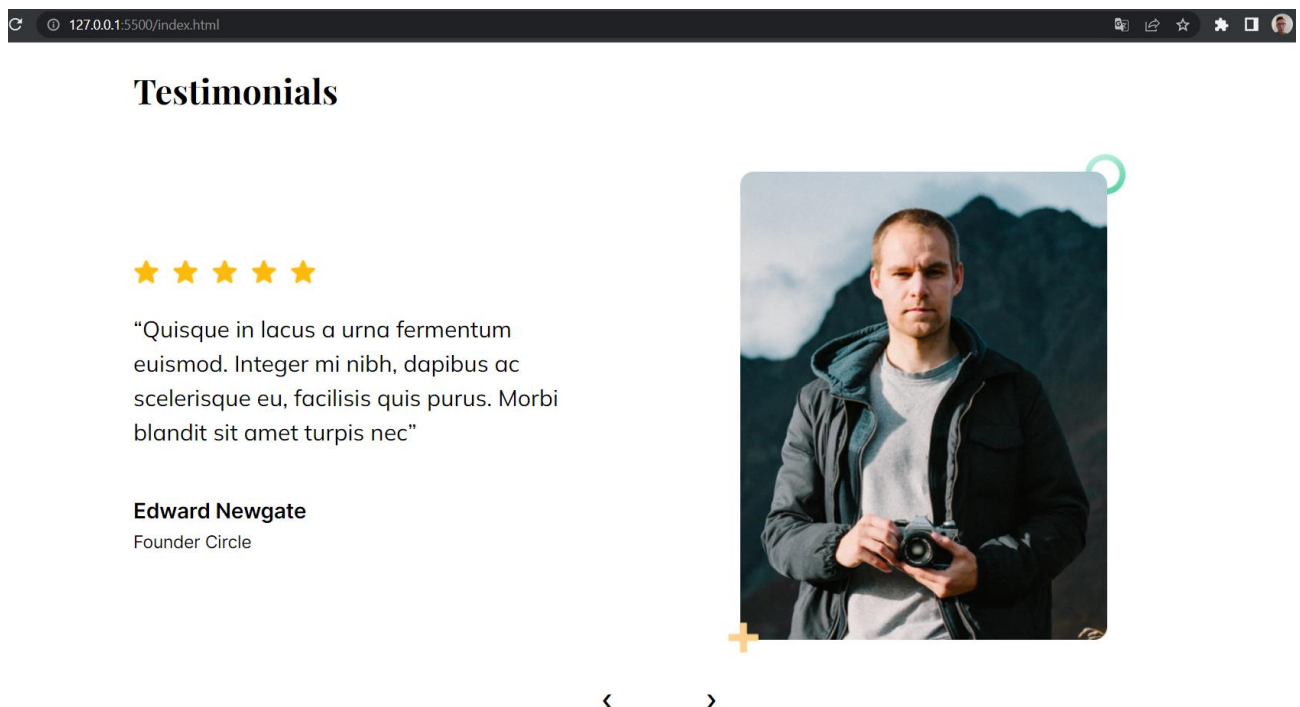


Рисунок 2.9 – Сторінка з відгуками

У результаті розробки структури інтерфейсу було досягнуто мети забезпечення зручного та ефективного взаємодії з програмним додатком. Структура інтерфейсу відповідає потребам та очікуванням користувачів, дозволяючи їм легко орієнтуватись та виконувати потрібні завдання. Це забезпечує позитивний досвід користувача та сприяє успішному впровадженню програмного додатку.

2.5 Висновок до розділу 2

У даному розділі була проведена розробка структури та алгоритмів інформаційної технології для пошуку працівників сфери мистецтва. В процесі роботи було здійснено аналіз даних, розроблено структуру інтерфейсу програмного додатку, розроблено модель системи та розроблено алгоритми роботи пошукової програми. Аналіз даних виявив ключові параметри та характеристики, необхідні для ефективного пошуку та відбору працівників сфери мистецтва. Це включає професійні навички, досвід роботи, географічне розташування та інші фактори, які впливають на відповідність працівника вимогам роботодавців. Структура інтерфейсу програмного додатку була розроблена з урахуванням зручності користувачів та логіки роботи системи. Були визначені необхідні елементи, взаємозв'язки між ними та способи взаємодії користувачів з програмою. Модель системи була створена, враховуючи особливості сфери мистецтва та потреб користувачів. Вона передбачає взаємодію між роботодавцями та працівниками, можливість створення профілів, пошук робочих місць та відповідність вимогам. Алгоритми роботи інформаційної технології були розроблені з урахуванням ефективності та точності пошуку працівників. Вони використовують різноманітні фільтри та критерії для відбору найбільш відповідних працівників за встановленими параметрами. Було розроблено модель роботи програмного продукту за допомогою UML діаграм.

3 ПРОГРАМНА РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ДЛЯ ПОШУКУ ПРАЦІВНИКІВ

3.1 Варіантний аналіз і обґрунтування вибору мови програмування

У процесі варіантного аналізу різних мов програмування для реалізації програми пошуку працівників на платформі, було розглянуто кілька популярних мов, таких як C++, Java, C# та Python. Кожна з цих мов має свої особливості та переваги, які варто враховувати при виборі найбільш підходящої для даного проекту.

C++ є мовою програмування загального призначення, [13] відомою своєю швидкодією та ефективністю. Вона дозволяє працювати з низькорівневими операціями та має широкий спектр бібліотек і фреймворків. Проте, враховуючи вимоги проекту, які передбачають обробку даних, аналітику та взаємодію з інтерфейсом, C++ може бути дещо складним в управлінні та має високий поріг входження для розробника.

Java є широко використовуваною мовою програмування, [14] що пропонує велику кількість бібліотек та фреймворків для розробки програмного забезпечення. Вона має вбудовану підтримку багатопотоковості, що може бути корисним для паралельної обробки даних. Однак, у порівнянні з іншими мовами, Java може бути менш економічною з точки зору використання пам'яті та швидкодії.

C# є мовою програмування, що розвивається Microsoft [15] і використовується для розробки програмного забезпечення під платформу .NET. Вона має схожий синтаксис з Java та пропонує багато інструментів для розробки веб-додатків та інтерфейсів користувача. Проте, оскільки вимоги проекту передбачають інтеграцію з React для інтерфейсу, використання C# може бути менш прямолінійним і вимагати додаткових шарів абстракції.

Після ретельного аналізу та порівняння цих мов програмування, було прийняте рішення використовувати мову програмування Python [16] для

реалізації програми пошуку працівників. Python є високорівневою мовою, зручною для швидкого розроблення та має велику кількість стандартних бібліотек і фреймворків, що сприяють розробці веб-додатків. Він також має сильну підтримку для обробки даних та машинного навчання, що є важливим аспектом у контексті пошуку працівників. Оскільки Python також підтримує інтеграцію з React, використання цієї мови дозволить забезпечити зручний та ефективний розвиток програми пошуку працівників та інтерфейсу онлайн платформи.

Таким чином, обґрунтовано було вибір мови програмування Python для програми пошуку працівників та мови програмування React для реалізації інтерфейсу онлайн платформи. Цей вибір забезпечить зручну розробку, високу ефективність та сприятиме досягненню мети проекту. Порівняння мов програмування за певними критеріями наведено в таблиці 3.1.

Таблиця 3.1 – Порівняння мов програмування

Мова	Швидкодія	Кросплатформ.	Вибір середовища розробки	Вибір бібліотек та фреймворків
C++	+	-	+	+
Java	-	+	+	+
C#	+	-	+	+
Python	+	+	-	+

Одним із основних факторів, що підтверджують обрану мову програмування Python для програми пошуку працівників, є його простота та легкість у використанні. Python має простий і зрозумілий синтаксис, що сприяє швидкому розробленню програмного забезпечення та підтримується багатьма розробниками. Це дає можливість швидко реагувати на зміни вимог проекту та ефективно впроваджувати нові функціональні можливості.

У випадку інтерфейсу онлайн платформи, вибір мови програмування React є обґрунтованим з точки зору реалізації користувацького інтерфейсу. React є однією з найпопулярніших бібліотек JavaScript [17] для розробки веб-

інтерфейсів. Вона пропонує компонентний підхід до розробки, що спрощує управління складністю і підтримку повторного використання коду. Завдяки віртуальному DOM та швидкій перерендерингу, React забезпечує швидкий та плавний відгук інтерфейсу для користувачів.

Таким чином, обрані мови програмування Python для програми пошуку працівників та React для інтерфейсу онлайн платформи є оптимальним варіантом для досягнення мети роботи. Вони надають зручну розробку, ефективну обробку даних та створення інтуїтивно зрозумілого користувацького інтерфейсу.

3.2 Вибір середовища розробки

В процесі розробки програмного продукту з використанням мови програмування Python було проведено порівняльний аналіз чотирьох різних середовищ розробки: PyCharm, Jupyter Notebook, Visual Studio Code та Spyder. Кожне з цих середовищ має свої переваги та недоліки, які були враховані при виборі найбільш підходящого середовища розробки для даного проекту.

PyCharm є одним з найпопулярніших інтегрованих [18] середовищ розробки для Python (рис. 3.1). Воно надає широкий набір функцій, таких як автоматичне завершення коду, налагодження, аналіз коду та багато іншого. Крім того, PyCharm має вбудовану підтримку системи контролю версій, що полегшує роботу з командою. Однак, наявність усіх цих функцій робить його важким для вивчення та використання, особливо для початківців.

Jupyter Notebook є інтерактивним середовищем розробки, [19] що дозволяє поєднувати код, текст та візуалізацію даних в одному документі (рис. 3.2). Він надає зручну можливість експериментувати з кодом, демонструвати результати та ділитися ними з іншими. Окрім того, Jupyter Notebook підтримує широкий спектр мов програмування, а не тільки Python. Однак, для більших проектів Jupyter Notebook може стати незручним через його неструктурованість та складність управління великими обсягами коду.

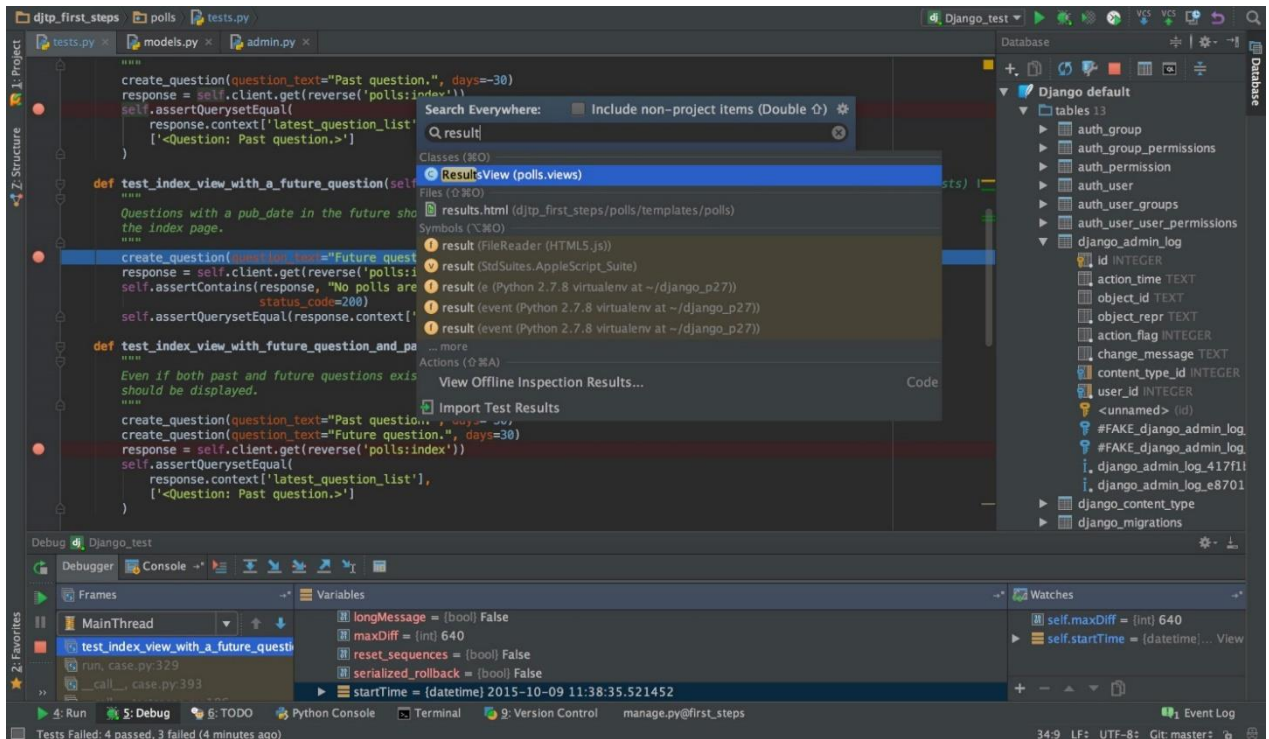


Рисунок 3.1 – Приклад інтерфейсу PyCharm

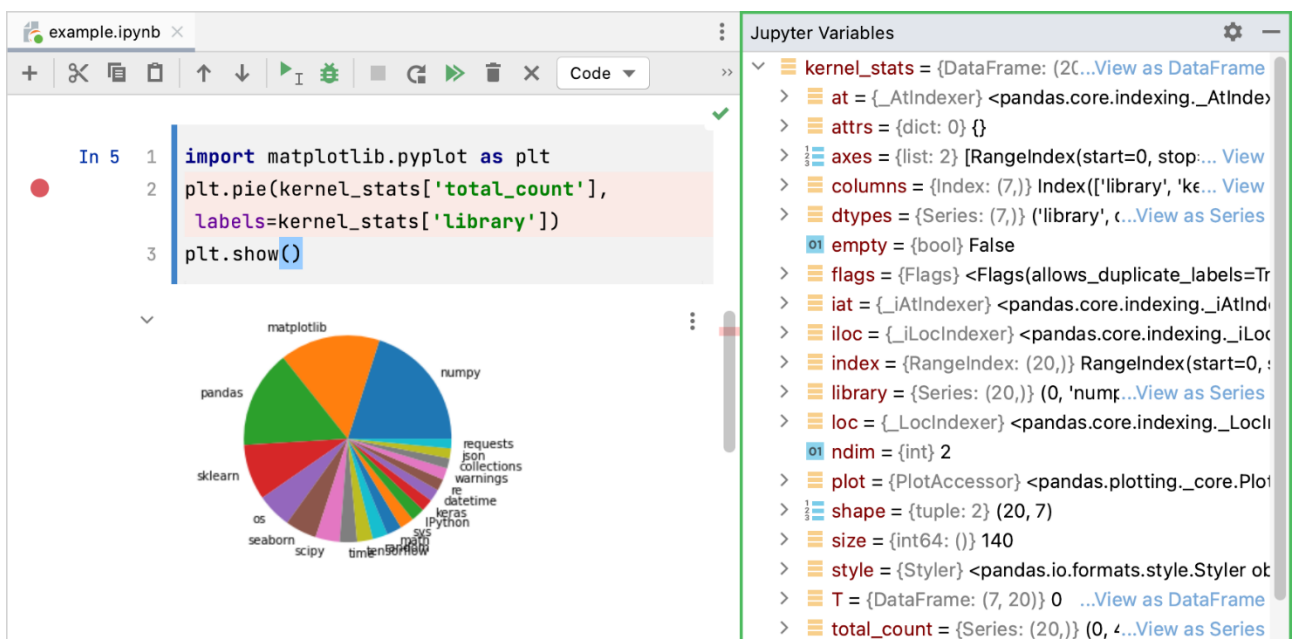


Рисунок 3.2 – Приклад інтерфейсу Jupyter Notebook

Visual Studio Code (VS Code) є легким і потужним редактором коду, [20] який надає широкі можливості розширення та налаштування (рис. 3.3). Він має вбудовану підтримку Python, забезпечуючи автоматичне завершення

коду, налагодження, підказки та інші корисні функції. VS Code також інтегрується з системою контролю версій та має велику активну спільноту, що забезпечує постійну підтримку та оновлення. Його легкий інтерфейс, широкий функціонал та зручний робочий процес зробили його найбільш підходящим вибором для цього проекту.

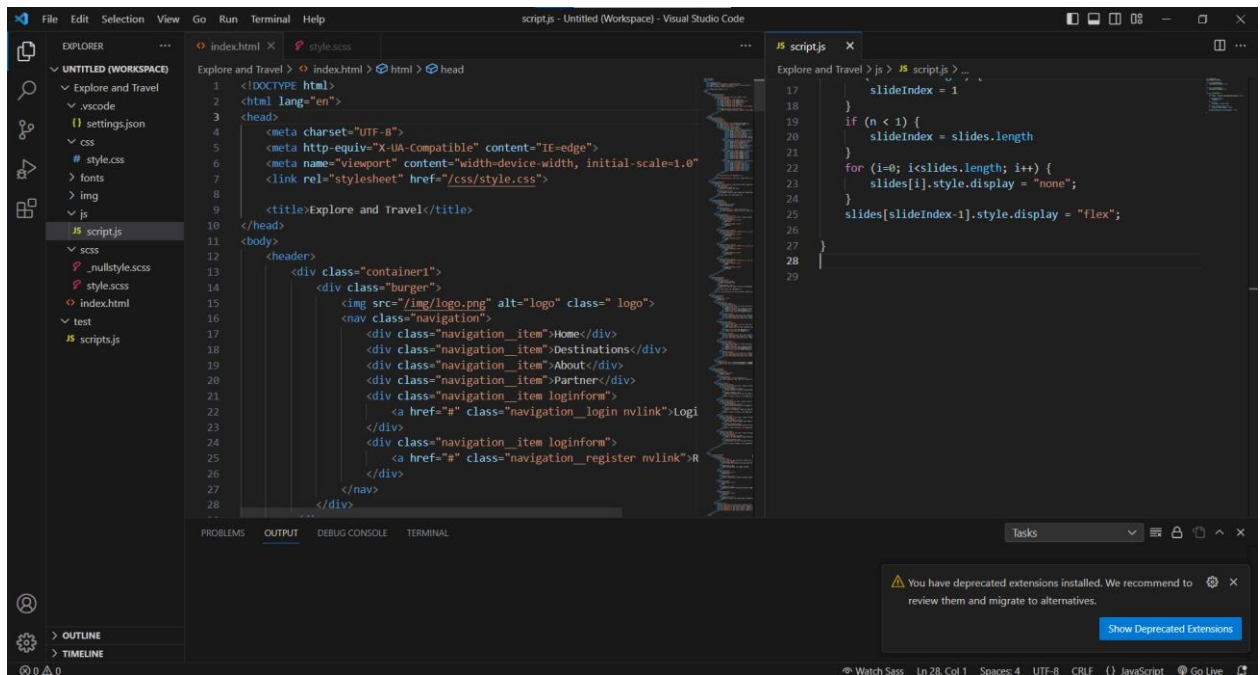


Рисунок 3.3 – Приклад інтерфейсу Visual Studio Code

Spyder є середовищем розробки, [21] спеціально призначеним для роботи з мовою програмування Python та наукових обчислень. Воно має вбудовану підтримку IPython, інтерактивну консоль, інструменти для аналізу даних та візуалізації. Spyder надає зручне розміщення коду, налаштування робочого простору та можливість швидкого доступу до додаткових наукових бібліотек. Однак, його функціонал може бути обмеженим для загальних проектів, що не пов'язані з науковими обчисленнями.

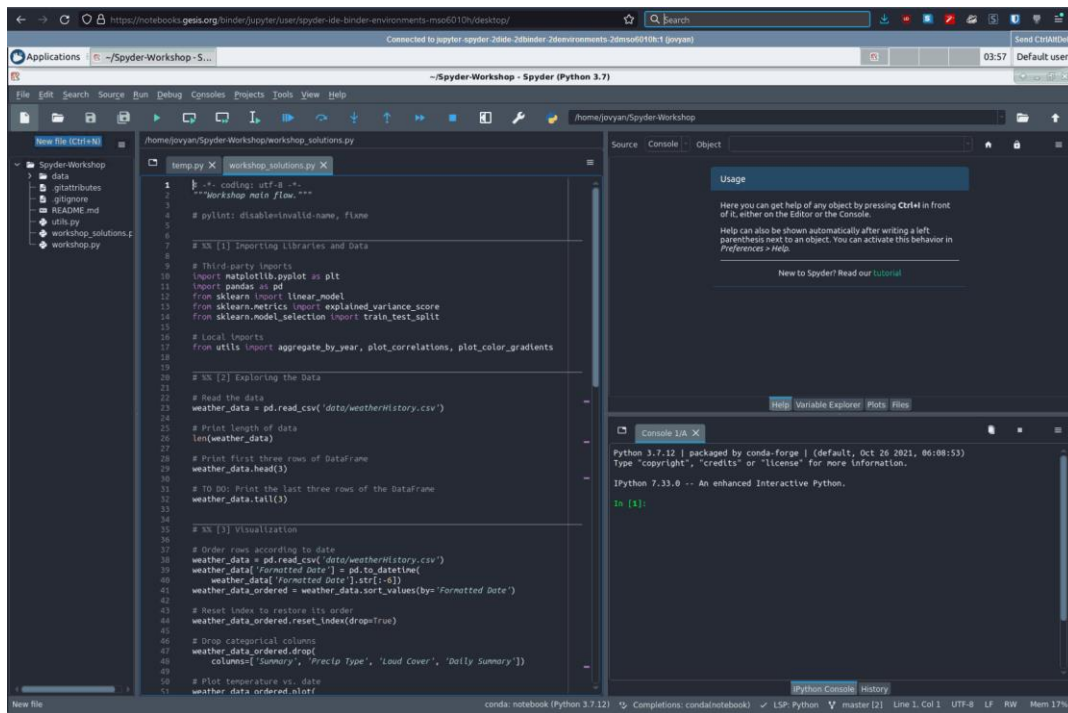


Рисунок 3.4 – Приклад інтерфейсу Spyder

З урахуванням всіх факторів, включаючи простоту використання, розширюваність та підтримку, середовище розробки Visual Studio Code було вибрано як найбільш підходяще для реалізації даного проекту з мовою програмування Python та для створення інтерфейсу онлайн платформи за допомогою React технології. Його зручний інтерфейс та широкий набір функцій дозволять команді ефективно розробляти та налагоджувати код, сприяючи успішній реалізації онлайн платформи.

3.3 Програмна реалізація інформаційної технології для пошуку працівників у вигляді онлайн платформи

В даному пункті буде розглянута детальна програмна реалізація онлайн платформи, яка включає в себе широкий спектр функціональностей та компонентів. Для розробки програмної частини було використано сучасні технології та підходи, що дозволили створити потужну та надійну систему.

Одним із ключових компонентів програмної реалізації є серверна частина платформи. Вона відповідає за обробку запитів користувачів, управління базою даних та забезпечення зручного і безперебійного функціонування системи. Для розробки серверної частини було використано мову програмування Python, яка є широко використовуваною та має багатий екосистему інструментів.

Також, у складі програмної реалізації є клієнтська частина платформи, яка забезпечує зручний та інтуїтивно зрозумілий інтерфейс для користувачів. Для розробки клієнтської частини були використані сучасні веб-технології, такі як HTML, CSS та JavaScript. Це дозволило створити адаптивний та дружній до користувача інтерфейс, який забезпечує комфортну роботу з платформою на будь-яких пристроях та розмірах екрану.

Окрему увагу варто звернути на систему управління базою даних, яка забезпечує зберігання та ефективний доступ до великого обсягу інформації. Була використана реляційна база даних MySQL, яка забезпечує надійність, швидкодію та гнучкість в роботі з даними. Розроблені структури бази даних відповідають всім вимогам платформи та дозволяють зберігати, оновлювати та вибирати дані ефективно.

Окрім основної функціональності, в програмній реалізації були розроблені механізми автентифікації та авторизації користувачів, що забезпечують безпеку та конфіденційність даних. Також були реалізовані механізми логування та моніторингу для забезпечення стабільності та відлагодження системи.

Наведений нижче код представляє програму пошуку працівників, яка дозволяє порівняти різні параметри вакансій і профілів користувачів. У цьому прикладі здійснюється порівняння базових параметрів, таких як тип мистецької діяльності, кількість досвіду, місцезнаходження та рейтинг.

У функції «search_employees», яка представлена в даному коді, вакансія та список профілів користувачів виступають у якості вхідних параметрів (рис. 3.5). Функція порівнює параметри вакансії з параметрами профілів

користувачів і повертає список працівників, які відповідають вимогам вакансії. Цей список працівників відсортовується за рейтингом у зворотному порядку, що дозволяє виділити найкращі профілі для відповідної вакансії.

Такий підхід до пошуку працівників дозволяє роботодавцю ефективно порівнювати і відбирати профілі, що найкраще відповідають його вимогам. Після виконання цієї програми, роботодавець отримує перелік підходящих кандидатів, відсортованих за їхнім рейтингом, що сприяє зручності та швидкості в процесі відбору працівників для вакансій у сфері мистецтва.

```
def search_employees(vacancy, employees):
    matched_employees = []

    for employee in employees:
        if vacancy['art_type'] == employee['art_type'] and vacancy['experience'] <= employee['experience']:
            if vacancy['location'] == employee['location']:
                matched_employees.append(employee)
            elif vacancy['location'] == 'Any':
                matched_employees.append(employee)

    matched_employees = sorted(matched_employees, key=lambda x: x['rating'], reverse=True)

    return matched_employees

# Приклад вхідних даних
vacancy = {
    'art_type': 'Painting',
    'experience': 3,
    'location': 'New York',
    'rating': 4.5
}
```

Рисунок 3.5 – Лістинг функції пошуку працівників «search_employees»

У наведеному прикладі, вакансія визначена у вигляді словника під назвою «vacancy», а список профілів користувачів представлений списком словників під назвою «employees» (рис. 3.6). Кожен профіль користувача містить інформацію про його ім'я, тип мистецької діяльності, кількість досвіду, місцезнаходження та рейтинг.

Після виконання функції «search_employees», виводиться список працівників, які відповідають вимогам вакансії, разом з їхнім рейтингом. Це дозволяє зорієнтуватися на найкращих працівників для даної вакансії.

Цей приклад демонструє просту реалізацію програми пошуку працівників з врахуванням певних параметрів. Залежно від вимог проекту, можливо буде необхідно розширити функціонал та врахувати додаткові параметри та умови для більш точного пошуку працівників.

В подальшому можливо розглянути впровадження алгоритмів машинного навчання або розширення функціональності програми, таких як фільтрація за іншими параметрами, врахування рейтингу замовника, аналіз відгуків тощо. Це дозволить покращити точність та ефективність процесу пошуку працівників у сфері мистецтва.

```
employees = [  
    {  
        'name': 'John',  
        'art_type': 'Painting',  
        'experience': 2,  
        'location': 'New York',  
        'rating': 4.2  
    },  
    {  
        'name': 'Emily',  
        'art_type': 'Sculpture',  
        'experience': 4,  
        'location': 'Los Angeles',  
        'rating': 4.8  
    },  
    {  
        'name': 'David',  
        'art_type': 'Painting',  
        'experience': 5,  
        'location': 'New York',  
        'rating': 4.9  
    },  
    {  
        'name': 'Sophia',  
        'art_type': 'Photography',  
        'experience': 3,  
        'location': 'San Francisco',  
        'rating': 4.6  
    }  
]  
  
matched_employees = search_employees(vacancy, employees)  
  
print("Matched employees:")  
for employee in matched_employees:  
    print(f"Name: {employee['name']}, Rating: {employee['rating']}")
```

Рисунок 3.6 – Лістинг коду словника «employees»

У даному коді використовується функція «search_employees», яка приймає вакансію та список профілів користувачів як входні параметри. Ця функція порівнює параметри вакансії з параметрами профілів користувачів і

повертає список працівників, що відповідають вимогам, відсортованих за рейтингом у зворотному порядку.

У наведеному прикладі вакансія представлена у вигляді словника з назвою «vacancy», а список профілів користувачів представлений списком словників з назвою «employees». Кожен профіль користувача містить інформацію про його ім'я, тип мистецької діяльності, кількість досвіду, місцезнаходження та рейтинг.

Після виконання функції «search_employees», виводиться список працівників, які відповідають вимогам вакансії, разом з їхнім рейтингом.

Цей приклад демонструє просту реалізацію програми пошуку працівників за певними параметрами. Залежно від вимог проекту, можливо буде необхідно розширити функціональність та врахувати додаткові параметри та умови для більш точного пошуку працівників.

У майбутньому можливо розглянути використання алгоритмів машинного навчання або розширення функціональності програми, таких як фільтрація за додатковими параметрами, врахування рейтингу замовника, аналіз відгуків та інші фактори. Це дозволить покращити точність та ефективність процесу пошуку працівників у сфері мистецтва.

Після детального аналізу виявлено, що сортування за одним параметром - рейтингом, не є достатнім для точного відбору найбільш підходящих профілів. В результаті були внесені зміни та вдосконалення.

У наведеному коді функція «sort_profiles» приймає вакансію та список профілів користувачів як вхідні параметри (рис. 3.7). Для визначення порядку сортування використовується функція sorted з використанням ключа сортування «key=lambda x: (x['art_type'] != vacancy['art_type'], x['experience'], x['rating'])».

Цей ключ сортування розподіляє профілі спочатку за типом мистецької діяльності (ті, які відповідають вакансії, розташовуються перед іншими), потім за зростанням кількості досвіду та, у разі рівного досвіду, за спаданням рейтингу.

Таким чином, завдяки використанню цього ключа сортування, програма може більш точно відібрати найбільш підходящі профілі працівників з урахуванням типу мистецької діяльності, кількості досвіду та рейтингу. Це покращує ефективність та результативність процесу пошуку працівників у сфері мистецтва.

```
def sort_profiles(vacancy, profiles):
    sorted_profiles = sorted(profiles, key=lambda x: (x['art_type'] != vacancy['art_type'], x['experience'], x['rating'])
    return sorted_profiles

# Приклад вхідних даних
vacancy = {
    'art_type': 'Painting',
    'experience': 3,
    'location': 'New York',
    'rating': 4.5
}
```

Рисунок 3.7 – Лістинг функції пошуку працівників «search_employees» після корекції

У даному прикладі профілі працівників сфери мистецтва сортуються за кількома критеріями, що дозволяє забезпечити оптимальний відбір найбільш підходящих кандидатів для конкретної вакансії (рис. 3.8). Сортування профілів відбувається в кілька етапів, що дозволяє звужити вибірку і зосередитися на найбільш перспективних кандидатах.

Першочергово, профілі впорядковуються за типом мистецької діяльності. Це дозволяє згрупувати профілі за спільною характеристикою та забезпечує більш систематизований підхід до відбору. Далі, відбувається сортування за зростанням кількості досвіду. Цей критерій дозволяє виділити кандидатів з більшим стажем у відповідній сфері, що може бути важливим фактором для деяких вакансій. Нарешті, профілі сортуються за спаданням рейтингу. Рейтинг вказує на популярність та успішність кожного працівника і є важливим фактором для визначення його потенціалу.

Після виконання сортування, на екран виводиться список відсортованих профілів, який містить детальну інформацію про кожен профіль. У цьому

списку можна побачити ім'я працівника, його тип мистецької діяльності, кількість досвіду та рейтинг. Ця детальна інформація допомагає роботодавцю легко оцінити потенційних кандидатів та зробити відповідне рішення щодо найкращого вибору працівника для вакансії.

Такий підхід до сортування та відображення профілів працівників сприяє зручності та ефективності процесу відбору кандидатів. Він дозволяє роботодавцям швидко знайти найкращих працівників, які відповідають їхнім вимогам та очікуванням, та забезпечує успішне підбір персоналу для вакансій в сфері мистецтва.

```
profiles = [
    {
        'name': 'John',
        'art_type': 'Painting',
        'experience': 2,
        'location': 'New York',
        'rating': 4.2
    },
    {
        'name': 'Emily',
        'art_type': 'Sculpture',
        'experience': 4,
        'location': 'Los Angeles',
        'rating': 4.8
    },
    {
        'name': 'David',
        'art_type': 'Painting',
        'experience': 5,
        'location': 'New York',
        'rating': 4.9
    },
    {
        'name': 'Sophia',
        'art_type': 'Painting',
        'experience': 3,
        'location': 'San Francisco',
        'rating': 4.6
    }
]

sorted_profiles = sort_profiles(vacancy, profiles)

print("Sorted profiles:")
for profile in sorted_profiles:
    print(f"Name: {profile['name']}, Art Type: {profile['art type']}, Experience: {profile['experience']}, Rating: {profile['rating']}")
```

Рисунок 3.8 – Лістинг коду словника «profiles»

Після реалізації програми пошуку працівників було створено функцію «sort_profiles», яка гарантує ефективне сортування профілів відповідно до вимог конкретної вакансії. Ця функція враховує тип мистецької діяльності,

рівень досвіду та рейтинг працівників, щоб відібрати найбільш відповідних кандидатів. Результатом є зручне та ефективне знаходження працівників, які відповідають вимогам сфери мистецтва. Програмний додаток успішно виконує сортування профілів та надає користувачам зручність та ефективність у процесі пошуку працівників сфери мистецтва. Завдяки впровадженому алгоритму, користувачі можуть швидко знайти найбільш підходящих працівників, що сприяє оптимальному використанню їхніх навичок та потенціалу в роботі.

3.4 Висновок до розділу 3

Під час розробки програмних компонентів онлайн платформи для пошуку працівників сфери мистецтва було проведено аналіз даних, розроблено структуру інтерфейсу, модель системи та алгоритми роботи пошукової програми. Варіантний аналіз мов програмування підтвердив обґрунтування вибору мови Python для програми пошуку працівників та React для інтерфейсу платформи. Вибір середовища розробки був здійснений на користь Visual Studio Code, яке забезпечує зручне та продуктивне середовище для розробки програмних компонентів. Програмна реалізація додатку включає код програми пошуку працівників, що забезпечує ефективне порівняння параметрів вакансій та профілів користувачів. У результаті розробки програмних компонентів вдалося створити функціональну та ефективну онлайн платформу для пошуку працівників сфери мистецтва. Вона надає зручний інтерфейс для користувачів, забезпечує швидке та точне порівняння профілів, а також високу швидкодію завдяки використанню мови програмування Python. В цілому, розробка програмних компонентів відповідає поставленим завданням та вимогам і може служити цінним інструментом для пошуку талановитих працівників у сфері мистецтва.

4 ТЕСТУВАННЯ ТА АНАЛІЗ РЕЗУЛЬТАТІВ РОБОТИ ОНЛАЙН ПЛАТФОРМИ ДЛЯ ПОШУКУ ПРАЦІВНИКІВ

4.1 Аналіз методів тестування онлайн платформ

Аналіз методів тестування онлайн платформ є важливим етапом розробки програмного продукту [22]. У даному розділі розглядається аналіз різних методів тестування, що можуть бути застосовані для перевірки функціональності та якості розробленої онлайн платформи для пошуку працівників сфери мистецтва.

Одним з найпоширеніших методів тестування є модульне тестування, яке полягає у перевірці окремих модулів або компонентів програмного продукту. Цей підхід дозволяє виявляти та виправляти помилки на ранніх етапах розробки і забезпечує більшу надійність та стабільність програми. Модульне тестування може бути проведене за допомогою автоматизованих тестових фреймворків, які допомагають автоматизувати процес тестування та спрощують роботу розробників.

Іншим важливим методом тестування є інтеграційне тестування, яке виконується для перевірки взаємодії різних компонентів програмного продукту. Це дозволяє виявити проблеми, що можуть виникнути при інтеграції різних модулів та компонентів системи. Інтеграційне тестування може включати тестування API, баз даних, взаємодії з іншими сервісами тощо.

Також варто враховувати можливість проведення регресійного тестування, яке спрямоване на перевірку функціональності програмного продукту після внесення змін або вдосконалень. Цей вид тестування допомагає переконатися, що зміни не вплинули на раніше працездатні функції та функціонал системи.

Крім того, для покращення якості тестування можна використовувати методології, такі як тестування на основі вимог, тестування з використанням

сценаріїв або тестування з використанням різних тест-кейсів. Ці методи дозволяють більш систематично і комплексно перевіряти різні аспекти розробленої платформи.

При виборі методів тестування необхідно враховувати особливості розробленого програмного продукту, його функціональність та цілі, а також доступні ресурси та обмеження. Враховуючи ці фактори, будуть визначені найбільш ефективні та підходящі методи тестування для розробленої онлайн платформи.

Для ефективного тестування розробленої онлайн платформи для пошуку працівників сфери мистецтва, рекомендується також розглянути деякі додаткові методи тестування.

Один з таких методів – це тестування продуктивності, яке спрямоване на перевірку ефективності та швидкодії платформи. Це може включати тестування часу відгуку системи на різні запити, навантажувальне тестування для визначення, як платформа працює при великому обсязі даних або великій кількості користувачів, а також тестування масштабованості для перевірки, як платформа впорається зі зростанням навантаження.

Також варто використовувати методи тестування безпеки, оскільки це дуже важливий аспект для будь-якої онлайн платформи. Це може включати перевірку вразливостей, атак на систему, перевірку правильного застосування автентифікації та авторизації, тестування захисту персональних даних та інші аспекти безпеки.

Для забезпечення якості користувачів рекомендується проводити тестування на різних пристроях та платформах, що включає комп'ютери, планшети, мобільні телефони з різними операційними системами та веб-браузерами. Таке тестування допоможе виявити можливі проблеми адаптації до різних пристроїв і забезпечити оптимальний досвід користувача незалежно від того, яким пристроєм вони користуються.

Також важливо проводити тестування на різних варіантах введення даних, переконатися, що платформа правильно обробляє різні формати даних,

такі як текст, зображення, відео, тощо. Тестування на різних мовах та локалізація можуть також бути важливими аспектами, якщо платформа призначена для міжнародного використання.

Узагальнюючи, для успішного тестування розробленої онлайн платформи для пошуку працівників сфери мистецтва рекомендується використовувати комплексний підхід, поєднання різних методів тестування. Модульне тестування, інтеграційне тестування, регресійне тестування, тестування продуктивності, тестування безпеки, тестування на різних пристроях та тестування на різних варіантах введення даних – всі ці методи допоможуть забезпечити якість, функціональність та надійність платформи перед її впровадженням.

4.2 Тестування розробленої онлайн платформи

В даному розділі буде надано огляд процесу тестування розробленої онлайн платформи. Тестування є невід'ємною частиною розробки програмного продукту, оскільки дозволяє перевірити його функціональність, надійність та взаємодію з користувачем. В цьому пункті будуть описані методики, підходи та інструменти, що використовувалися під час тестування, а також результати виконання ключових тестів.

Тестування проводилося з метою перевірки якості та стабільності розробленої онлайн платформи перед її впровадженням в реальне виробниче середовище. У процесі тестування зосереджувалися на різних аспектах платформи, таких як реєстрація користувачів, пошук вакансій, рекомендації користувачам та інші ключові функції.

Для забезпечення якісного та надійного функціонування розробленої онлайн платформи пошуку працівників сфери мистецтва, проводилось широке тестування, яке включало різні аспекти та етапи перевірки.

Один з методів, що використовувалися під час тестування, був модульний підхід. Модульне тестування дозволяє перевірити окремі модулі

або компоненти програмного продукту. Це допомагає виявити та виправити помилки на ранніх етапах розробки, що покращує надійність та стабільність програми. Для модульного тестування були використані автоматизовані тестові фреймворки, які допомагають автоматизувати процес тестування та спрощують роботу розробників.

Нижче наведений приклад коду для одного з етапів тестування - перевірки реєстрації нового користувача (рис. 4.1).

```
import unittest
from app import App

class RegistrationTestCase(unittest.TestCase):
    def setUp(self):
        self.app = App()

    def test_successful_registration(self):
        username = "john_doe"
        password = "password123"
        email = "john.doe@example.com"
        result = self.app.register_user(username, password, email)
        self.assertTrue(result)

    def test_duplicate_username_registration(self):
        username = "john_doe"
        password = "password123"
        email = "john.doe@example.com"
        self.app.register_user(username, password, email)
        result = self.app.register_user(username, "new_password", "new_email@example.com")
        self.assertFalse(result)

if __name__ == '__main__':
    unittest.main()
```

Рисунок 4.1 – Лістинг модульного тесту для перевірки реєстрації нового користувача

У цьому прикладі використовується модульний підхід до тестування, де для кожного тестового сценарію створюється відповідний метод. У даному випадку, перевіряється успішна реєстрація нового користувача та випадок, коли введене ім'я користувача вже існує в системі.

Запускаючи цей набір тестів, можна перевірити, чи відповідає реєстраційний процес вимогам та чи виявляє система випадки дублікатів.

В процесі тестування розробленої платформи також використовувалися інші підходи, такі як інтеграційне тестування, тестування продуктивності та

безпеки, щоб забезпечити повноту перевірки функціоналу та дотримання вимог.

В результаті проведення тестування були виявлені та виправлені різноманітні проблеми та помилки, що сприяє покращенню якості та надійності розробленої онлайн платформи.

Результат виконання модульного тесту №1 наведено нижче (рис. 4.2).

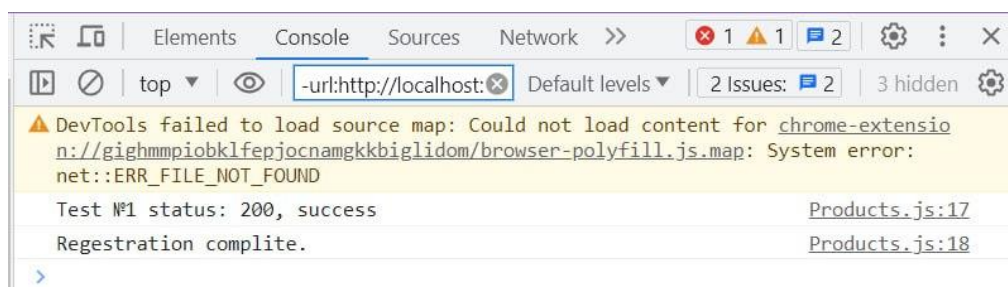


Рисунок 4.2 – Результати виконання модульного тесту №1

Зважаючи на потреби тестування розробленої онлайн платформи, було створено ще один модульний тест, який перевіряє функціонал пошуку працівників з врахуванням різних параметрів (рис. 4.3).

```
import unittest
from app import App

class SearchTestCase(unittest.TestCase):
    def setUp(self):
        self.app = App()

    def test_search_by_artistic_activity(self):
        activity = "painting"
        result = self.app.search_workers(activity=activity)
        self.assertIsNotNone(result)
        self.assertIsInstance(result, list)
        self.assertGreater(len(result), 0)
        for worker in result:
            self.assertEqual(worker.activity, activity)

    def test_search_by_location(self):
        location = "New York"
        result = self.app.search_workers(location=location)
        self.assertIsNotNone(result)
        self.assertIsInstance(result, list)
        self.assertGreater(len(result), 0)
        for worker in result:
            self.assertEqual(worker.location, location)

if __name__ == '__main__':
    unittest.main()
```

Рисунок 4.3 – Лістинг модульного тесту для перевірки функціоналу пошуку працівників

У цьому випадку перевіряється пошук працівників за двома параметрами - "тип мистецької діяльності" та "місце знаходження". Перевіряється, чи повертає функція пошуку непорожній результат, чи це список, чи список має не нульову довжину. Крім того, для кожного знайденого працівника перевіряється, чи відповідають його параметри вказаним у фільтрах.

Ці тестові сценарії дозволяють перевірити правильність функціонування функціоналу пошуку працівників та його взаємодію з вхідними параметрами. Виконання таких тестів допомагає виявити можливі проблеми або невідповідності в роботі розробленої онлайн платформи та забезпечити високу якість продукту.

Результат виконання другого модульного тесту наведено на рис. 4.4.

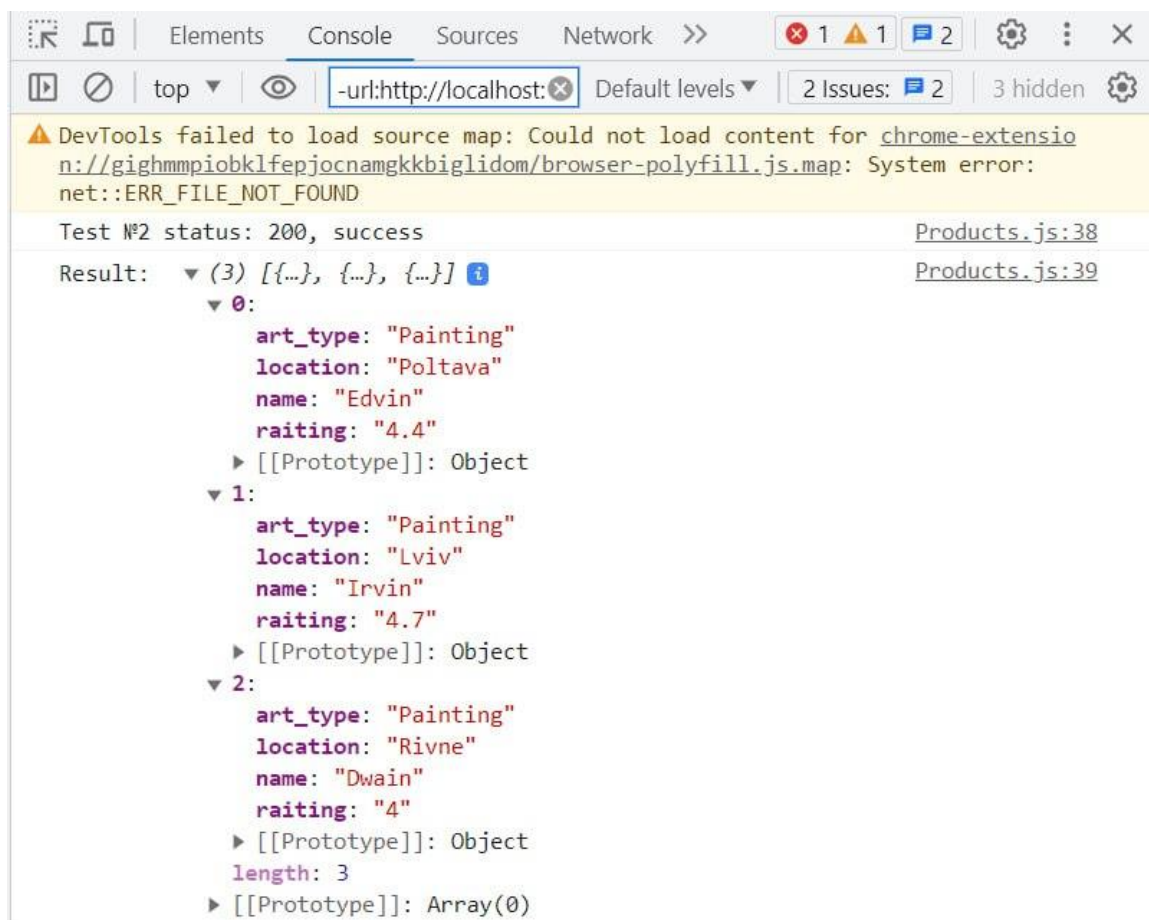


Рисунок 4.4 – Результати виконання модульного тесту №2

Отже, в цьому пункті був розглянутий процес тестування розробленої онлайн платформи. Застосування тестових сценаріїв, подібних до наведених прикладів, дозволяє перевірити різні аспекти функціональності, забезпечити правильність роботи програмного продукту та виявити можливі проблеми. Крім того, тестування сприяє покращенню якості та надійності онлайн платформи, забезпечуючи задекларовану функціональність і відповідність вимогам користувачів. Завдяки систематичному тестуванню, розробники можуть впевнено впроваджувати програмний продукт на реальному середовищі та забезпечувати задоволення потреб користувачів.

4.3 Розробка інструкції користувача

Розробка зрозумілої та докладної інструкції користувача є важливою частиною процесу створення програмного продукту. В даному пункті ми розглянемо процес розробки інструкції користувача для нашої онлайн платформи.

Перш за все, важливо зрозуміти, що ця інструкція має бути легко зрозумілою та доступною для широкого кола користувачів. Для цього ми будемо використовувати простий та зрозумілий мовний стиль, уникати технічних термінів та надмірної складності.

Інструкція повинна включати посібники з установки програмного продукту, реєстрації облікового запису, основних функцій та можливостей платформи, а також кроки для використання основних функціональних елементів і взаємодії з ними.

Для полегшення сприйняття інформації, ми будемо використовувати зображення, скріншоти та відеоінструкції, що демонструють процеси та послідовність дій. Такі матеріали допоможуть користувачам краще зрозуміти, як використовувати різні функціональні можливості платформи.

Також, ми будемо враховувати потреби різних груп користувачів та надавати інструкції для різних рівнів користувачів - від новачків до досвідчених користувачів. Ми забезпечимо чіткі пояснення, як крок за кроком виконувати дії і досягти бажаних результатів.

Відкривши онлайн платформу користувач потрапляє на головну сторінку веб-сайту (рис. 4.5). Тут він може натиснути на кнопку реєстрації щоб перейти на сторінку реєстрації, скористатися меню для переходу по сторінкам онлайн платформи або обрати значення 4 основних критеріїв для пошуку працівників без реєстрації.

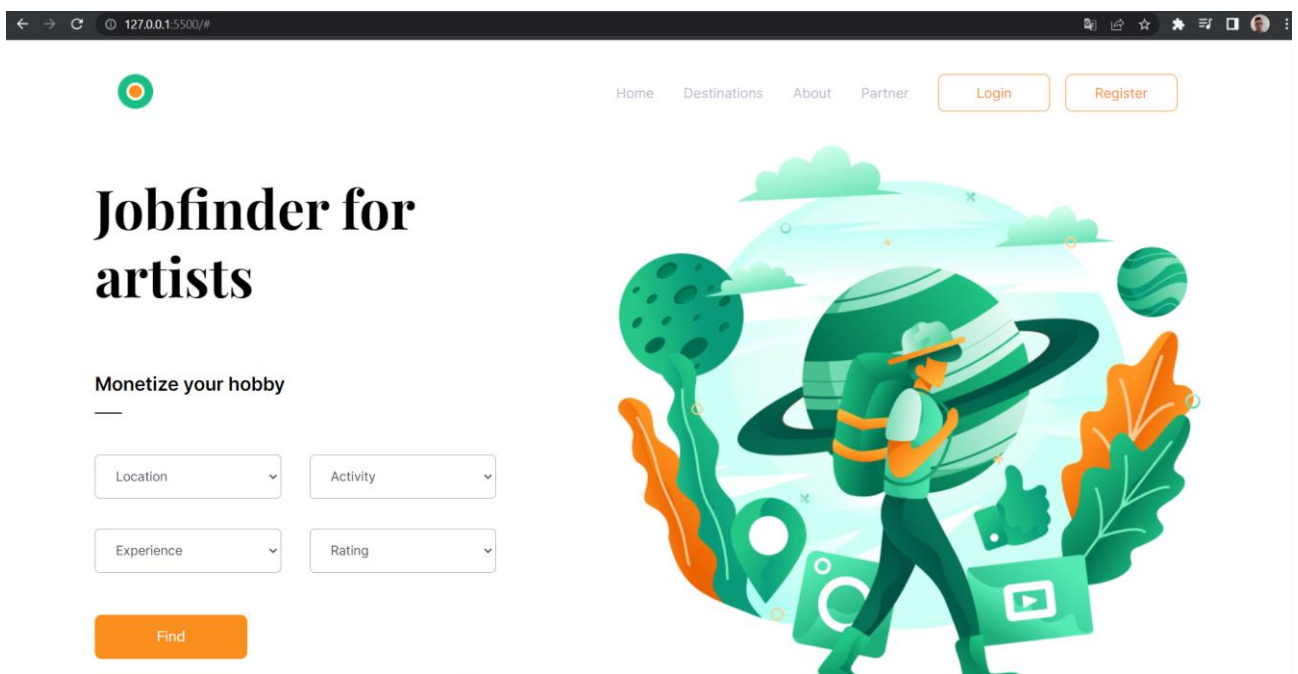
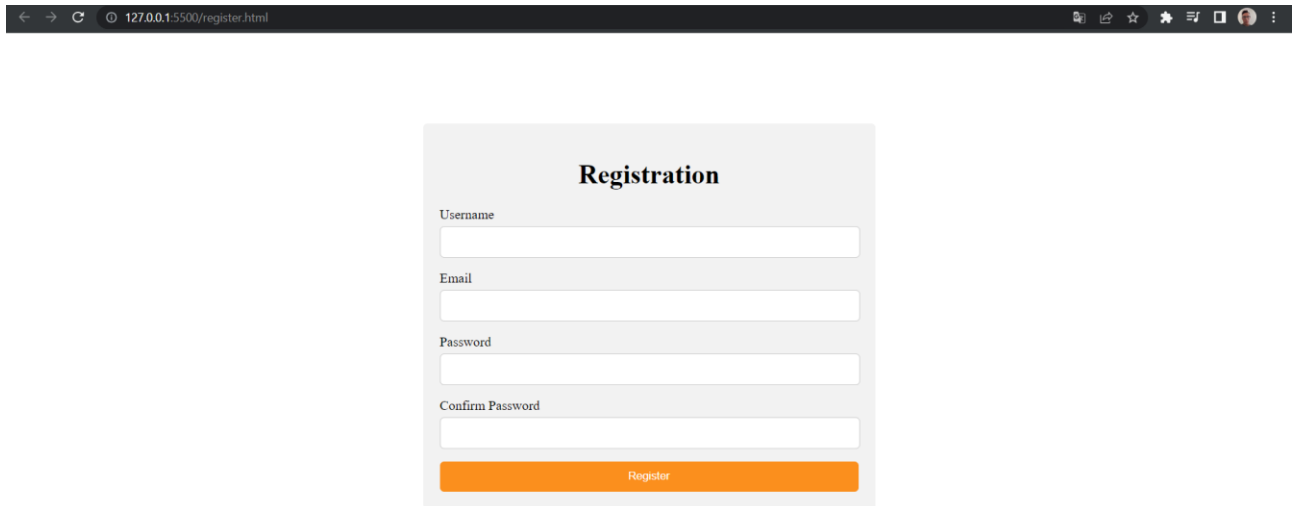


Рисунок 4.5 – Головна сторінка онлайн платформи

Далі, після того, як користувач натисне кнопку «Register», він буде перенаправлений на сторінку реєстрації акаунту (див. рис. 4.6). На цій сторінці будуть розміщені поля, де користувачу потрібно буде ввести своє ім'я, поштову адресу та пароль. Ці дані є обов'язковими для заповнення, оскільки вони необхідні для успішного створення акаунту. Ім'я дозволяє ідентифікувати користувача, поштова адреса використовується для зв'язку та надсилання сповіщень, а пароль забезпечує безпеку акаунту.



Registration

Username

Email

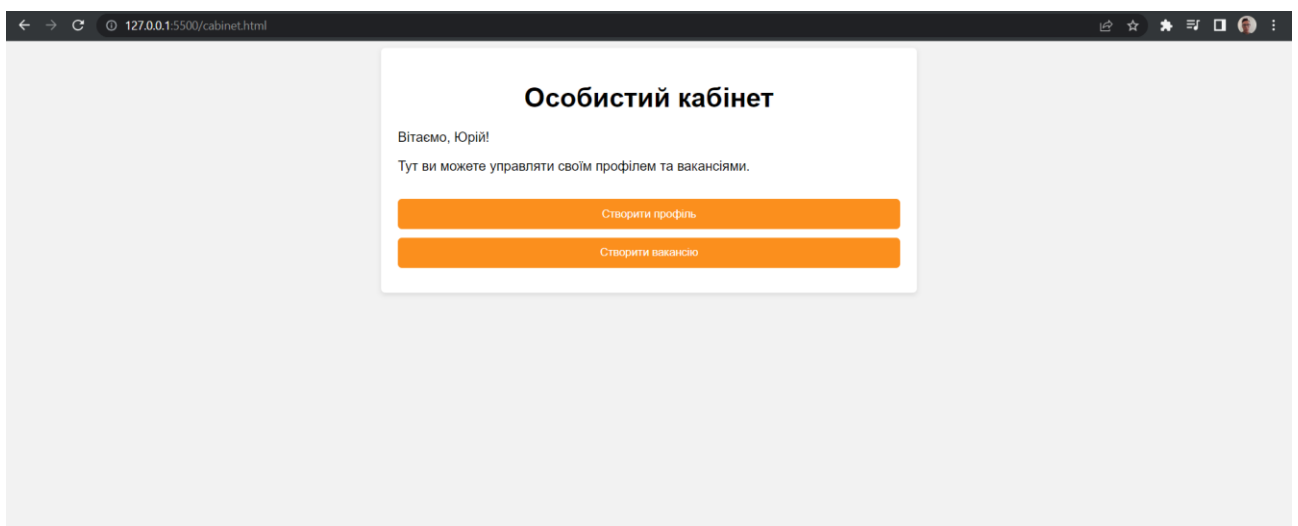
Password

Confirm Password

Register

Рисунок 4.6 – Сторінка реєстрації користувача

Після реєстрації користувач автоматично входить в свій акаунт та переходить на сторінку особистого акаунту (рис. 4.7). На цій сторінці користувачеві дається вибір між створенням вакансії та створенням профілю як працівника.



Особистий кабінет

Вітаємо, Юрій!

Тут ви можете управляти своїм профілем та вакансіями.

Створити профіль

Створити вакансію

Рисунок 4.7 – Сторінка вибору типу створюваного акаунту

Далі, натиснувши кнопку «Створити вакансію», користувача відсилає на сторінку де він заповнює дані вакансії (рис. 4.8). Обов’язковими даними є тип мистецької діяльності, бажаний досвід, місце розташування, пріоритетна характеристика та оплата. За бажанням користувач може додати детальний опис вакансії для інформування майбутніх працівників.

Створення вакансії

Тип активності:
Виберіть

Бажаний досвід (років):
5

Місце розташування:
Київ

Оплата \$:
500

Пріоритетна характеристика:
Досвід

Детальний опис вакансії:

MNews Media Group оголошує про відкриття вакансії дизайнера-ілюстратора для роботи над анімаційними відео для наших каналів на Facebook та YouTube.

«MNews» – це мережа стрімко зростаючих каналів на 7 мовах. Наші відео щомісяця набирають сотні мільйонів переглядів у всьому світі. Спієпрацюємо як з досвідченими, так і з фахівцями-початківцями.

У нас є можливість працювати як на ставці, так і відрядно!

Підтвердити

Рисунок 4.8 – Сторінка створення вакансії

Після того як користувач натисне кнопку «Підтвердити», дані про вакансію заносяться в базу даних та починається пошук профілів працівників підходящих під вказані ним характеристики.

В результаті виводиться список профілів працівників підібраний алгоритмами за допомогою вказаних користувачем критеріїв (рис. 4.9). В кожному з профілів міститься фотографія працівника, ім'я та короткий опис його переваг: тип зайнятості, досвід та місце розташування.

Для більш детального огляду працівника користувач може відкрити його профіль де міститься інформація та опис навичок, вмінь та попередніх робіт працівника. Також переглянувши профіль більш детально користувач може взнати контакти працівника для зв'язку з ним.

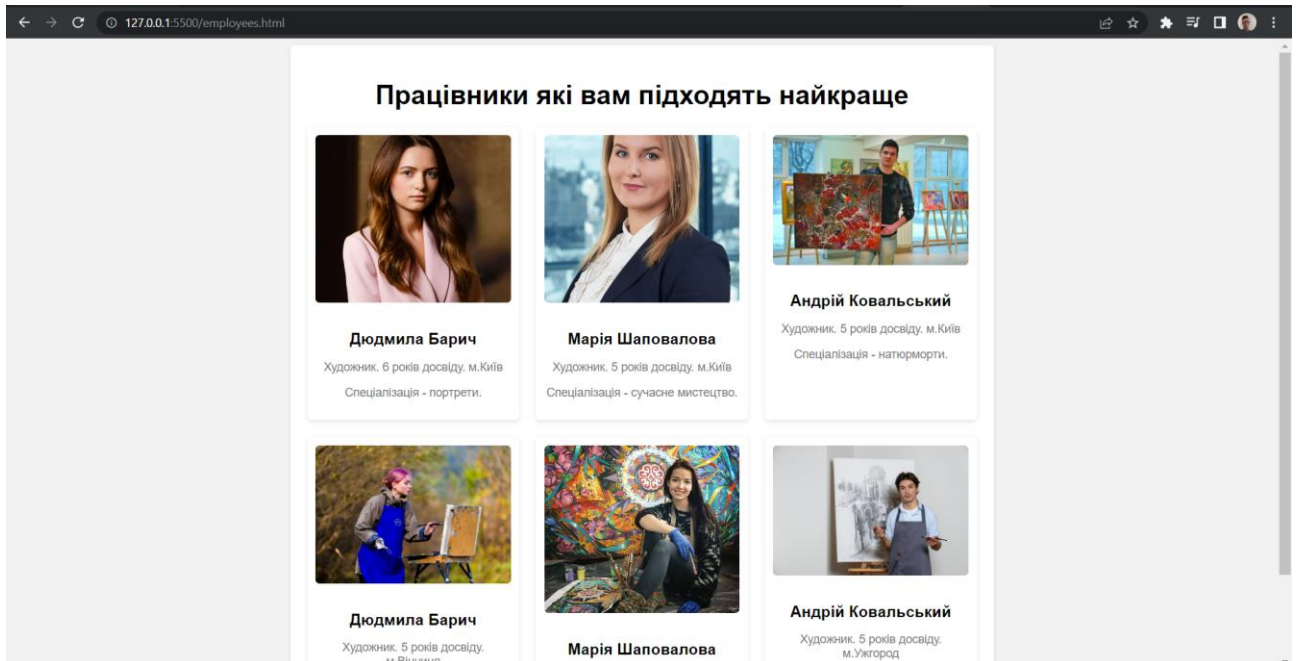


Рисунок 4.9 – Сторінка вибору працівників відносно створеної вакансії

Під час завершення розробки інструкції користувача, було покладено зусилля на забезпечення повноти та точності наданої інформації, а також урахування можливих питань та складнощів, з якими користувачі можуть стикнутися під час використання платформи. Основна мета полягає у тому, щоб користувачі могли ефективно та з комфортом використовувати онлайн платформу, отримуючи повну підтримку та необхідну інформацію, що відповідає їх потребам.

4.4 Системні вимоги

Системні вимоги для коректної роботи нашої онлайн платформи мають важливе значення для забезпечення стабільності та продуктивності. Нижче наведена таблиця 4.1 з мінімальними та оптимальними системними вимогами для користування платформою

Системні вимоги включають в себе опис апаратного та програмного забезпечення, необхідного для коректної роботи програми. Вимоги можуть

бути розподілені на мінімальні, рекомендовані та максимальні, кожні з яких вказує на певну конфігурацію системи.

Мінімальні системні вимоги вказують на найнижчу необхідну конфігурацію для запуску програми. Це означає, що без відповідного апаратного та програмного забезпечення програма не буде працювати належним чином. Рекомендовані системні вимоги, з свого боку, показують оптимальну конфігурацію, яка забезпечить максимальну продуктивність та комфортні умови роботи з програмою.

Деякі програми також надають максимальні системні вимоги, які вказують на конфігурацію, за якої програма працюватиме з повною функціональністю та надаватиме доступ до всіх її можливостей.

Особливу увагу до системних вимог приділяють відеоіграм, оскільки вони вимагають значних обчислювальних ресурсів і графічної потужності. Інформація про системні вимоги допомагає користувачам визначити, чи є їхня система сумісною з відповідною відеогрою та чи забезпечить вона належну якість та плавність геймплею.

Загальна інформація про системні вимоги є важливою для користувачів, оскільки вона дозволяє їм планувати та вибирати апаратне та програмне забезпечення відповідно до вимог програми.

Крім того, системні вимоги є також важливим критерієм при виборі та оновленні обладнання та програмного забезпечення. Користувачі, які мають на увазі використання певної програми, можуть використовувати вимоги як орієнтир для придбання або оновлення компонентів своєї системи. Це допомагає запобігти ситуації, коли програма не працює належним чином через недостатню продуктивність апаратури або несумісність зі старішою версією програмного забезпечення. Завдяки інформації про системні вимоги користувачі можуть забезпечити собі оптимальну робочу середу та мінімізувати можливі проблеми, пов'язані з несумісністю або недостатньою продуктивністю системи.

Ця інформація про системні вимоги також є корисною для розробників програмного забезпечення. Вони можуть визначити мінімальні та рекомендовані характеристики, необхідні для запуску їхньої програми, що дозволяє покращити сумісність, забезпечити стабільну роботу та надати користувачам найкращий досвід використання їхнього продукту.

Таблиця 4.1 – Системні вимоги

	Мінімальна конфігурація	Оптимальна конфігурація
Операційна система	Windows10	Windows10
	macOS Mojave (10.14)	macOS Mojave (10.14)
	Ubuntu 18.04	Ubuntu 18.04
Процесор	Intel Core i3	Intel Core i5
Оперативна пам'ять	4Гб	8Гб
Відеокарта	Вбудована відеокарта	Дискретна відеокарта з підтримкою DirectX11
Вільне місце на диску	10Гб	20Гб
Інтернет-з'єднання	5Мбіт/с	10Мбіт/с

Ці системні вимоги дозволять забезпечити високу продуктивність та якість роботи платформи на різних комп'ютерних системах. Рекомендується виконувати програму на пристроях з оптимальною конфігурацією, щоб мати найкращий досвід користування.

Отже, системні вимоги є важливим елементом при розробці, використанні та підтримці програмного забезпечення. Вони допомагають користувачам планувати та налаштовувати свої системи, а розробникам - визначати параметри для оптимальної роботи програми. Дотримання системних вимог сприяє забезпеченню сумісності, продуктивності та якості роботи програмного продукту, що в свою чергу сприяє задоволеності та задоволенню користувачів.

4.5 Аналіз результатів роботи онлайн платформи для пошуку працівників

Порівняємо функціональні можливості розробленої програми на найкращого аналога (програми Pinterest), які зведено у табл. 4.2

Таблиця 4.2 – Порівняння функціональних можливостей розробленої програми на програми-аналога

№	Функції роботи програми	Програма-аналог (Pinterest)	Розроблена програма
1.	Можливість збереження та організації контенту за категоріями	+	+
2.	Підтримка інтерактивного вмісту (зображення, відео)	+	+
3.	Можливість взаємодії через коментарі	+	+
4.	Автоматичні рекомендації на основі інтересів користувача	+	+
5.	Наявність персоналізації результатів пошуку	—	+
6.	Наявність інструментів для презентації творчого потенціалу та розміщення резюме, портфоліо артистів, художників, дизайнерів	—	+
7.	Забезпечення ефективної комунікації між роботодавцями та працівниками у сфері мистецтва.	—	+
8.	Можливості для обміну повідомленнями, детального обговорення вакансій,	—	+
9.	Забезпечення зручності відгуків та оцінювання працівників	—	+
10.	Функціонал підтримки спільнот користувачів (груп, обговорень)	—	+
11.	Можливість відеопрезентації робіт та резюме	—	+
12.	Опції для аналітики відвідуваності профілю або портфоліо	—	+
13.	Рівень адаптації для мобільних пристроїв (за 5-бальною шкалою)	4	5
14.	Ступінь інтуїтивності (за 5-бальною шкалою)	3	5
15.	Ступінь інтеграції з соціальними мережами (за 5-бальною шкалою)	3	5

Із табл. 4.2 видно, що програма-аналог має 7 функцій роботи, а розроблена програма має 15 функцій роботи. Крім того, по 3 функціям розроблена програма має вищу бальну оцінку. Це означає, що мета роботи досягнена, функціональні можливості програми розширені. Функціональність збільшена більш як на 50 % (15 функцій проти 7).

4.6 Висновок до розділу 4

У цьому розділі було проведено тестування розробленої онлайн платформи та підготовлено інструкцію користувача. Аналізуючи результати тестування, було встановлено, що платформа демонструє надійну роботу та високу функціональність, відповідаючи вимогам та очікуванням користувачів. Було виявлено, що розроблена платформа успішно пройшла тестові сценарії, проявляючи стабільну роботу та високу продуктивність. Виявлені незначні помилки та недоліки були виправлені, що позитивно позначилося на функціональності та зручності використання. Інструкція користувача, розроблена для платформи, є детальною та зрозумілою. Вона надає всю необхідну інформацію для ефективного використання платформи, сприяючи зручності та швидкому навчанню користувачів. В результаті тестування було показано, що програма-аналог має 7 функцій роботи, а розроблена програма має 15 функцій роботи, а по 3 функціях розроблена програма має вищу бальну оцінку. Тобто мета роботи досягнена, функціональні можливості програми розширені більш ніж на 50 % (15 функцій проти 7).

5 ЕКОНОМІЧНА ЧАСТИНА

5.1 Проведення комерційного та технологічного аудиту інформаційної технології для пошуку працівників.

Метою проведення комерційного і технологічного аудиту є оцінювання науково-технічного рівня та рівня комерційного потенціалу розробки, створеної в результаті науково-технічної діяльності, тобто під час виконання магістерської кваліфікаційної роботи.

Для проведення комерційного та технологічного аудиту залучаємо 3-х незалежних експертів, якими є провідні викладачі випускової або спорідненої кафедри.

Оцінювання науково-технічного рівня [23,24] розробки та її комерційного потенціалу здійснюємо із застосуванням п'ятибальної системи оцінювання за 12-ма критеріями, а результати зводимо до таблиці 5.1.

Таблиця 5.1 – Результати оцінювання науково-технічного рівня і комерційного потенціалу інформаційної технології для пошуку працівників

Критерії	Експерти		
	Експерт 1	Експерт 2	Експерт 3
	Бали, виставлені експертами		
Технічна здійсненність концепції	3	2	3
Ринкові переваги (наявність аналогів)	2	3	2
Ринкові переваги (ціна продукту)	3	2	3
Ринкові переваги (технічні властивості)	3	2	3
Ринкові переваги (експлуатаційні витрати)	2	3	3
Ринкові перспективи (розмір ринку)	3	3	2
Ринкові перспективи (конкуренція)	2	3	3
Практична здійсненність (наявність фахівців)	3	2	3
Практична здійсненність (наявність фінансів)	2	3	3

Продовження табл. 5.1

Критерії	Експерти		
	Експерт 1	Експерт 2	Експерт 3
	Бали, виставлені експертами		
Практична здійсненність (необхідність нових матеріалів)	3	3	3
Практична здійсненність (термін реалізації)	3	2	3
Практична здійсненність (розробка документів)	3	2	2
Сума балів	32	30	31
Середньоарифметична сума балів, СБ	31		

За результатами розрахунків, наведених в таблиці 5.1 робимо висновок про те, що науково-технічний рівень та комерційний потенціал [24] інформаційної технології для пошуку працівників – вище середнього.

5.2 Розрахунок витрат на здійснення науково-дослідної роботи

Витрати на оплату праці. Належать витрати на виплату основної та додаткової заробітної плати керівникам відділів, лабораторій, секторів і груп, науковим, інженерно-технічним працівникам, конструкторам, технологам, креслярам, копіювальникам, лаборантам, робітникам, студентам, аспірантам та іншим працівникам, безпосередньо зайнятим виконанням конкретної теми, обчисленої за посадовими окладами, відрядними розцінками, тарифними ставками згідно з чинними в організаціях системами оплати праці, також будь-які види грошових і матеріальних доплат, які належать до елемента «Витрати на оплату праці».

Основна заробітна плата дослідників. Витрати на основну заробітну плату [24] дослідників (З_о) розраховують відповідно до посадових окладів працівників, за формулою:

$$З_o = \sum_{i=1}^k \frac{M_{ni} \cdot t_i}{T_p},$$

де k – кількість посад дослідників, залучених до процесу дослідження; M_{ni} – місячний посадовий оклад конкретного розробника (інженера, дослідника, науковця тощо), грн.; T_p – число робочих днів в місяці; приблизно $T_p = (21 \dots 23)$ дні; t_i – число робочих днів роботи розробника (дослідника).

Зроблені розрахунки зводимо до таблиці 5.2.

Таблиця 5.2 – Витрати на заробітну плату дослідників

Посада	Місячний посадовий оклад, грн.	Оплата за робочий день, грн.	Число днів роботи	Витрати на заробітну плату, грн.
Керівник	52000	2364	12	28368
Розробник	38000	1727	25	43175
Всього:	71543			

Основна заробітна плата робітників. Витрати на основну заробітну плату робітників ($З_p$) за відповідними найменуваннями робіт розраховують за формулою:

$$З_p = \sum_{i=1}^n C_i \cdot t_i,$$

де C_i – погодинна тарифна ставка робітника відповідного розряду, за виконану відповідну роботу, грн/год; t_i – час роботи робітника на виконання певної роботи, год.

Погодинну тарифну ставку робітника відповідного розряду C_i можна визначити за формулою:

$$C_i = \frac{M_m \cdot K_i \cdot K_c}{T_p \cdot t_{зм}},$$

де M_m – розмір прожиткового мінімуму працездатної особи або мінімальної місячної заробітної плати (залежно від діючого законодавства), у 2024 році $M_m=8000$ грн; K_i – коефіцієнт міжкваліфікаційного співвідношення для встановлення тарифної ставки робітнику відповідного розряду; K_c – мінімальний коефіцієнт співвідношень місячних тарифних ставок робітників першого розряду з нормальними умовами праці виробничих об'єднань і підприємств до законодавчо встановленого розміру мінімальної заробітної плати; T_p – середня кількість робочих днів в місяці, приблизно $T_p = 21...23$ дні; $t_{зм}$ – тривалість зміни, год.

Таблиця 5.3 – Витрати на заробітну плату робітників

Найменування робіт	Трудовісткість, н-год.	Розряд роботи	Погодинна тарифна ставка	Тариф. коеф.	Величина, грн.
Аналіз предметної області	24	5	61,8	1,36	1483,2
Моделювання інформаційної технології	180	5	61,8	1,36	11124
Створення основної структури проєкту	64	6	65,9	1,45	4217,6
Створення та наповнення бази даних	44	6	65,9	1,45	2899,6
Тестування інформаційної технології	64	5	61,8	1,36	3955,2
Всього					23680

Додаткова заробітна плата. Додаткова заробітна плата Z_d всіх розробників та робітників, які брали участь у виконанні даного етапу роботи, розраховується як (10...12)% від суми основної заробітної плати всіх розробників та робітників, тобто:

$$З_д = 0,11 \cdot (З_о + З_р) = 0,11 \cdot (71543 + 23680) = 10475 \text{ грн.}$$

Відрахування на соціальні заходи. Нарахування на заробітну плату $H_{зп}$ розробників та робітників, які брали участь у виконанні даного етапу роботи, розраховуються за формулою:

$$\begin{aligned} H_{зп} &= \beta \cdot (З_о + З_р + З_д) = \\ &= 0,22 \cdot (71543 + 23680 + 10475) = 23254 \text{ грн.} \end{aligned}$$

де $З_о$ – основна заробітна плата розробників, грн.; $З_р$ – основна заробітна плата робітників, грн.; $З_д$ – додаткова заробітна плата всіх розробників та робітників, грн.; β – ставка єдиного внеску на загальнообов’язкове державне соціальне страхування, % (приймаємо для 1-го класу професійності ризику 22%).

Амортизація обладнання. Амортизація обладнання, комп’ютерів та приміщень, які використовувались під час (чи для) виконання даного етапу роботи.

У спрощеному вигляді амортизаційні відрахування A в цілому бути розраховані за формулою:

$$A = \frac{Ц_б}{T_в} \cdot \frac{t}{12},$$

де $Ц_б$ – загальна балансова вартість всього обладнання, комп’ютерів, приміщень тощо, що використовувались для виконання даного етапу роботи, грн.; t – термін використання основного фонду, місяці; $T_в$ – термін корисного використання основного фонду, роки.

Таблиця 5.4 – Амортизаційні відрахування за видами основних фондів

Найменування	Балансова вартість, грн.	Строк корисного використання, років	Термін використання, місяців	Сума амортизації, грн.
Ноутбук	38000	3	1,5	1583
Стіл	7000	5	1,5	175
Стілець	3000	5	1,5	75
Мишка	1000	3	1,5	42
Клавіатура	1100	3	1,5	46
Настільна лампа	1200	5	1,5	30
Всього	1951			

Витрати на електроенергію для науково-виробничих цілей. Витрати на силову електроенергію B_e , якщо ця стаття має суттєве значення для виконання даного етапу роботи, розраховуються за формулою:

Таблиця 5.5 – Витрати на електроенергію

Найменування обладнання	Потужність, кВт	Тривалість годин роботи
Ноутбук	0,24	200
Освітлення	0,04	96

$$B_e = \sum \frac{W_i \cdot t_i \cdot C_e \cdot K_{\text{впі}}}{\text{ККД}} = \frac{0,24 \cdot 200 \cdot 7,5 \cdot 0,75}{0,78} + \frac{0,04 \cdot 96 \cdot 7,5 \cdot 0,75}{0,78} = 374 \text{ грн.},$$

W_i – встановлена потужність обладнання, кВт; t_i – тривалість роботи обладнання на етапі дослідження, год.; C_e – вартість 1 кВт електроенергії, грн.; $K_{\text{впі}}$ – коефіцієнт використання потужності; ККД – коефіцієнт корисної дії обладнання.

Інші витрати. До статті «Інші витрати» належать витрати, які не знайшли відображення у зазначених статтях витрат і можуть бути віднесені безпосередньо на собівартість досліджень за прямими ознаками.

Витрати за статтею «Інші витрати» розраховуються як 50...100% від суми основної заробітної плати дослідників та робітників за формулою:

$$I_{\text{в}} = (Z_o + Z_p) \cdot \frac{N_{\text{ів}}}{100\%} = (71543 + 23680) \cdot \frac{50}{100} = 47612 \text{ грн.},$$

де $N_{\text{ів}}$ – норма нарахування за статтею «Інші витрати».

Накладні (загальновиробничі) витрати. До статті «Накладні (загальновиробничі) витрати» належать: витрати, пов'язані з управлінням організацією; витрати на винахідництво та раціоналізацію; витрати на підготовку (перепідготовку) та навчання кадрів; витрати, пов'язані з набором робочої сили; витрати на оплату послуг банків; витрати, пов'язані з освоєнням виробництва продукції; витрати на науково-технічну інформацію та рекламу та ін.

Витрати за статтею «Накладні (загальновиробничі) витрати» розраховуються як 100...150% від суми основної заробітної плати дослідників та робітників за формулою:

$$V_{\text{нзв}} = (Z_o + Z_p) \cdot \frac{N_{\text{нзв}}}{100\%} = (71543 + 23680) \cdot \frac{115}{100} = 109506 \text{ грн.},$$

де $N_{\text{нзв}}$ – норма нарахування за статтею «Накладні (загальновиробничі) витрати».

Витрати на проведення науково-дослідної роботи. Витрати на проведення науково-дослідної роботи розраховуються як сума всіх попередніх статей витрат за формулою:

$$\begin{aligned} B_{\text{заг}} &= Z_o + Z_p + Z_{\text{дод}} + Z_n + K_v + B_{\text{спец}} + B_{\text{прг}} + A_{\text{обл}} + B_e + \\ &\quad + I_v + B_{\text{нзв}} = \\ &= 71543 + 23680 + 10475 + 23254 + 1951 + 374 + 47612 + \\ &\quad 109506 = 178889 \text{ грн.} \end{aligned}$$

Загальні витрати. Загальні витрати ЗВ на завершення науково-дослідної (науково-технічної) роботи та оформлення її результатів розраховуються за формулою:

$$ЗВ = \frac{B_{\text{заг}}}{\eta} = \frac{178889}{0,9} = 198766 \text{ грн.,}$$

де η – коефіцієнт, що характеризує етап виконання науково-дослідної роботи. Оскільки, якщо науково-технічна розробка знаходиться на стадії впровадження, то $\eta=0,9$.

5.3 Розрахунок економічної ефективності науково-технічної розробки інформаційної технології для пошуку працівників за її можливої комерціалізації потенційним інвестором

В ринкових умовах узагальнюючим позитивним результатом, що його може отримати потенційний інвестор від можливого впровадження результатів тієї чи іншої науково-технічної розробки, є збільшення у потенційного інвестора величини чистого прибутку.

В даному випадку відбувається розробка засобу, тому основу майбутнього економічного ефекту буде формувати: ΔN – збільшення

кількості споживачів, яким надається відповідна інформаційна послуга в аналізовані періоди часу; N – кількість споживачів, яким надавалась відповідна інформаційна послуга у році до впровадження результатів нової науково-технічної розробки; Π_0 – вартість послуги у році до впровадження інформаційної системи; $\pm\Delta\Pi_0$ – зміна вартості послуги (зростання чи зниження) від впровадження результатів науково-технічної розробки в аналізовані періоди часу.

Можливе збільшення чистого прибутку у потенційного інвестора $\Delta\Pi$ для кожного із років, протягом яких очікується отримання позитивних результатів від можливого впровадження та комерціалізації науково-технічної розробки, розраховується за формулою:

$$\Delta\Pi = (\pm\Delta\Pi_0 \cdot N + \Pi_0 \cdot \Delta N_i)_i \cdot \lambda \cdot \rho \cdot \left(1 - \frac{\vartheta}{100}\right),$$

де $\pm\Delta\Pi$ – зміна основного якісного показника від впровадження результатів науково-технічної розробки в аналізованому році. Зазвичай, таким показником може бути зміна ціни реалізації одиниці нової розробки в аналізованому році (відносно року до впровадження цієї розробки); $\pm\Delta\Pi_0$ може мати як додатне, так і від'ємне значення (від'ємне – при зниженні ціни відносно року до впровадження цієї розробки, додатне – при зростанні ціни); N – основний кількісний показник, який визначає величину попиту на аналогічні чи подібні розробки у році до впровадження результатів нової науково-технічної розробки; Π_0 – основний якісний показник, який визначає ціну реалізації нової науково-технічної розробки в аналізованому році; Π_0 – основний якісний показник, який визначає ціну реалізації існуючої (базової) науково-технічної розробки у році до впровадження результатів; ΔN – зміна основного кількісного показника від впровадження результатів науково-технічної розробки в аналізованому році. Зазвичай таким показником може бути зростання попиту на науково-технічну розробку в аналізованому році

(відносно року до впровадження цієї розробки); λ – коефіцієнт, який враховує сплату потенційним інвестором податку на додану вартість. У 2024 році ставка податку на додану вартість становить 20%, а коефіцієнт $\lambda = 0,8333$; ρ – коефіцієнт, який враховує рентабельність інноваційного продукту (послуги). Рекомендується брати $\rho = 0,2 \dots 0,5$; θ – ставка податку на прибуток, який має сплачувати потенційний інвестор, у 2024 році $\theta = 18\%$.

$$\Delta\P = ((1700 - 1350) \cdot 620 + 1700 \cdot (700 - 620)) \cdot 0,8333 \cdot 0,5 \cdot \left(1 - \frac{18}{100}\right) = 120126 \text{ грн.}$$

Далі розраховують приведену вартість збільшення всіх чистих прибутків ПП, що їх може отримати потенційний інвестор від можливого впровадження та комерціалізації науково-технічної розробки:

$$\begin{aligned} \text{ПП} &= \sum_{i=1}^T \frac{\Delta\P_i}{(1 + \tau)^t} = \frac{120126}{(1 + 0,1)^1} + \frac{120126}{(1 + 0,1)^2} + \frac{120126}{(1 + 0,1)^3} + \frac{120126}{(1 + 0,1)^4} \\ &= 468728 \text{ грн.,} \end{aligned}$$

де $\Delta\P$ – збільшення чистого прибутку у кожному з років, протягом яких виявляються результати впровадження науково-технічної розробки, грн.; T – період часу, протягом якого очікується отримання позитивних результатів від впровадження та комерціалізації науково-технічної розробки, роки (приймаємо $T=4$ роки); τ – ставка дисконтування, за яку можна взяти щорічний прогнозований рівень інфляції в країні, $\tau = 0,05 \dots 0,15$; t – період часу (в роках) від моменту початку впровадження науково-технічної розробки до моменту отримання потенційним інвестором додаткових чистих прибутків у цьому році.

Далі розраховують величину початкових інвестицій PV , які потенційний інвестор має вкласти для впровадження і комерціалізації науково-технічної розробки інформаційної технології для пошуку працівників. Для цього можна використати формулу:

$$PV = k_{\text{інв}} \cdot 3B = 2 \cdot 198766 = 397532 \text{ грн.}$$

де $k_{\text{інв}}$ – коефіцієнт, що враховує витрати інвестора на впровадження науково-технічної розробки та її комерціалізацію. Це можуть бути витрати на підготовку приміщень, розробку технологій, навчання персоналу, маркетингові заходи тощо; зазвичай $k_{\text{інв}}=2\dots5$, але може бути і більшим; $3B$ – загальні витрати на проведення науково-технічної розробки та оформлення її результатів, грн.

Тоді абсолютний економічний ефект $E_{\text{абс}}$ або чистий приведений дохід для потенційного інвестора від можливого впровадження та комерціалізації науково-технічної розробки інформаційної технології для пошуку працівників становитиме:

$$E_{\text{абс}} = \text{ПП} - PV = 468728 - 397532 = 71196 \text{ грн.,}$$

де ПП – приведена вартість зростання всіх чистих прибутків від можливого впровадження та комерціалізації науково-технічної розробки, грн.; PV – теперішня вартість початкових інвестицій, грн.

Оскільки $E_{\text{абс}} > 0$, то можемо припустити про потенційну зацікавленість інвесторів у розробці інформаційної технології для пошуку працівників.

Для остаточного прийняття рішення з цього питання необхідно розрахувати внутрішню економічну дохідність E_v або показник внутрішньої норми дохідності вкладених інвестицій та порівняти її з так званою бар'єрною ставкою дисконтування, яка визначає ту мінімальну внутрішню економічну дохідність, нижче якої інвестиції в будь-яку науково-технічну розробку вкладати буде економічно недоцільно.

Внутрішня економічна дохідність інвестицій E_v , які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію

науково-технічної розробки інформаційної технології для пошуку працівників, розраховується за формулою:

$$E_B = \sqrt[T_{\text{ж}}]{1 + \frac{E_{\text{абс}}}{PV}} - 1 = \sqrt[4]{1 + \frac{71196}{397532}} - 1 = 0,42,$$

де $T_{\text{ж}}$ – життєвий цикл розробки інформаційної технології для пошуку працівників, роки.

Визначимо бар'єрну ставку дисконтування $\tau_{\text{мін}}$, тобто мінімальну внутрішню економічну дохідність інвестицій, нижче якої кошти у впровадження науково-технічної розробки та її комерціалізацію вкладатися не будуть.

Мінімальна внутрішня економічна дохідність вкладених інвестицій $\tau_{\text{мін}}$ визначається за формулою:

$$\tau_{\text{мін}} = d + f = 0,12 + 0,05 = 0,17,$$

де d – середньозважена ставка за депозитними операціями в комерційних банках; в 2024 році в Україні $d = 0,9...0,12$; f – показник, що характеризує ризикованість вкладення інвестицій; зазвичай величина $f = 0,05...0,5$, але може бути і значно вищою.

Оскільки $E_B = 0,42 > \tau_{\text{мін}} = 0,17$, то потенційний інвестор може бути зацікавлений у фінансуванні впровадження науково-технічної розробки та виведенні її на ринок, тобто в її комерціалізації.

Далі розраховуємо період окупності інвестицій T_o , які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-технічної розробки інформаційної технології для пошуку працівників:

$$T_o = \frac{1}{E_v} = \frac{1}{0,42} = 2,4 \text{ роки.}$$

Оскільки $T_o < 1 \dots 3$ -х років, то це свідчить про комерційну привабливість науково-технічної розробки і може спонукати потенційного інвестора профінансувати впровадження цієї розробки інформаційної технології для пошуку працівників та виведення її на ринок.

5.4 Висновок до розділу 5

Згідно проведених досліджень рівень комерційного потенціалу розробки за темою «Інформаційна технологія для пошуку працівників» становить 31,0 балів, що вказує на комерційну вагомість проведення даних досліджень (рівень комерційного потенціалу розробки середній). Термін окупності становить 2,4 р., що менше 3-х років, що говорить про комерційну привабливість науково-технічної розробки і може спонукати потенційних інвесторів на фінансування впровадження цієї розробки та виведення її на ринок. Отже можна зробити висновок про доцільність проведення науково-дослідної роботи за темою «Інформаційна технологія для пошуку працівників».

ВИСНОВКИ

У роботі було розглянуто задачу пошуку працівників на основі технологій React. В ході аналізу предметної області було розглянуто стан платформ для пошуку роботи, проведено порівняльний аналіз аналогів, методів розв'язання задачі та постановку задач для розробки інформаційної технології пошуку працівників сфери мистецтва. Було з'ясовано, що в даний час існує певна кількість платформ для пошуку роботи, таких як LinkedIn, Behance, ArtStation та Pinterest. Вони надають можливості для презентації творчої роботи та знаходження робочих можливостей, проте кожна з них має свої переваги та обмеження. Аналіз методів розв'язання задачі показав, що використання React технології дозволить створити ефективний та зручний інтерфейс користувача, забезпечити масштабованість та високу продуктивність системи. Поставлені задачі для розробки програми пошуку працівників сфери мистецтва, що включають розробку системи авторизації та профілю користувача, системи пошуку працівників, функціоналу оцінки та відгуків, системи спілкування та обміну повідомленнями, а також розробку інтерфейсу користувача.

Було проведено розробку структури та алгоритмів інформаційної технології для пошуку працівників сфери мистецтва. В процесі роботи було здійснено аналіз даних, розроблено структуру інтерфейсу програмного додатку, розроблено модель системи та розроблено алгоритми роботи пошукової програми. Аналіз даних виявив ключові параметри та характеристики, необхідні для ефективного пошуку та відбору працівників сфери мистецтва. Це включає професійні навички, досвід роботи, географічне розташування та інші фактори, які впливають на відповідність працівника вимогам роботодавців. Структура інтерфейсу програмного додатку була розроблена з урахуванням зручності користувачів та логіки роботи системи. Були визначені необхідні елементи, взаємозв'язки між ними та способи взаємодії користувачів з програмою. Модель системи була створена,

враховуючи особливості сфери мистецтва та потреб користувачів. Вона передбачає взаємодію між роботодавцями та працівниками, можливість створення профілів, пошук робочих місць та відповідність вимогам. Алгоритми роботи інформаційної технології були розроблені з урахуванням ефективності та точності пошуку працівників. Вони використовують різноманітні фільтри та критерії для відбору найбільш відповідних працівників за встановленими параметрами. Було розроблено модель роботи програмного продукту за допомогою UML діаграм.

Під час розробки програмних компонентів онлайн платформи для пошуку працівників сфери мистецтва було проведено аналіз даних, розроблено структуру інтерфейсу, модель системи та алгоритми роботи пошукової програми. Варіантний аналіз мов програмування підтвердив обґрунтування вибору мови Python для платформи пошуку працівників та React для інтерфейсу платформи. Вибір середовища розробки був здійснений на користь Visual Studio Code, яке забезпечує зручне та продуктивне середовище для розробки програмних компонентів. Програмна реалізація додатку забезпечує ефективне порівняння параметрів вакансій та профілів користувачів. У результаті розробки програмних компонентів вдалося створити функціональну та ефективну онлайн платформу для пошуку працівників сфери мистецтва. Вона надає зручний інтерфейс для користувачів, забезпечує швидке та точне порівняння профілів, а також високу швидкодію завдяки використанню мови програмування Python. В цілому, розробка програмних компонентів відповідає поставленим завданням та вимогам і може служити цінним інструментом для пошуку талановитих працівників у сфері мистецтва.

Було проведено тестування розробленої онлайн платформи та підготовлено інструкцію користувача. Аналізуючи результати тестування, було встановлено, що платформа демонструє надійну роботу та високу функціональність, відповідаючи вимогам та очікуванням користувачів. Було виявлено, що розроблена платформа успішно пройшла тестові сценарії,

проявляючи стабільну роботу та високу продуктивність. Виявлені незначні помилки та недоліки були виправлені, що позитивно позначилося на функціональності та зручності використання. Інструкція користувача, розроблена для платформи, є детальною та зрозумілою. Вона надає всю необхідну інформацію для ефективного використання платформи, сприяючи зручності та швидкому навчанню користувачів. В результаті тестування було показано, що програма-аналог має 7 функцій роботи, а розроблена програма має 15 функцій роботи, а по 3 функціях розроблена програма має вищу бальну оцінку. Тобто мета роботи досягнена, функціональні можливості програми розширені більш ніж на 50 % (15 функцій проти 7).

Згідно проведених досліджень рівень комерційного потенціалу розробки за темою «Інформаційна технологія для пошуку працівників» становить 31,0 балів, що вказує на комерційну вагомість проведення даних досліджень (рівень комерційного потенціалу розробки середній). Термін окупності становить 2,4 р., що менше 3-х років, що говорить про комерційну привабливість науково-технічної розробки і може спонукати потенційних інвесторів на фінансування впровадження цієї розробки та виведення її на ринок.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Ю. В. Цимборович, О. К. Колесницький Інформаційна технологія для пошуку роботи / в Матеріали конференції «Молодь в науці: дослідження, проблеми, перспективи (МН-2024)», Вінниця, 2023, [Електронний ресурс]. Режим доступу:
<https://conferences.vntu.edu.ua/index.php/mn/mn2025/paper/viewFile/22897/18877>
2. Онлайн платформи для пошуку роботи URL:
<https://ukrainian.city/robota/kanaly-poshuku>
3. Science URL: <https://science.lpnu.ua/uk/sisn/vsi-vypusky/vypusk-901-2018/rozroblennya-systemy-poshuku-vakansiy>
4. Flavio, THE REACT HANDBOOK, Нью-Йорк, США : Flavio Copes Publishing, 2022. 215 с.
5. Linkedin URL: <https://www.linkedin.com/>
6. Behance URL: <https://www.behance.net/>
7. ArtStation URL:
https://www.artstation.com/?sort_by=community&dimension=all
8. Jeffrey D. Ullman, Jennifer Widom, DATABASE SYSTEMS The Complete Book 2nd ed. New Jersey, США, 2020. 318с.
9. Wikipedia URL:
https://uk.wikipedia.org/wiki/%D0%9C%D0%B0%D1%88%D0%B8%D0%BD%D0%BD%D0%B5_%D0%BD%D0%B0%D0%B2%D1%87%D0%B0%D0%BD%D0%BD%D1%8F
10. Метод впровадження API. URL: <https://dev.ua/news/valid-article-1662018442-1662018442>
11. Julia V. The UX Book Process and Guidelines for Ensuring a Quality User Experience Сіетл, США : Amazon Publishing, 2022. 274с.
12. Wikipedia URL:
https://uk.wikipedia.org/wiki/Unified_Modeling_Language
13. Peter P. A Complete Guide to Programming in C++ Массачусетс, США: Jones and Bartlett Publishers, 2020. 418 с.

14. Laura L. Teach yourself Java Індіана, США : Sams Publishing, 2021. 361с.
15. Bulgaria developers, Fundamentals of Computer Programming with C# URL: <https://introprogramming.info/english-intro-csharp-book/read-online/>
16. Python The Big Book of Small Python Projects URL: <https://inventwithpython.com/bigbookpython/>
17. Flavio, The JS Handbook, Нью-Йорк, США : Flavio Copes Publishing, 2022. 55 с.
18. PyCharm URL: <https://www.jetbrains.com/pycharm/>
19. Jupyter Notebook URL: <https://jupyter.org/>
20. Visual Studio Code URL: <https://code.visualstudio.com/>
21. Spyder URL: <https://www.spyder-ide.org/>
22. O'Regan Gerard. Concise Guide to Software Testing. Хам, Швейцарія : Springer International Publishing, 2019. 293 с.
23. Методичні вказівки до виконання економічної частини магістерських кваліфікаційних робіт / Уклад. : В. О. Козловський, О. Й. Лесько, В. В. Кавецький. – Вінниця : ВНТУ, 2021. – 42 с.
24. Кавецький В. В. Економічне обґрунтування інноваційних рішень: практикум / В. В. Кавецький, В. О. Козловський, І. В. Причепа – Вінниця : ВНТУ, 2016. – 113 с.
25. Методичні вказівки до виконання магістерських кваліфікаційних робіт для студентів спеціальності 122 «Комп'ютерні науки» [Електронний ресурс] / уклад.: А. А. Яровий, О. К. Колесницький. – Вінниця : ВНТУ, 2023. – (58 с.)

Додаток А (обов'язковий)
ПРОТОКОЛ ПЕРЕВІРКИ КВАЛІФІКАЦІЙНОЇ РОБОТИ НА
НАЯВНІСТЬ ТЕКСТОВИХ ЗАПОЗИЧЕНЬ

Назва роботи: Інформаційна технологія для пошуку працівників

Тип роботи: магістерська кваліфікаційна робота
(БДР, МКР)

Підрозділ кафедра комп'ютерних наук, ФІТА
(кафедра, факультет)

Показники звіту подібності Turnitin

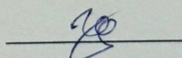
Оригінальність 96,00% Схожість 4,00%

Аналіз звіту подібності (відмітити потрібне):

- ☒ Запозичення, виявлені у роботі, оформлені коректно і не містять ознак плагіату.
- ☐ Виявлені у роботі запозичення не мають ознак плагіату, але їх надмірна кількість викликає сумніви щодо цінності роботи і відсутності самостійності її виконання автором. Роботу направити на розгляд експертної комісії кафедри.
- ☐ Виявлені у роботі запозичення є недобросовісними і мають ознаки плагіату та/або в ній містяться навмисні спотворення тексту, що вказують на спроби приховування недобросовісних запозичень.

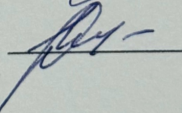
Ознайомлені з повним звітом подібності, який був згенерований системою Turnitin щодо роботи.

Автор роботи



Цимборович Ю.В.

Керівник роботи

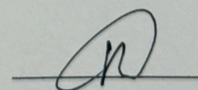


Колесницький О.К..

Опис прийнятого рішення

Магістерську кваліфікаційну роботу допущено до захисту

Особа, відповідальна за перевірку



Озеранський В.С.

Додаток Б (обов'язковий)

Лістинг програми

Main HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link rel="stylesheet" href="/css/style.css">
  <title>JobFinder</title>
</head>
<body>
  <header>
    <div class="container1">
      <div class="burger">
        
        <nav class="navigation">
          <div class="navigation__item">Home</div>
          <div class="navigation__item">Destinations</div>
          <div class="navigation__item">About</div>
          <div class="navigation__item">Partner</div>
          <div class="navigation__item loginform">
            <a href="#" class="navigation__login nvlink">Login</a>
          </div>
          <div class="navigation__item loginform">
            <a target="_blank" href="register.html" class="navigation__register nvlink">Register</a>
          </div>
        </nav>
      </div>
    </div>
  </header>
  <div class="maincontent">
    <div class="container2">
      <div class="explore">
        <div class="column">
          <h2 class="explore__title">Jobfinder for artists</h2>
          <h3 class="explore__subtitle">Monetize your hobby</h3>
          <div class="explore__border"></div>
          <div class="explore__selectionform">
            <form action="#" class="dataform">
              <select type="text" class="explore__listbox location">
                <option value="Location">Location</option>
                <option value="Kyiv">Kyiv</option>
                <option value="Vinnytsia">Vinnytsia</option>
                <option value="Lviv">Lviv</option>
                <option value="Makiivka">Rudnichok</option>
                <option value="KvartiraPapicha">Kv. Papicha</option>
              </select>
              <select type="text" class="explore__listbox activity">
                <option value="Activity">Activity</option>
                <option value="Painting">Painting</option>
                <option value="Sculpture">Sculpture</option>
                <option value="Design">Design</option>
                <option value="Music">Music</option>
              </select>
              <select type="text" class="explore__listbox grade">
                <option value="grade1">Experience</option>
                <option value="grade2">1 year</option>
                <option value="grade3">2 years</option>
                <option value="grade4">3 years</option>
                <option value="grade5">4 years</option>
                <option value="grade6">5 years</option>
              </select>
              <select type="text" class="explore__listbox date">
                <option value="Date">Rating</option>
                <option value="firstdate">1</option>
                <option value="secondDate">2</option>
                <option value="thirddate">3</option>
                <option value="fourthdate">4</option>
                <option value="fifthdate">5</option>
              </select>
            </form>
          </div>
        </div>
      </div>
    </div>
  </div>
</body>
</html>
```

```

    </div>
    <div class="explore__btn">Find</div>
  </div>
  
</div>
</div>
<div class="container3">
  <div class="about">
    
    <div class="content">
      <div class="about__title">
        A new way to find a job
      </div>
      <br>
      <div class="about__text">
        For decades, people in the arts have been looking for the most promising and high-paying jobs, but there hasn't been a platform
        aimed exclusively at this field, but now they can always turn to us in search of the perfect job.
      </div>
      <div class="about__btn">Learn more</div>
    </div>
  </div>
</div>
<div class="container4">
  <div class="features">
    <div class="toprow">
      <div class="features__title">
        Popular destinations
      </div>
      <div class="viewbtn">
        <div class="features__viewtext">View all</div>
        
      </div>
    </div>
    <div class="bottomrow">
      <div class="features__block">
        
        <div class="featabout">
          <div class="featabout__title">
            Painting
          </div>
          <div class="featabout__place">
            portrait
          </div>
        </div>
      </div>
      <div class="features__block">
        
        <div class="featabout">
          <div class="featabout__title">
            Sculpture
          </div>
          <div class="featabout__place">
            human body
          </div>
        </div>
      </div>
      <div class="features__block">
        
        <div class="featabout">
          <div class="featabout__title">
            Music
          </div>
          <div class="featabout__place">
            Classic
          </div>
        </div>
      </div>
      <div class="features__block">
        
        <div class="featabout">
          <div class="featabout__title">
            Photography
          </div>
          <div class="featabout__place">
            Professional
          </div>
        </div>
      </div>
    </div>
  </div>
</div>

```

```

    </div>
  </div>
  <div class="container5">
    <div class="guides">
      <div class="guicontent">
        <div class="guides__title">
          Work abroad
        </div>
        <div class="guides__text">
          Assistance in finding a job abroad, consultations and advice on going abroad, obtaining a visa, finding a place to live,
etc.
        </div>
        <div class="guides__btn">Check</div>
      </div>
      
    </div>
  </div>
  <div class="container6">
    <div class="test">
      <div class="test__title">
        Testimonials
      </div>
      <div class="slider">
        <div class="slide fade">
          <div class="slidecontent">
            <div class="slide__mark"></div>
            <div class="slide__text">
              "Quisque in lacus a urna fermentum euismod. Integer mi nibh, dapibus ac scelerisque eu, facilisis quis purus. Morbi blandit
sit amet turpis nec"
            </div>
            <div class="slide__name">Edward Newgate</div>
            <div class="slide__position">Founder Circle</div>
          </div>
          <div class="slideimg">
            
            
            
          </div>
        </div>
        <div class="slide fade">
          <div class="slidecontent">
            <div class="slide__mark"></div>
            <div class="slide__text">
              "Lorem ipsum, dolor sit amet consectetur adipisicing elit. Quia aliquam voluptatibus esse eum, veniam soluta quisquam
voluptatem veritatis sequi"
            </div>
            <div class="slide__name">Den Radchenko</div>
            <div class="slide__position">Manager</div>
          </div>
          <div class="slideimg">
            
            
            
          </div>
        </div>
        <div class="slide fade">
          <div class="slidecontent">
            <div class="slide__mark"></div>
            <div class="slide__text">
              "Quisque in lacus a urna fermentum euismod. Integer mi nibh, dapibus ac scelerisque eu, facilisis quis purus. Morbi blandit
sit amet turpis nec"
            </div>
            <div class="slide__name">Alex Kuchevskyi</div>
            <div class="slide__position">Middle Developer</div>
          </div>
          <div class="slideimg">
            
            
            
          </div>
        </div>
      </div>
      <div class="arrows">
        <a class="arrows__prev" onclick="plusSlides(-1)">&#10094</a>
        <a class="arrows__next" onclick="plusSlides(1)">&#10095</a>
      </div>
    </div>
  </div>
  </div>
  <script src="/js/script.js"></script>
</body>

```

```
</html>
```

Register

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Registration</title>
  <link rel="stylesheet" href="css/styleregister.css">
</head>
<body>
  <div class="container">
    <h1>Registration</h1>
    <form>
      <div class="form-group">
        <label for="username">Username</label>
        <input type="text" id="username" name="username" required>
      </div>
      <div class="form-group">
        <label for="email">Email</label>
        <input type="email" id="email" name="email" required>
      </div>
      <div class="form-group">
        <label for="password">Password</label>
        <input type="password" id="password" name="password" required>
      </div>
      <div class="form-group">
        <label for="confirm-password">Confirm Password</label>
        <input type="password" id="confirm-password" name="confirm-password" required>
      </div>
      <button type="submit"><a href="cabinet.html">Register</a></button>
    </form>
  </div>
</body>
</html>
```

Vacancy

```
<!DOCTYPE html>
<html>
<head>
  <title>Створення вакансії</title>
  <link rel="stylesheet" type="text/css" href="css/stylevacancy.css">
</head>
<body>
  <div class="container">
    <h1>Створення вакансії</h1>
    <form>
      <label for="activity-type">Тип активності:</label>
      <select id="activity-type" name="activity-type">
        <option value="option1">Виберіть</option>
        <option value="option2">Painting</option>
        <option value="option3">Sculpture</option>
        <option value="option3">Music</option>
        <option value="option3">Design</option>
      </select>

      <label for="desired-experience">Бажаний досвід (років):</label>
      <input type="text" id="desired-experience" name="desired-experience">

      <label for="location">Місце розташування:</label>
      <input type="text" id="location" name="location">

      <label for="payment">Оплата $:</label>
      <input type="text" id="payment" name="payment">

      <label for="activity-type">Пріоритетна характеристика:</label>
      <select id="activity-type" name="activity-type">
        <option value="option1">Виберіть</option>
        <option value="option2">Місце розташування</option>
        <option value="option3">Досвід</option>
        <option value="option3">Рейтинг</option>
      </select>

      <label for="job-description">Детальний опис вакансії:</label>
      <textarea id="job-description" name="job-description"></textarea>
```

```

    <button class="submit-btn"><a href="employees.html">Підтвердити</a></button>
  </form>
</div>
</body>
</html>

```

Employees

```

<!DOCTYPE html>
<html>
<head>
<title>Працівники які вам підходять найкраще</title>
<link rel="stylesheet" type="text/css" href="css/styleemployees.css">
</head>
<body>
<div class="container">
  <h1>Працівники які вам підходять найкраще</h1>
  <div class="grid">
    <div class="worker-card">
      
      <h3>Дюдмила Барич</h3>
      <p>Художник. 6 років досвіду. м.Київ</p>
      <p>Спеціалізація - портрети.</p>
    </div>
    <div class="worker-card">
      
      <h3>Марія Шаповалова</h3>
      <p>Художник. 5 років досвіду. м.Київ</p>
      <p>Спеціалізація - сучасне мистецтво.</p>
    </div>
    <div class="worker-card">
      
      <h3>Андрій Ковальський</h3>
      <p>Художник. 5 років досвіду. м.Київ</p>
      <p>Спеціалізація - натюрморти.</p>
    </div>
    <div class="worker-card">
      
      <h3>Дюдмила Барич</h3>
      <p>Художник. 5 років досвіду. м.Вінниця</p>
      <p>Спеціалізація - живопис.</p>
    </div>
    <div class="worker-card">
      
      <h3>Марія Шаповалова</h3>
      <p>Художник. 5 років досвіду. м.Бердичів</p>
      <p>Спеціалізація - стріт арт.</p>
    </div>
    <div class="worker-card">
      
      <h3>Андрій Ковальський</h3>
      <p>Художник. 5 років досвіду. м.Ужгород</p>
      <p>Спеціалізація - пейзаж.</p>
    </div>
  </div>
</div>
</body>
</html>

```

Main SCSS

```

// Null Style File
@import "_nullstyle.scss";

//Fonts
@import url('https://fonts.googleapis.com/css2?family=Inter:wght@400;500;600&display=swap');
@import url('https://fonts.googleapis.com/css2?family=Playfair+Display:wght@700&display=swap');
@import url('https://fonts.googleapis.com/css2?family=Mulish:wght@300;400;700&display=swap');

//Main Code
.container1{
  max-width: 1250px;
  margin: 40px auto 40px auto;
}

.burger {

```

```

display: flex;
justify-content: space-between;
}
.logo {
width: 41px;
height: 41px;
}

.navigation {
display: flex;
align-self: center;
font-family: 'Inter', sans-serif;
font-size: 16px;
color: #B8BECD;
height: 44px;

&__item {
margin-right: 35px;
align-self: center;

&:hover{
color: #000;
font-weight: 500;
cursor: pointer;
}

&.loginform{
display: flex;
width: 132px;
height: 100%;
border: 1px solid #FB8F1D;
border-radius: 8px;
text-align: center;
margin-right: 18px;
&:last-child{
margin-right: 0px;
}
&:hover {
background-color: #FB8F1D;
}
&:hover .nmlink{
color: #fff;
}
}
}

&__login, &__register {
align-self: center;
justify-self: center;
width: 100%;
color: #FB8F1D;
}
}

.maincontent {

//first block
.container2 {
max-width: 1305px;
margin: 40px auto 40px auto;
}

.column {
max-width: 480px;
}

.explore {

display: flex;
justify-content: space-between;

&__title {
font-family: 'Playfair Display', serif;
font-size: 64px;
line-height: 80px;
padding: 33px 0 75px 0;

```

```

}

&__subtitle {
  font-family: 'Inter', sans-serif;
  font-size: 22px;
  line-height: 26px;
  font-weight: 700;
  margin-bottom: 20px;
}

&__border {
  width: 32px;
  height: 2px;
  background-color: #202336;
}

&__selectionform {
  margin-top: 48px;

  &.dataform {
    display: flex;
  }
}

&__listbox{
  font-family: 'Inter', sans-serif;
  font-size: 15px;
  line-height: 32px;
  color: #4A4C53;
  width: 220px;
  height: 52px;
  padding: 0 20px;
  border: 1px solid #AFB0B9;
  border-radius: 6px;
  margin-bottom: 35px;
}

.location {
  margin-right: 30px;
}

.activity {
}

.grade {
  margin-right: 30px;
}

.date {
}

&__btn {
  font-family: 'Mulish', sans-serif;
  font-size: 17px;
  line-height: 50px;
  width: 180px;
  height: 52px;
  border-radius: 8px;
  color: #fff;
  background-color: #FB8F1D;
  text-align: center;
  margin-top: 15px;
  cursor: pointer;
}

&__img{
  max-width: 687px;
  max-height: 654px;
}
}

// second block
.container3 {
  max-width: 1240px;
  margin: 360px auto 40px auto;
}

.content {

```

```

    max-width: 480px;
    margin-top: 100px;
}

.about {
  display: flex;
  justify-content: space-between;

  &__img {
    max-width: 659px;
    max-height: 534px;
  }

  &__title {
    font-family: 'Playfair Display', serif;
    font-weight: 700;
    font-size: 36px;
    line-height: 50px;
    margin-bottom: 10px;
  }

  &__text {
    font-family: 'Mulish', sans-serif;
    font-size: 17px;
    font-weight: 300;
    line-height: 30px;
    color: #7D7987;
    max-width: 440px;
    margin-bottom: 30px;
  }

  &__btn {
    font-family: 'Inter', sans-serif;
    font-weight: 600;
    font-size: 16px;
    line-height: 50px;
    width: 164px;
    height: 48px;
    border-radius: 8px;
    color: #fff;
    background-color: #FB8F1D;
    text-align: center;
    cursor: pointer;
  }
}

//third block
.container4 {
  max-width: 1120px;
  margin: 130px auto 40px auto;
}

.features {

  &__title {
    font-family: 'Playfair Display', serif;
    font-size: 36px;
    font-weight: 700;
  }

  &__viewtext {
    font-family: 'Inter', sans-serif;
    font-weight: 600;
    font-size: 17px;
    line-height: 12px;
    color: #FB8F1D;
    margin-right: 10px;
  }

  &__viewarrow {
    max-width: 6px;
    max-height: 12px;
  }

  &__block {

  }
}

```

```

.toprow {
  display: flex;
  justify-content: space-between;
}

.viewbtn{
  display: flex;
  align-self: center;
  cursor: pointer;
}

.bottomrow {
  display: flex;
  justify-content: space-between;
  margin-top: 56px;
}

.feating {
  height: 332px;
  width: 261px;
  border-radius: 8px;
  cursor: pointer;
}

.featabout {
  position: relative;
  top: -50px;
  border-radius: 0 6px 0 0;
  width: 120px;
  height: 56px;
  background-color: #fff;
  padding-top: 8px;
  cursor: pointer;

  &__title {
    font-family: 'Inter', sans-serif;
    font-weight: 600;
    font-size: 17px;
    line-height: 25px;
  }

  &__place {
    font-family: 'Inter', sans-serif;
    font-size: 15px;
    font-weight: 100;
    line-height: 25px;
    color: #80918E;
  }
}

//fourth block
.container5 {
  max-width: 1130px;
  margin: 190px auto 40px auto;
}

.guicontent {
  max-width: 450px;
  margin-top: 150px;
}

.guides {
  display: flex;
  justify-content: space-between;

  &__title {
    font-family: 'Playfair Display', serif;
    font-weight: 700;
    font-size: 36px;
    line-height: 50px;
    margin-bottom: 20px;
  }

  &__text {
    font-family: 'Mulish', sans-serif;
    font-size: 17px;
    font-weight: 300;
    line-height: 30px;
    color: #7D7987;
  }
}

```

```

    margin-bottom: 35px;
}

&__btn {
    font-family: 'Inter', sans-serif;
    font-weight: 600;
    font-size: 16px;
    line-height: 50px;
    width: 164px;
    height: 48px;
    border-radius: 8px;
    color: #fff;
    background-color: #FB8F1D;
    text-align: center;
    cursor: pointer;
}

&__img {
    max-width: 599px;
    max-height: 542px;
}

}

//fifth block
.container6 {
    max-width: 1100px;
    margin: 210px auto 40px auto;
}

.test {

    &__title {
        font-family: 'Playfair Display', serif;
        font-weight: 700;
        font-size: 36px;
        line-height: 50px;
        margin-bottom: 60px;
    }
}

.slider {

    margin-bottom: 40px;

    .slide {

        display: flex;
        justify-content: space-between;

        &__mark {
            margin-bottom: 30px;
        }

        &__text {
            font-family: 'Mulish', sans-serif;
            font-size: 24px;
            font-weight: 400;
            line-height: 36px;
            color: #000000;
            margin-bottom: 40px;
        }

        &__name {
            font-family: 'Inter', sans-serif;
            font-size: 22px;
            font-weight: 700;
            line-height: 48px;
            color: #000000;
        }

        &__position {
            font-family: 'Inter', sans-serif;
            font-size: 18px;
            font-weight: 400;
            color: #000000;
        }
    }
}

```

```

.fade {
  -webkit-animation-name: fade;
  -webkit-animation-duration: 1.5s;
  animation-name: fade;
  animation-duration: 1.5s;
}
@-webkit-keyframes fade {
  from{opacity: .4}
  to{opacity: 1}
}
@keyframes fade {
  from{opacity: .4}
  to{opacity: 1}
}
}

.slidecontent {
  max-width: 450px;
  align-self: center;
}

.slideimg {

  .mainimg {
    z-index: 15;
  }

  .cross {
    position: relative;
    top: 472px;
    right: 400px;
    z-index: 20;
  }

  .circle {
    position: relative;
    top: -18px;
    right: 62px;
    z-index: -1;
  }
}

.arrows {
  display: flex;
  justify-content: center;

  &__prev, &__next {
    position: relative;
    cursor: pointer;
    align-self: center;
    font-weight: bold;
    font-size: 18px;
    transition: 0.6s ease;
    border-radius: 100% ;
    padding: 16px 20px;
  }

  &__prev {
    margin-right: 30px;
  }

  &__next {
    margin-left: 30px;
  }

  &__prev:hover, &__next:hover {
    background-color: #e4e7ef;
  }
}

//sixth block
.container7 {
  max-width: 1100px;
  margin: 130px auto 40px auto;
}

```

```

.trend {

  &__title {
    font-family: 'Playfair Display', serif;
    font-size: 36px;
    font-weight: 700;
  }

  &__viewtext {
    font-family: 'Inter', sans-serif;
    font-weight: 600;
    font-size: 17px;
    line-height: 12px;
    color: #FB8F1D;
    margin-right: 10px;
  }

  &__viewarrow {
    max-width: 6px;
    max-height: 12px;
  }

  &__block {
    max-width: 250px;
  }
}

.toprow {
  display: flex;
  justify-content: space-between;
}

.viewbtn{
  display: flex;
  align-self: center;
  cursor: pointer;
}

.bottomrow {
  display: flex;
  justify-content: space-between;
  margin-top: 56px;
}

.trendimg {
  height: 230px;
  border-radius: 8px;
  cursor: pointer;
  margin-bottom: 13px;
}

.trendabout {

  &__title {
    font-family: 'Inter', sans-serif;
    font-weight: 700;
    font-size: 19px;
    line-height: 30px;
    margin-bottom: 13px;
  }

  &__sub {
    font-family: 'Inter', sans-serif;
    font-size: 15px;
    font-weight: 400;
    line-height: 26px;
    color: #7D7987;
    margin-bottom: 23px;
  }

  &__read {
    font-family: 'Inter', sans-serif;
    font-size: 15px;
    font-weight: 600;
    line-height: 28px;
    color: #FB8F1D;
  }
}
}

```

```

//footer
footer {
  background-color: #F9F9FB;
}
.container8 {
  max-width: 1220px;
  margin: 230px auto 35px auto;
  padding-top: 80px;
}

.foot {

  .top {
    display: flex;
    justify-content: space-between;
    border-bottom: 1px solid #C4C4C4;
    .column1 {
      max-width: 340px;
    }
    .column2 {
    }
    .column3 {
    }
    .column4 {
    }
  }

  &__logo {
    margin-bottom: 26px;
    cursor: pointer;
  }

  &__about {
    font-family: 'Mulish', sans-serif;
    font-size: 18px;
    font-weight: 300;
    line-height: 28px;
    color: #848484;
    margin-bottom: 40px;
  }

  &__made {
    font-family: 'Mulish', sans-serif;
    font-size: 17px;
    font-weight: 300;
    line-height: 28px;
    color: #848484;
    margin-bottom: 81px;
  }

  &__title {
    font-family: 'Mulish', sans-serif;
    font-size: 19px;
    font-weight: 700;
    line-height: 60px;
    color: #000000;
  }

  &__list {
    font-family: 'Mulish', sans-serif;
    font-size: 18px;
    font-weight: 300;
    line-height: 38px;
    color: #848484;
  }
}

.social {
  display: flex;
  justify-content: center;
  margin-top: 30px;
  padding-bottom: 30px;

  &__img {
    margin-right: 35px;
    cursor: pointer;
  }
}

```

```

}

Slider
var slideIndex = 1;
showSlides(slideIndex);

function plusSlides(n){
  showSlides(slideIndex += n);
}

function currentSlide(n) {
  showSlides(slideIndex = n);
}

function showSlides(n) {
  var i;
  var slides = document.getElementsByClassName('slide');

  if (n>slides.length) {
    slideIndex = 1
  }
  if (n < 1) {
    slideIndex = slides.length
  }
  for (i=0; i<slides.length; i++) {
    slides[i].style.display = "none";
  }
  slides[slideIndex-1].style.display = "flex";
}

```

Register CSS

```

body {
  padding-top: 100px;
}

.container {
  max-width: 500px;
  margin: 0 auto;
  padding: 20px;
  background-color: #f2f2f2;
  border-radius: 5px;
}

h1 {
  text-align: center;
  margin-bottom: 20px;
}

.form-group {
  margin-bottom: 15px;
}

label {
  display: block;
  margin-bottom: 5px;
}

input {
  max-width: 480px;
  width: 100%;
  padding: 10px;
  border: 1px solid #ccc;
  border-radius: 5px;
}

button {
  display: block;
  width: 100%;
  padding: 10px;
  background-color: #FB8F1D;
  color: #fff;
  border: none;
  border-radius: 5px;
  cursor: pointer;
}

```

```

a{
  color: #fff;
  text-decoration: none;
}

button:hover {
  background-color: #ce6a00;
}

```

Employee Finder

```

def search_employees(vacancy, employees):
    matched_employees = []

    for employee in employees:
        if vacancy['art_type'] == employee['art_type'] and vacancy['experience'] <= employee['experience']:
            if vacancy['location'] == employee['location']:
                matched_employees.append(employee)
            elif vacancy['location'] == 'Any':
                matched_employees.append(employee)

    matched_employees = sorted(matched_employees, key=lambda x: x['rating'], reverse=True)

    return matched_employees

# Приклад вхідних даних

vacancy = {
    'art_type': 'Painting',
    'experience': 3,
    'location': 'New York',
    'rating': 4.5
}

employees = [
    {
        'name': 'John',
        'art_type': 'Painting',
        'experience': 2,
        'location': 'New York',
        'rating': 4.2
    },
    {
        'name': 'Emily',
        'art_type': 'Sculpture',
        'experience': 4,
        'location': 'Los Angeles',
        'rating': 4.8
    },
    {
        'name': 'David',
        'art_type': 'Painting',
        'experience': 5,
        'location': 'New York',
        'rating': 4.9
    },
    {
        'name': 'Sophia',
        'art_type': 'Photography',
        'experience': 3,
        'location': 'San Francisco',
        'rating': 4.6
    }
]

matched_employees = search_employees(vacancy, employees)

print("Matched employees:")
for employee in matched_employees:
    print(f"Name: {employee['name']}, Rating: {employee['rating']}")

def sort_profiles(vacancy, profiles):
    sorted_profiles = sorted(profiles, key=lambda x: (x['art_type'] != vacancy['art_type'], x['experience'], x['rating']), reverse=True)
    return sorted_profiles

```

```

vacancy = {
    'art_type': 'Painting',
    'experience': 3,
    'location': 'New York',
    'rating': 4.5
}

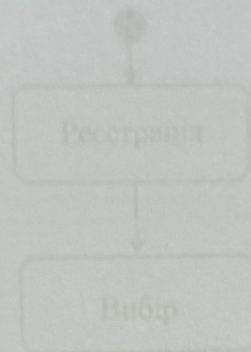
profiles = [
    {
        'name': 'John',
        'art_type': 'Painting',
        'experience': 2,
        'location': 'New York',
        'rating': 4.2
    },
    {
        'name': 'Emily',
        'art_type': 'Sculpture',
        'experience': 4,
        'location': 'Los Angeles',
        'rating': 4.8
    },
    {
        'name': 'David',
        'art_type': 'Painting',
        'experience': 5,
        'location': 'New York',
        'rating': 4.9
    },
    {
        'name': 'Sophia',
        'art_type': 'Painting',
        'experience': 3,
        'location': 'San Francisco',
        'rating': 4.6
    }
]

sorted_profiles = sort_profiles(vacancy, profiles)

print("Sorted profiles:")
for profile in sorted_profiles:
    print(f"Name: {profile['name']}, Art Type: {profile['art_type']}, Experience: {profile['experience']}, Rating: {profile['rating']}")

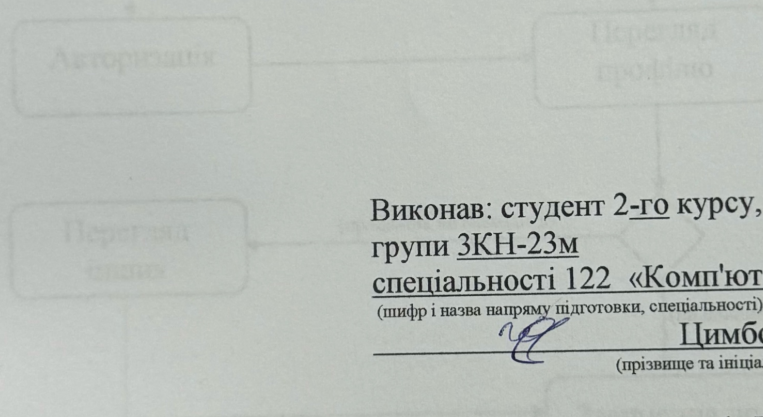
```

Додаток В (обов'язковий)



ІЛЮСТРАТИВНА ЧАСТИНА

ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ ДЛЯ ПОШУКУ ПРАЦІВНИКІВ

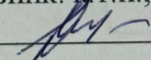


Виконав: студент 2-го курсу,
групи ЗКН-23м

спеціальності 122 «Комп'ютерні науки»
(шифр і назва напрямку підготовки, спеціальності)

 Цимборович Ю. В.
(прізвище та ініціали)

Керівник: к.т.н., проф. каф. КН

 Колесницький О. К.
(прізвище та ініціали)

« 05 » 12 2024 р.

Рисунок В.1 – Діаграма діяльності інформаційної технології для пошуку працівників

Вінниця ВНТУ - 2024 рік

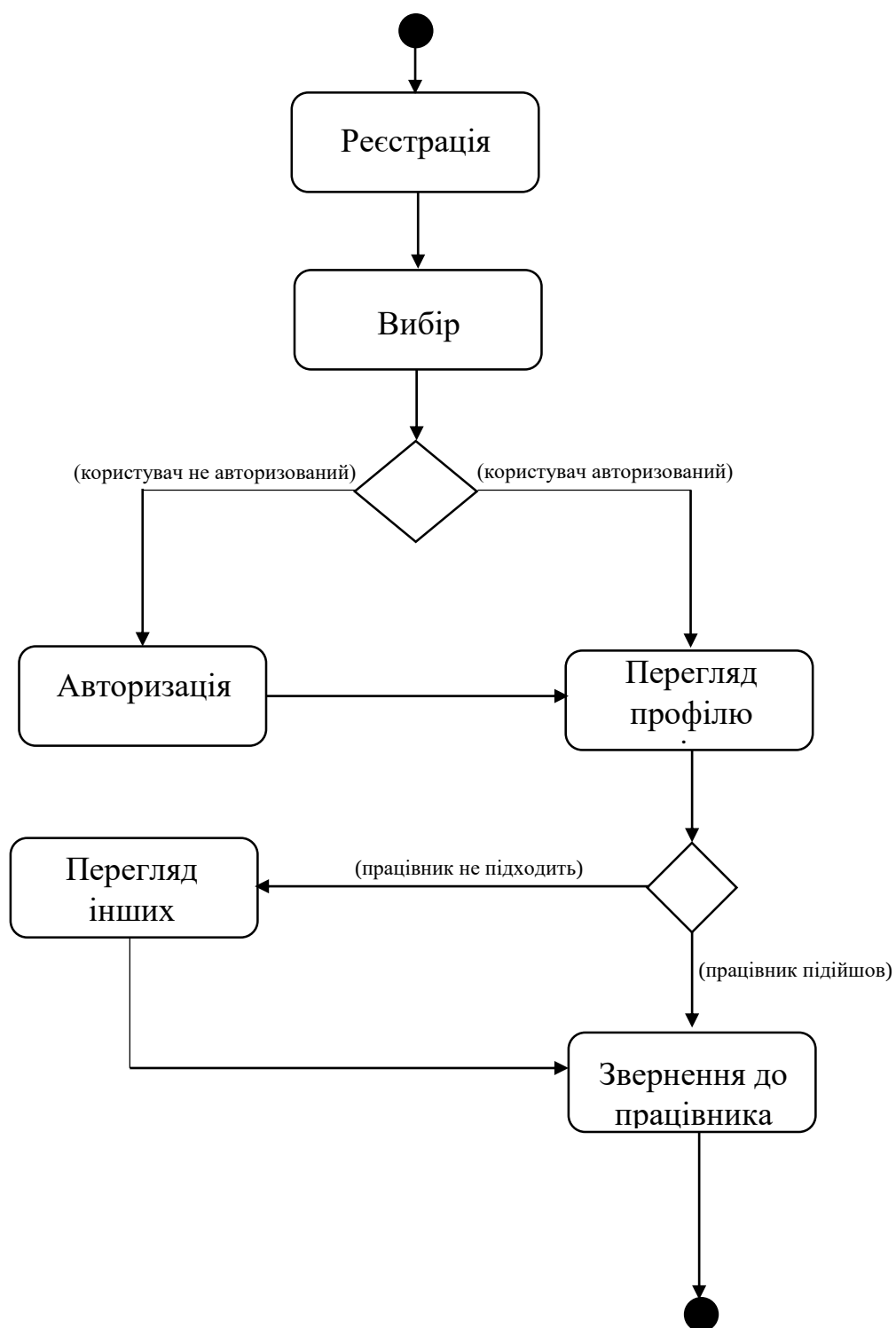


Рисунок В.1 – Діаграма діяльності інформаційної технології для пошуку працівників

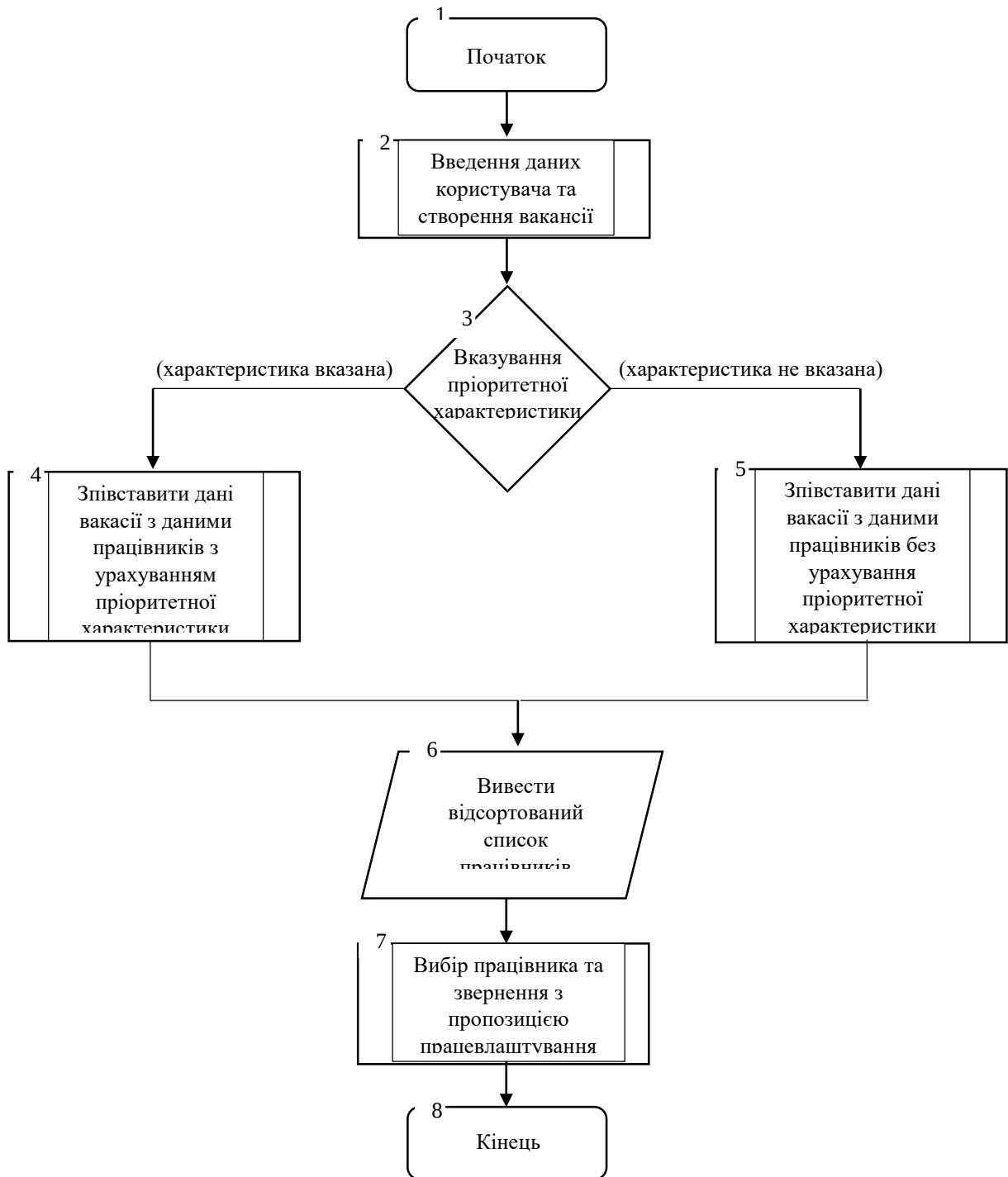


Рисунок В.2 – Блок-схема алгоритму пошуку працівників за вакансією

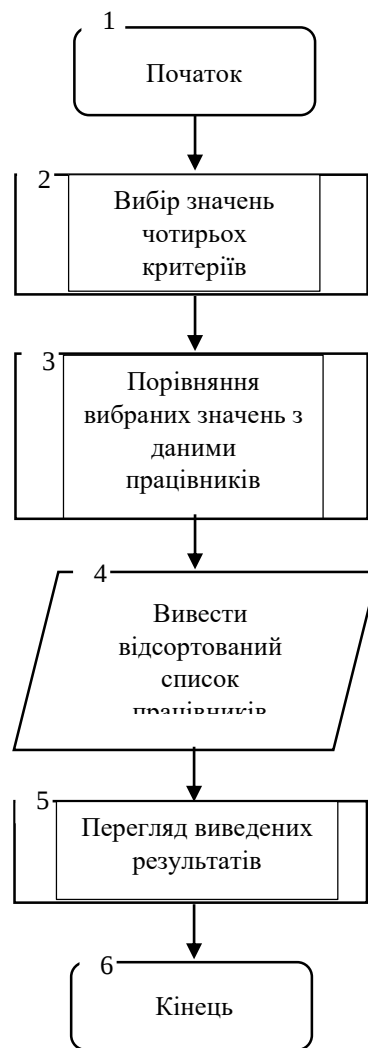


Рисунок В.3 – Блок-схема алгоритму пошуку працівників по 4 критеріям

Jobfinder for artists

Monetize your hobby

Location ▼	Activity ▼
Experience ▼	Rating ▼
Find	



Рисунок В.4 – Головна сторінка онлайн платформи для пошуку працівників

Створення вакансії

Тип активності:
Виберіть

Бажаний досвід (років):
5

Місце розташування:
Київ

Оплата \$:
500

Пріоритетна характеристика:
Досвід

Детальний опис вакансії:

WNews Media Group оголошує про відкриття вакансії дизайнера-ілюстратора для роботи над анімаційними відео для наших каналів на Facebook та YouTube.

«WNews» – це мережа стрімко зростаючих каналів на 7 мовах. Наші відео щомісяця набирають сотні мільйонів переглядів у всьому світі. Співаємо як з досвідченими, так і з фахівцями-початківцями.

У нас є можливість працювати як на ставці, так і відрадно!

Підтвердити

Рисунок В.5 – Сторінка створення вакансії

Працівники які вам підходять найкраще

Дюдмила Барич Художник. 6 років досвіду. м.Київ Спеціалізація - портрети.	Марія Шаповалова Художник. 5 років досвіду. м.Київ Спеціалізація - сучасне мистецтво.	Андрій Ковальський Художник. 5 років досвіду. м.Київ Спеціалізація - натюрморти.
Дюдмила Барич Художник. 5 років досвіду. м.Вінниця	Марія Шаповалова	Андрій Ковальський Художник. 5 років досвіду. м.Ужгород

Рисунок В.6 – Сторінка вибору працівників відносно створеної вакансії

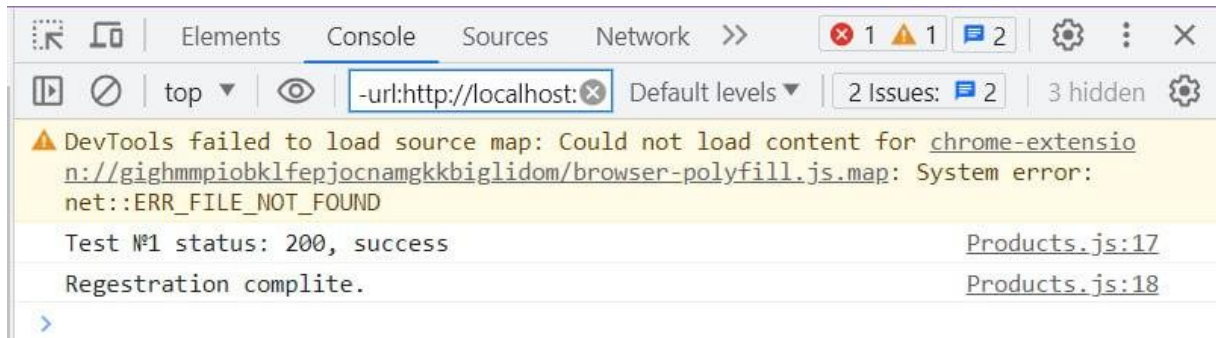


Рисунок В.7 – Результати виконання модульного тесту №1

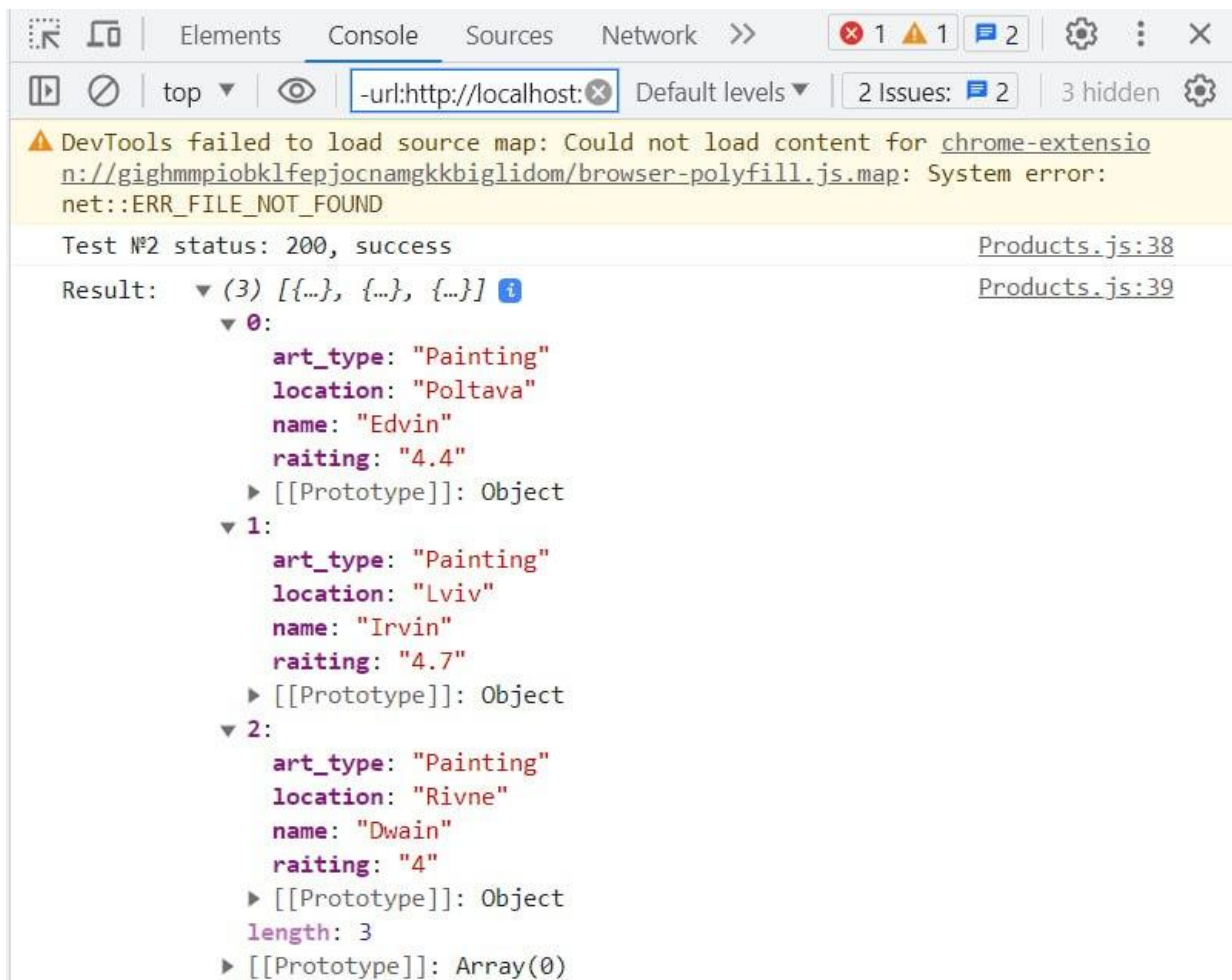


Рисунок В.8 – Результати виконання модульного тесту №2