

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА»
ФАКУЛЬТЕТ ІНФОРМАЦІЙНО-КОМП'ЮТЕРНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

ПОЯСНЮВАЛЬНА ЗАПИСКА

до кваліфікаційної роботи освітнього ступеня «магістр»
за спеціальністю 121 «Інженерія програмного забезпечення»
(освітня програма «Інженерія програмного забезпечення»)
на тему:

«Платформа для пошуку та наймання працівників з використанням AI»

Виконав студентка групи ЗІПЗмд-21-1
ШЕРЕМЕТ Крістіна Ральфівна

Керівник роботи:
ВЛАСЕНКО Олег Васильович

Рецензент:
ЄФРЕМОВ Юрій Миколайович

Житомир – 2023

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА»
ФАКУЛЬТЕТ ІНФОРМАЦІЙНО-КОМП'ЮТЕРНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

«ЗАТВЕРДЖУЮ»

В.о. зав. кафедри інженерії
програмного забезпечення
Андрій МОРОЗОВ
«01» лютого 2023 р.

ЗАВДАННЯ
на кваліфікаційну роботу

Здобувач вищої освіти: ШЕРЕМЕТ Крістіна Ральфівна

Керівник роботи: ВЛАСЕНКО Олег Васильович

Тема роботи: **«Платформа для пошуку та наймання працівників з використанням AI»**,

затверджена Наказом закладу вищої освіти від «01» лютого 2022 р. № 58\с

Вихідні дані для роботи:

Об'єкт дослідження – процес прийняття рішення при відборі IT спеціалістів.

Предмет дослідження – застосування системи відстеження кандидатів з AI для відбору кандидатів на посаду.

Консультанти з магістерської кваліфікаційної роботи із зазначенням розділів, що їх стосуються:

Розділ	Консультант	Завдання видав	Завдання прийняв
1	Власенко О.В.	03.02.2023р.	03.02.2023р.
2	Власенко О.В.	03.02.2023р.	03.02.2023р.
3	Власенко О.В.	03.02.2023р.	03.02.2023р.

Дата видачі завдання «03» лютого 2023 р.

РЕФЕРАТ

У цій дипломній роботі було проведено ретельний аналіз існуючих методів найму та програмного забезпечення для оптимізації найму. Метою було покращити та оптимізувати існуючі процеси підбору персоналу в компанії Ахіот Про. У результаті стало зрозуміло, що поточне рішення, яке використовується в компанії, має бути автоматизовано за допомогою більш широкого спектру програмного забезпечення. Після аналізу поточних рішень на ринку було вирішено вдосконалити внутрішню CRM, щоб підвищити її зручність у використанні, прискорити процес найму, заощадити зусилля компанії та уникнути використання занадто дорогого або обмеженого програмного забезпечення на ринку. Таке вдосконалення мало б однозначний позитивний ефект як з економічної точки зору, так і з точки зору стратегічного розвитку компанії та її спеціалістів.

Дипломна робота складається з: вступу, 3 розділів, висновків, списку використаних джерел, 26 ілюстрацій та 7 таблиць у тексті. Повний обсяг роботи 61 сторінка.

Ключові слова: система відстеження заявників, CRM, автоматизація, штучний інтелект, AI, програмна платформа

ABSTRACT

This thesis work was to execute a close analysis of existing methods of recruitment and recruitment optimization software. The goal was to improve and optimize the existing recruitment processes within Axiom Pro company. As a result, it became clear that the current solution used within the company has to be automated by using a wider spectrum of software. After analyzing current solutions in the market was decided to improve internal CRM in order to increase its usability, speed up the recruitment process, save the company's effort and avoid using overpriced or limited software on the market. Such an improvement would have an unequivocal positive effect both from an economic point of view and from the point of view of the strategic development of the company and its specialists.

This thesis consists of: an introduction, 3 chapters, conclusions, a list of used sources, 26 illustrations in the text. The full size of the work is 56 pages.

Keywords: applicant tracking system, CRM, automation, AI, software platform

ЗМІСТ

РЕФЕРАТ	2
ВСТУП	7
РОЗДІЛ 1. СУЧАСНИЙ ПОШУК ІТ-РОЗРОБНИКА НА ПОСАДУ ТА ЙОГО МЕТОДИ	9
1.1. Аналіз сучасного пошуку кваліфікованих програмістів	10
1.2. Методи сучасного пошуку кваліфікованих програмістів	11
1.3. Типовий процес відбору кандидата	11
РОЗДІЛ 2. СИСТЕМИ ПОШУКУ ТА ВІДБОРУ КАНДИДАТІВ	14
2.1. Що таке система відстеження кандидатів	14
2.2. Існуючі ATS	16
2.2.1. LinkedIn Talent Solutions	16
2.2.2. Workable	29
2.2.3. HackerRank	23
2.3. Висновки та порівняння ATS	26
2.4. Порівняння існуючих технологій	28
РОЗДІЛ 3. ПРОГРАМНЕ РІШЕННЯ	30
3.1. Постановка задачі	30
3.1.1. PHP	31
3.1.2. Laravel	34
3.1.3. Java Script	35
3.1.4. СКБД MySQL	37
3.1.5. SpeechAce	39
3.1.6. Jude0	41
3.1.7. Affinda	42
3.2. Опис модулю виконаного у рамках цієї дипломної роботи	43
3.2.1. Процес створення профілю кандидата	43

3.2.2. Процес відбору кандидату рекрутером	49
3.3. Результати роботи Recruiter після додавання модулю	55
ВИСНОВКИ	57
ЛІТЕРАТУРА	59

ВСТУП

Актуальність теми. За останні десятиліття інформаційні технології зазнали значного розквіту. Із зростанням вимог ринку рекрутингу, навиків та знань, пошук кваліфікованих кандидатів на посади став складним та часовитратним процесом. Останні дослідження показують, що у 2022 році процес рекрутингу на позицію середньому 41 день. [1] Багато компаній звертаються до рекрутерів, але навіть професійні рекрутери не завжди можуть забезпечити швидкий та ефективний пошук. Однак, з розвитком технологій, автоматизація пошуку та найму стала все більш популярною серед компаній.

Автоматизовані платформи для пошуку та найму спеціалістів стали рішенням для багатьох компаній, що дозволяє ефективно та швидко відібрати потрібних кандидатів. За допомогою цих платформ рекрутери можуть швидко створити вакансію, розмістити її на різних ресурсах, відібрати кандидатів за заданими параметрами та здійснювати відбір з використанням інструментів штучного інтелекту, що дозволяє автоматизувати та прискорити весь процес.

Однією з головних проблем із сучасними системами відстеження кандидатів (Applicant Tracking System, ATS) є відсутність універсальності і повна адаптованість до технічних та нетехнічних посад. А також потенційна упередженість у процесі найму. ATS розроблено, щоб допомогти рекрутерам і менеджерам з найму ефективніше керувати заявками на роботу, але вони можуть ненавмисно внести упередженість у процес найму, якщо не налаштувати та використовувати належним чином.

Загалом, сучасні ATS можуть бути дорогими, особливо для малого та середнього бізнесу.

Метою даної дипломної роботи є розширення функціоналу існуючої CRM щоб покращити існуючий цикл найму робітників, зробити його більш ефективним, заощадити зусилля співробітників і гроші компанії, швидше найняти якісних кандидатів. Крім того, зробити цю платформу привабливою для

потенційних покупців, якими будуть кадрові агентства та компанії, які хочуть заощадити гроші на внутрішньому підборі персоналу.

Встановлена мета обумовлює наступні завдання:

- аналіз сучасного циклу найму та його проблеми;
- аналіз існуючих автоматизованих платформ для пошуку та найму з використанням штучного інтелекту;
- вибір засобів вирішення для покращення та удешевлення існуючого функціоналу.

Об'єкт дослідження – пошук спеціалістів та процес прийняття рішення при їх відборі.

Предмет дослідження – розробка системи відстеження кандидатів за допомогою AI для відбору кандидатів на посаду.

Інформаційну базу дослідження складають матеріали світових комерційних організації, зовнішніх агентств, дослідних інститутів, а також розміщені у мережі Інтернет.

Наукова новизна. В цій роботі запропоновано застосування системи, яка розроблена для більш ефективного процесу найму, де штучний інтелект і людина можуть швидко разом приймати кращі рішення.

Інноваційність роботи: платформа, яка має функціональні можливості кращих ATS, але була б підлаштована під технічні та нетехнічні вакансії за набагато доступнішу ціну.

Практичне значення роботи полягає у розробці: модулю вбудованого у існуючу CRM, який тестує знання та порівнює навички та досвід кандидатів на посаду.

РОЗДІЛ 1. СУЧАСНИЙ ПОШУК ІТ-РОЗРОБНИКА НА ПОСАДУ ТА ЙОГО МЕТОДИ

1.1. Аналіз сучасного пошуку кваліфікованих кадрів

У більшості випадків від рекрутера залежить те, як кваліфікований персонал буде працювати в компанії та як швидко буде заповнено ту чи іншу вакансію. Рекрутер - це перший співрозмовник кандидата, і від його професіоналізму залежить враження фахівця про компанію. Згідно з дослідженням Candidate Experience Study компанії CareerArc [2], 60% фахівців мали поганий досвід взаємодії з компанією. У результаті:

- 63% відхилили пропозицію про роботу;
- 72% розповіли про свій негативний досвід друзям і в соціальних мережах;
- 64% відмовилися від продукції або послуг компанії-роботодавця.

Це всі показники того, що досвід кандидата та його враження від рекрутера, і, як наслідок, від компанії, є обов'язковим індикаторами, які можуть вплинути не тільки на швидкість найму, але й на репутацію компанії, а також на фінансову складову.

Проблеми сучасного пошуку кваліфікованих кадрів можуть відрізнятися залежно від регіону, сфери діяльності та конкуренції на ринку праці. [3, 4, 5] Однак, основні проблеми, з якими стикаються роботодавці при пошуку кваліфікованих програмістів, включають наступне:

1. Низький рівень освіти та навичок: у багатьох країнах рівень освіти в галузі не відповідає вимогам роботодавців. Недостатній рівень навичок може призводити до того, що кандидати не можуть виконувати потрібні завдання або не можуть ефективно працювати в команді.
2. Брак досвіду: більшість роботодавців шукають кваліфікованих кадрів з певним досвідом. Однак, відсутність досвіду може бути перешкодою для молодих випускників та новачків у галузі.

3. Висока конкуренція на ринку праці: зростання популярності певних галузей та технологій приводить до збільшення кількості кандидатів, які звертаються до роботодавців. Це може створювати проблему з відбором найбільш кваліфікованих та підходящих кандидатів.
4. Стрес та висока швидкість технологічних змін: галузі постійно змінюється та розвивається, що призводить до того, що кандидати повинні постійно навчатися та адаптуватися до нових технологій. Це може призвести до високого рівня стресу та вимог до кадрів.
5. Проблема зі збереженням персоналу: багато людей можуть змінювати місце роботи досить часто, що призводить до проблеми зі збереженням персоналу та пошуку нових кваліфікованих кандидатів. Більшість шукають роботу з високою зарплатою, можливістю професійного розвитку, гнучким графіком та хорошими умовами праці.
6. Брак комунікаційних та міжособистісних навичок: робота у галузі часто пов'язана з роботою в команді, де необхідно добре володіти комунікаційними та міжособистісними навичками. Брак таких навичок може призводити до того, що програмісти не можуть ефективно працювати в команді або не можуть взаємодіяти з клієнтами та іншими робітниками компанії.

Для вирішення цих проблем роботодавці можуть використовувати різні підходи, такі як пошук талановитих кадрів у університетах та коледжах, організація стажувань та навчальних програм для молодих випускників, а також використання різноманітних методів оцінювання та відбору кандидатів. Також важливо забезпечити кваліфікованим кандидатам хороші умови праці, щоб зберігати їх у компанії та створювати конкурентні переваги для приваблення нових.

1.2. Методи сучасного пошуку кваліфікованих програмістів

Існує декілька методів оцінювання та відбору кандидатів, які використовуються при пошуку кваліфікованих кандидатів. Ось декілька з них:

1. Інтерв'ю. Цей метод передбачає проведення інтерв'ю з кандидатом, під час якого він повинен відповісти на запитання щодо своїх знань, досвіду та здатності вирішувати технічні завдання. Цей метод дозволяє оцінити кваліфікацію кандидата та його здатність дійсно вирішувати завдання.
2. Тестування. Цей метод передбачає проведення різноманітних тестів, які оцінюють знання та навички кандидата у галузі програмування. Тести можуть бути різних рівнів складності та можуть включати як теоретичні запитання, так і практичні завдання. Цей метод дозволяє оцінити рівень знань та навичок кандидата у галузі програмування.
3. Портфоліо. Цей метод передбачає оцінювання робіт, які кандидат виконував раніше. Кандидат може надати своє портфоліо, де він демонструє свої навички у програмуванні. Цей метод дозволяє оцінити рівень досвіду та знань кандидата, а також його здатність розв'язувати реальні технічні завдання.
4. Резюме дозволяє зрозуміти який кандидат вже має досвід, його освіта, навички.
5. Соціальні мережі та профілі на сайтах з роботою. Цей метод полягає у вивченні профілю кандидата на соціальних мережах, таких як LinkedIn, або на сайтах з роботою, таких як Indeed чи Glassdoor. В цих профілях можуть бути вказані інформація про досвід роботи, навички та рекомендації колег або клієнтів. Цей метод дозволяє оцінити професійний досвід та рівень знань кандидата, а також його репутацію та комунікаційні навички.

1.3. Типовий процес відбору кандидата

Типовий процес відбору кандидата складається із наступних кроків (Рис. 1.1):

1. Визначте потреби в наймі: Першим кроком у будь-якому процесі найму є визначення потреб компанії в наймі. Це включає визначення навичок і досвіду, необхідних для посади, а також посадові обов'язки та структуру звітності.
2. Розробка опису посади та опублікування оголошення про вакансію: виходячи з потреб найму, розробляється посадова інструкція, яка чітко визначає обов'язки, вимоги та кваліфікацію. Далі розміщують оголошення на різних дошках вакансій і платформах соціальних мереж. Оголошення має бути стислим, привабливим і чітко викладати вимоги та очікування від роботи.
3. Перевірка резюме: в поданих заявках перевіряють резюме та супровідний лист на базову кваліфікацію та досвід, необхідні для посади. Складають список найкращих кандидатів на основі критеріїв, зазначених у описі вакансії.
4. Відео чи особиста співбесіда: під час співбесіди ставляться відкриті запитання, щоб оцінити досвід кандидата, навички вирішення проблем, комунікативні здібності та відповідність культурі.
5. Технічна оцінка: залежно від характеру посади може знадобитися провести технічну оцінку, щоб оцінити здібності кандидата до кодування. Це можна зробити, призначивши їм завдання для кодування та подальшого аналізу коду.
6. Перевірка знання англійської мови: залежно від характеру посади може знадобитися провести перевірку мовних навичок і впевнитися, що кандидат має змогу спілкуватися з командою та клієнтами.
7. Вибірка кращого кандидата: аналіз відгуків по співбесіді, результатів тестів, та резюме для обрання кращого кандидату.
8. Пропозиція про роботу: підготовка та відправка пропозиції про роботу, у якій зазначено зарплату, переваги та інші умови працевлаштування.

9. Підписання контракту: коли всі умови узгоджено, підготовлюється контракт для підпису з обох сторін, компанії та кандидата.

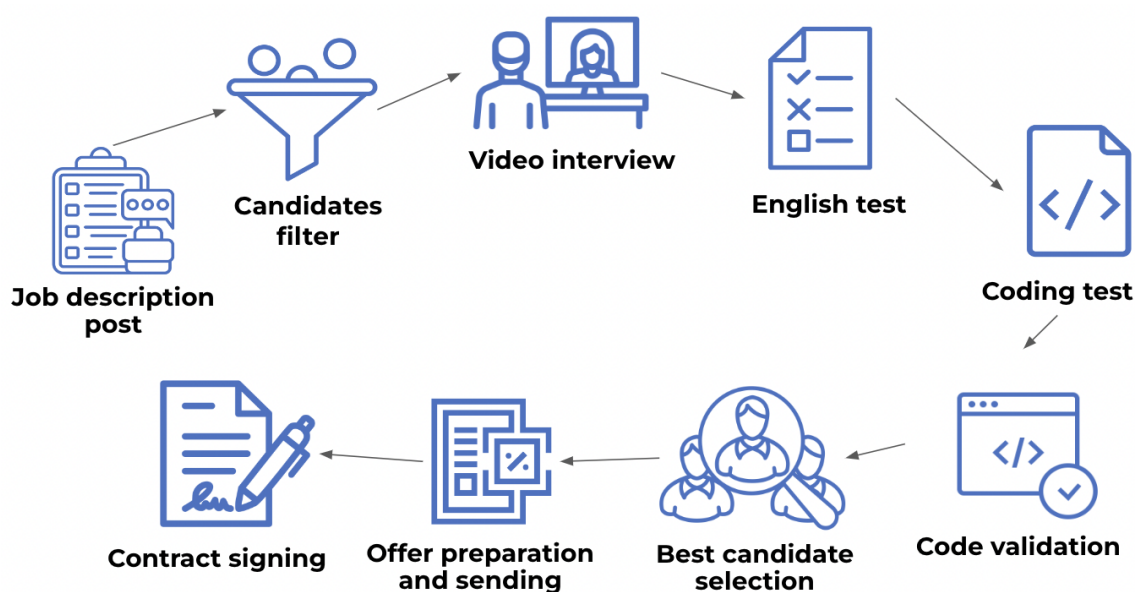


Рис. 1.1 Приклад циклу найму на роботу ІТ спеціаліста

Але як рекрутер може встигнути зібрати інформацію від керівників, створити вакансію, розмістити її на порталах, розібратися з відгуками, провести резюме, протестувати знання, відібрати кращих та при цьому не забути всім надати фідбек і відповісти на всі запитання? Це призводить до одного висновку - автоматизація є неминучою! У сучасному пошуку кваліфікованих програмістів все частіше використовуються інструменти для автоматизації відбору кандидатів, такі як системи керування кандидатами (Applicant Tracking System, ATS) та рекрутингові чат-боти. Ці інструменти дозволяють компаніям більш ефективно та швидко відбирати кандидатів за допомогою фільтрації резюме та іншої інформації про кандидатів.

РОЗДІЛ 2. СИСТЕМИ ПОШУКУ ТА ВІДБОРУ КАНДИДАТІВ

2.1. Що таке система відстеження кандидатів

Система відстеження кандидатів (ATS) — це програмний додаток, призначений для того, щоб допомогти компаніям оптимізувати процес набору та найму персоналу. Система служить центральною базою даних, де збираються та керуються всі заявки на роботу. (Рис. 2.1) [9]



Рис. 2.1 Складові системи відстеження кандидатів

Основною функцією ATS є автоматизація первинного перегляду заявок на роботу шляхом відсіювання кандидатів, які не відповідають мінімальним вимогам до посади. Це робиться шляхом сканування резюме та інших матеріалів заявки на наявність конкретних ключових слів і фраз, які відповідають вимогам вакансії.

Окрім автоматизації процесу первинного відбору, ATS також може полегшити спілкування між рекрутерами, менеджерами з найму та кандидатами.

Він може надсилати кандидатам автоматичні електронні листи, щоб повідомити їх про статус їхньої заявки та запланувати співбесіди.

ATS також можна використовувати для відстеження показників, пов'язаних із процесом найму, таких як кількість претендентів, час найму та джерело кандидата (наприклад, дошки вакансій, соціальні мережі, рекомендації тощо). Ці дані можуть бути використані для оптимізації процесу найму та підвищення якості кандидатів, які приймаються на роботу.

Система відстеження кандидатів (ATS) значно вплинула на процес найму персоналу та змінила підхід компаній до найму. Нижче наведено кілька способів, за допомогою яких ATS змінила наймання:

- підвищення ефективності: одним із найважливіших способів, якими ATS змінила процес найму, є підвищення ефективності процесу найму. ATS автоматизує багато звичайних завдань, таких як перевірка резюме та планування співбесід, звільняючи час рекрутерів, щоб зосередитися на більш стратегічних завданнях, таких як налагодження стосунків із кандидатами та наймання менеджерів;
- підвищена точність: використовує алгоритми для сканування та аналізу резюме та інших матеріалів заявки, що підвищило точність і послідовність процесу перевірки. Це зменшує ймовірність людської упередженості та гарантує об'єктивну оцінку кандидатів на основі їхньої кваліфікації та досвіду;
- покращений досвід роботи з кандидатами: за допомогою ATS кандидати можуть подавати заявки на роботу онлайн і відстежувати статус своєї заявки в режимі реального часу. Вони можуть отримувати автоматичні оновлення електронною поштою про статус своєї заявки та про наступні кроки в процесі. Це покращило досвід кандидатів, забезпечивши більшу прозорість і зменшивши невизначеність;
- кращі показники: ATS дозволяє рекрутерам відстежувати різні показники, пов'язані з процесом найму, наприклад, кількість отриманих заявок, час

найму та джерело кандидата. Ця інформація може бути використана для оптимізації процесу найму та підвищення якості найму;

- покращена відповідність: ATS також допомагає компаніям виконувати законодавчі та нормативні вимоги щодо найму персоналу. Наприклад, це може допомогти переконатися, що оголошення про роботу відповідають законам і нормам щодо рівних можливостей працевлаштування.

Загалом ATS змінила процес найму, підвищивши ефективність, точність, досвід кандидатів, показники та відповідність вимогам. Це дозволило компаніям приймати більш обґрунтовані рішення щодо найму та спростило процес найму, зробивши його швидшим та ефективнішим.

2.2. Існуючі ATS

2.2.1. LinkedIn Talent Solutions

Є однією з найбільших та найпопулярніших автоматизованих платформ для пошуку та найму IT-спеціалістів. LinkedIn Talent Solutions забезпечує доступ до великої бази даних професійних профілів, що дозволяє рекрутерам швидко та ефективно знайти кандидатів з потрібними навичками та досвідом. [6]

Однією з переваг LinkedIn Talent Solutions є можливість побудови зручної інтерактивної сторінки компанії, на якій можна публікувати вакансії та інформацію про бренд. Крім того, платформа забезпечує зручний інтерфейс для зв'язку з кандидатами, зокрема, можливість надсилання повідомлень та заявок на співбесіду.

Для забезпечення ефективності пошуку використовує алгоритми машинного навчання та інші інструменти для аналізу даних професійних профілів та рекомендацій кандидатів. Крім того, платформа надає можливість створення власних фільтрів для пошуку кандидатів з певним рівнем досвіду, навичками та іншими параметрами (Рис. 2.2).

Однією з особливостей LinkedIn Talent Solutions є те, що платформа дозволяє рекрутерам залучати до пошуку кандидатів інших співробітників компанії, що дозволяє залучити до пошуку більш розмаїтих та висококваліфікованих кандидатів.

Розробка LinkedIn Talent Solutions, ймовірно, передбачає використання кількох мов програмування та технологій. Як великий і складний пакет програмного забезпечення LinkedIn Talent Solutions, використовує низку мов і фреймворків для створення різноманітних інструментів і служб. Ось деякі з мов програмування, які використовували в розробці LinkedIn Talent Solutions [7]:

- Java - є популярною мовою програмування, яка широко використовується для розробки програмного забезпечення на рівні підприємства. LinkedIn побудовано в основному на віртуальній машині Java (JVM), і цілком ймовірно, що деякі частини LinkedIn Talent Solutions також побудовані з її використанням;
- JavaScript - це мова програмування, яка широко використовується для розробки веб-додатків. LinkedIn Talent Solutions, використовує JavaScript для своїх веб-інструментів та інтерфейсів найму персоналу;
- Espresso — це онлайнова, розподілена, відмовостійка база даних NoSQL, яка розроблена та використовується LinkedIn для її критично важливих програм. Це база даних документів, що означає, що вона зберігає дані як документи замість рядків і стовпців, як у традиційних реляційних базах даних. Цей підхід забезпечує більш гнучку та масштабовану модель даних, яка може краще обробляти складні та змінні структури даних. Бази даних документів, також зазвичай масштабуються краще, ніж реляційні бази даних, оскільки вони не вимагають складних об'єднань або змін схем для додавання нових даних [8];
- Scala — це мова програмування, яка працює на JVM і призначена для створення масштабованих і високопродуктивних програм. LinkedIn Talent

Solutions може використовувати Scala для обробки даних і інструментів аналізу, які вимагають швидкої та ефективної роботи.

Платформа також використовує низку технологій для розробки та надання своїх рішень з підбору персоналу:

- хмарні обчислення – LinkedIn Talent Solutions створено з використанням платформ хмарних обчислень які забезпечують масштабовану та безпечну інфраструктуру для запуску інструментів і служб найму;
- штучний інтелект (AI) і машинне навчання (ML) – LinkedIn використовує технології AI і ML для забезпечення своїх інструментів найму. Ці технології допомагають автоматизувати пошук і перевірку кандидатів, ідентифікувати релевантних кандидатів на роботу та надавати інформацію на основі даних для прийняття рішень щодо найму;
- аналітика великих даних. Інструменти підбору персоналу LinkedIn використовують аналітику великих даних, щоб надати рекрутерам уявлення про галузеві тенденції, аналіз конкурентів і дані про таланти. Платформа збирає та аналізує дані зі своєї величезної бази користувачів, щоб надавати актуальну інформацію рекрутерам;
- обробка природної мови (NLP) – LinkedIn використовує технології NLP для забезпечення своїх функцій підбору вакансій і пошуку. Ці технології аналізують описи посад і профілі кандидатів, щоб підібрати найбільш релевантних кандидатів для вакансій;
- API та інтеграції. Інструменти підбору персоналу LinkedIn створені за допомогою API та інтеграцій, які дозволяють їм підключатися до інших інструментів і платформ для підбору персоналу. Наприклад, LinkedIn інтегрується з системами відстеження кандидатів (ATS), щоб рекрутери могли керувати даними кандидатів і заявками зі своїх ATS.

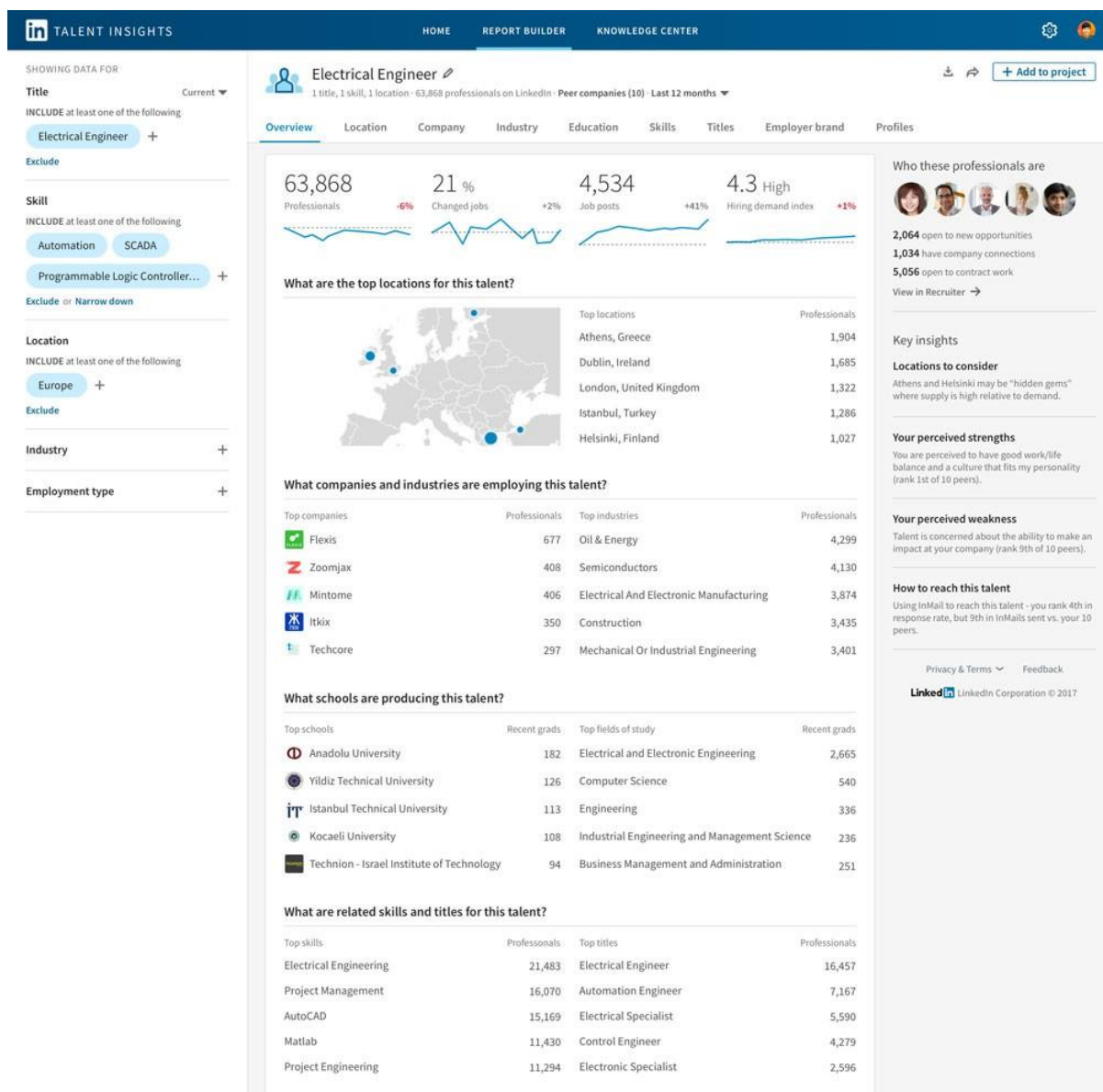


Рис. 2.2 Панель управління в LinkedIn Talent Solutions

2.2.2. Workable

Це інтерактивна платформа для рекрутингу, яка дозволяє компаніям швидко і ефективно знайти і найняти кандидатів. Розроблена в 2012 році, Workable стала однією з найбільш успішних та інноваційних платформ для рекрутингу, забезпечуючи компаніям інструменти для керування всіма етапами процесу найму. [9]

Workable дозволяє користувачам опублікувати вакансії на різних сайтах з пошуку роботи, таких як Indeed, Glassdoor, SimplyHired, і ZipRecruiter, а також на власному сайті компанії. Крім того, платформа надає користувачам можливість виконувати такі дії, як відстеження вакансій, збір заявок, розсилка повідомлень та інші.

Основні переваги Workable полягають у зручності використання, простоті, ефективності та інтеграції з іншими інструментами рекрутингу, такими як LinkedIn, GitHub та інші (Рис. 2.3).

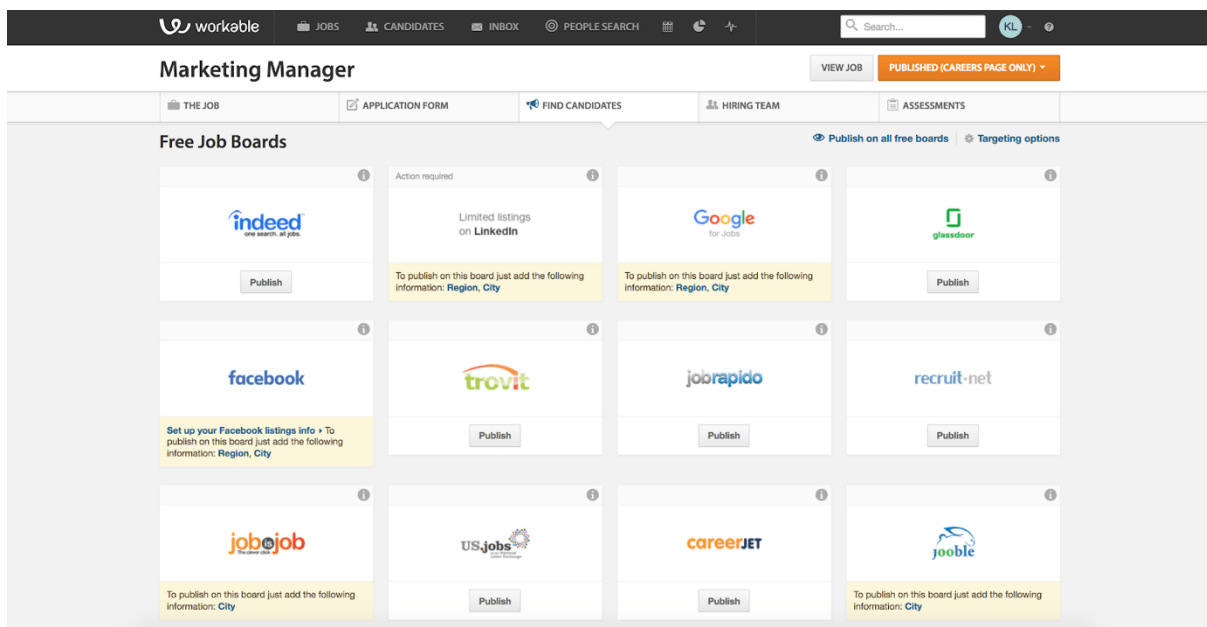


Рис. 2.3 Панель управління розсилкою об'яви на платформах з пошуку роботи на Workable

Workable дозволяє відстежувати ефективність ваших вакансій та стежити за кандидатами зі всіх джерел. Користувачі можуть використовувати платформу для керування всіма етапами процесу найму, включаючи опрацювання резюме, співбесіди та прийняття рішень щодо найму (Рис. 2.4).

Настроювані шаблони оголошень про вакансію, інтеграції, підбір кандидатів за допомогою штучного інтелекту, автоматизована перевірка резюме та

інструменти для співпраці роблять його хорошим вибором для рекрутерів, яким потрібна комплексна система відстеження кандидатів, яка допоможе їм ефективно шукати та наймати найкращий талант (Рис. 2.5).

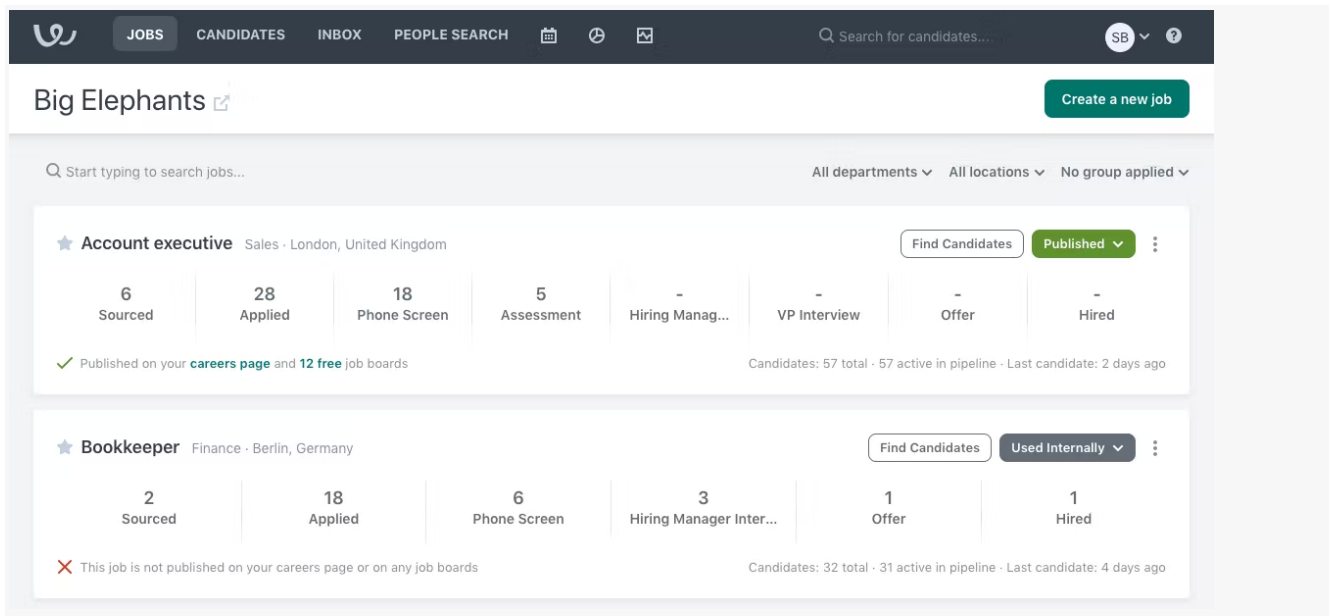


Рис. 2.4 Панель статистик за оголошеннями на Workable

Платформа побудована з використанням ряду технологій, включаючи кілька мов програмування. Ось деякі з мов програмування та технології, які використовуються для розробки Workable (зазначемо, що виявити мову на бек-енді не можливо, якщо інформації немає у вільному доступі):

- JavaScript – використовує для веб-інтерфейсу та різноманітних функцій на стороні клієнта;
- HTML і CSS – є основними мовами інтернету, за допомогою яких створено інтерфейс користувача. HTML використовується для структурування вмісту та визначення макета веб-сторінок, тоді як CSS використовується для стилізації та презентації;
- платформа хмарних обчислень, яка надає ряд послуг та інструментів для створення та розгортання хмарних програм. Використовується для забезпечення високої доступності, масштабованості та безпеки платформи;

- інтеграція зі Stripe – платформа обробки платежів та платіжних транзакцій за планами платної підписки.

Workable використовує технології штучного інтелекту (AI) і машинного навчання (ML) щоб допомогти роботодавцям оптимізувати процеси найму, скоротити час найму та приймати більш обґрунтовані рішення щодо найму. Наприклад:

- для перевірки резюме на основі штучного інтелекту Workable використовує алгоритми машинного навчання для аналізу та визначення найбільш кваліфікованих кандидатів на основі навичок, досвіду та інших факторів;
- підбір під оголошення про роботу з найбільш підходящі для кандидатів;
- прогностична аналітика використовується щоб допомогти роботодавцям приймати кращі рішення щодо найму;
- чат-боти використовують обробку природної мови (NLP), щоб взаємодіяти з кандидатами та надавати їм відповідну інформацію про вакансії, статус заявки та інші теми, пов'язані з працевлаштуванням.

Ці мови програмування та технології працюють разом, щоб створити масштабовану та ефективну платформу керування підбором персоналу.

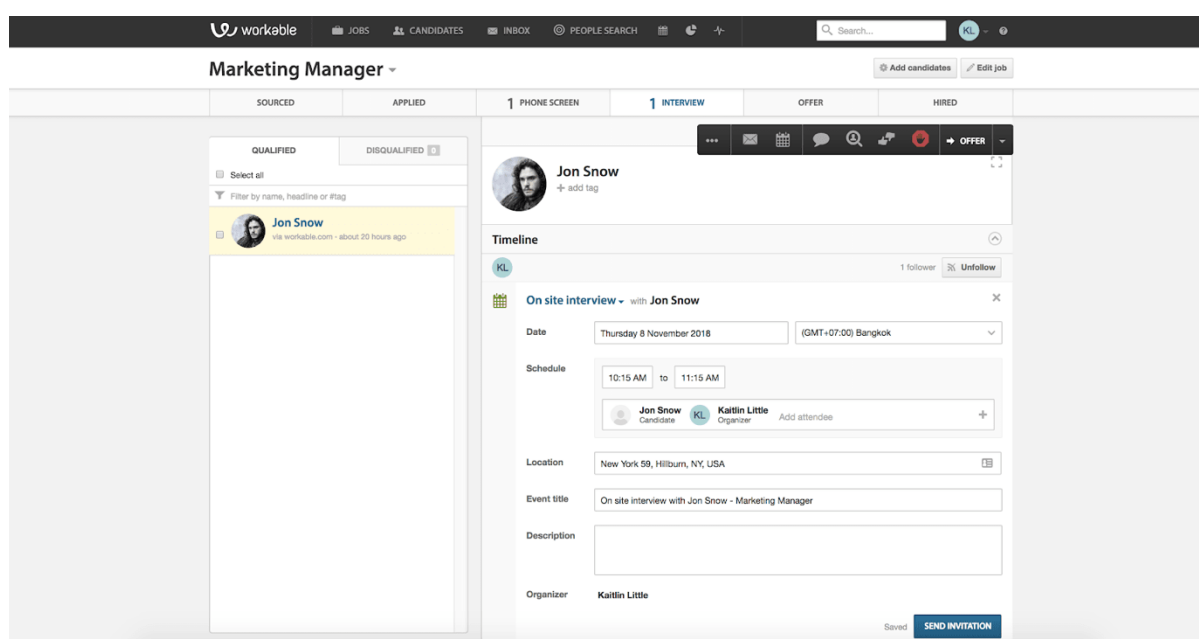


Рис. 2.5 Профіль кандидата на Workable

2.2.3. HackerRank

Автоматизована платформа для оцінки технічних навичок кандидатів на посади в галузі ІТ. Вона дозволяє компаніям швидко та ефективно перевіряти технічні знання кандидатів та відбирати тільки найбільш кваліфікованих кандидатів для подальшого співбесідування.[8]

Основною функцією платформи є проведення онлайн-тестів, де кандидатам пропонуються завдання з різних галузей ІТ, включаючи програмування, бази даних, аналіз даних, машинне навчання та багато іншого (Рис. 2.6). Тести на HackerRank можуть бути налаштовані під конкретні вакансії, що дозволяє оцінювати технічні знання кандидатів, які відповідають конкретній посаді.

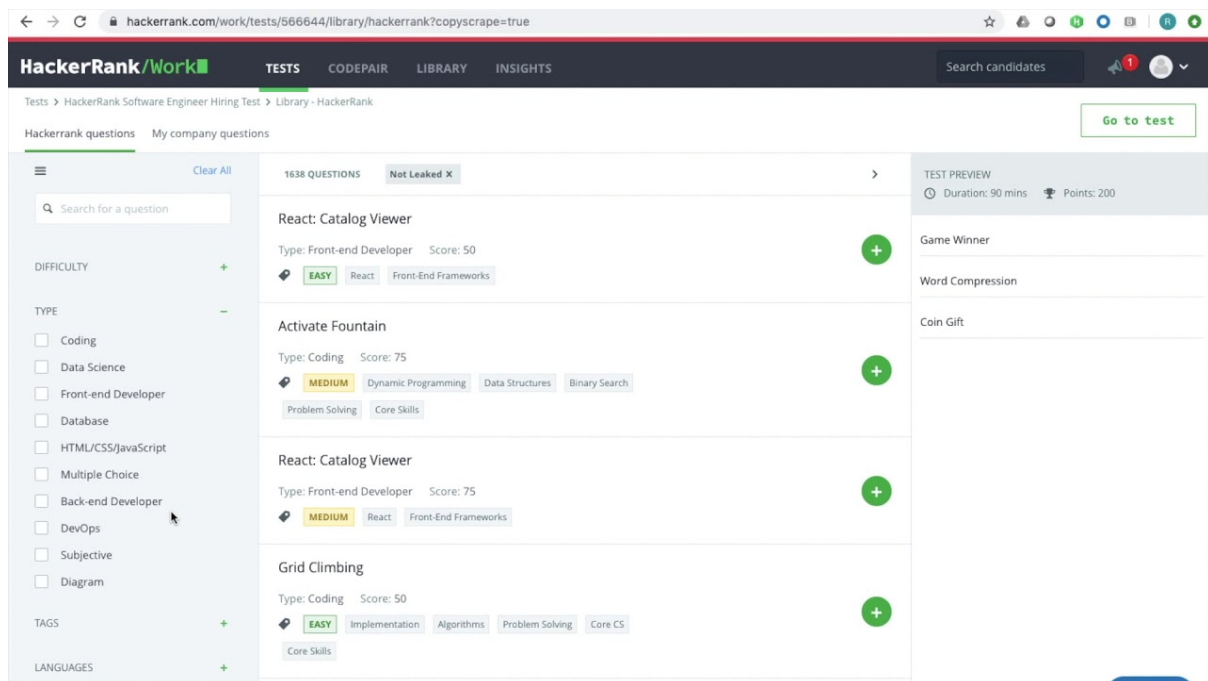


Рис. 2.6 Панель управління тестами у HackerRank

HackerRank також надає компаніям можливість проводити перегляд коду з використанням інструментів, які дозволяють швидко оцінювати код кандидатів (Рис. 2.7). Після проходження тестування та перегляду коду, кандидатам може

бути надісланий детальний звіт про їхні досягнення та рекомендації щодо подальших дій.

Однією з основних переваг HackerRank є можливість проведення онлайн-тестування, що дозволяє компаніям швидко та ефективно оцінювати кандидатів з різних куточків світу. Крім того, платформа має велику базу даних завдань, що дозволяє налаштовувати тести для різних вакансій та рівнів складності.

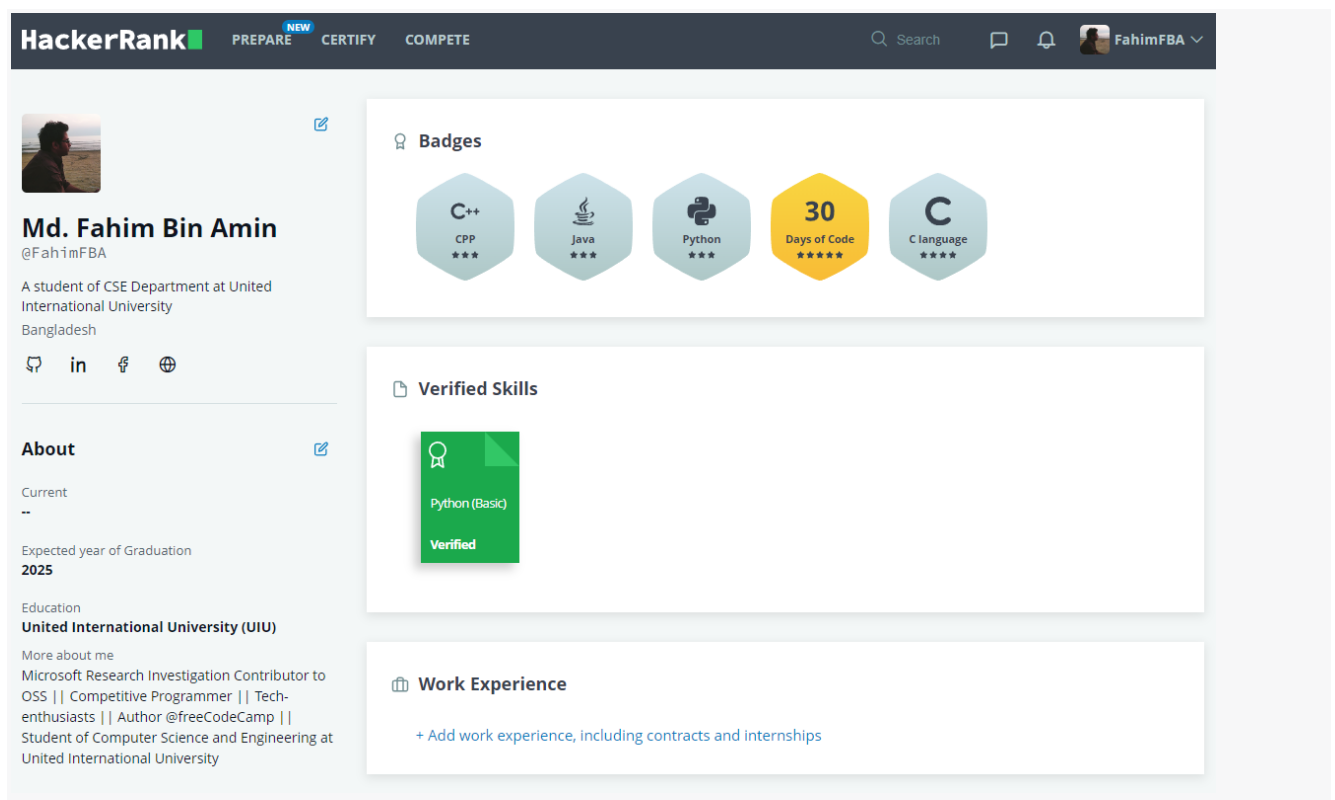


Рис. 2.7 Профіль кандидата на HackerRank

Платформа розроблена з використанням різноманітних технологій, зокрема (зазначемо, що виявити мову на бек-енді не можливо, якщо інформації немає у вільному доступі):

- HTML, CSS і JavaScript. Ці технології використовуються для створення інтерфейсу користувача та взаємодії з користувачем платформи;

- Хмарні платформи, що надають масштабовані обчислювальні ресурси, сховище та інші послуги, необхідні для ефективної роботи платформ;
- HackerRank збирає та аналізує дані про поведінку користувачів, продуктивність та інші показники;
- алгоритми штучного інтелекту та машинного навчання які розширюють можливості тестування та оцінювання кандидатів кількома способами. Ось кілька прикладів:
 - автоматизоване оцінювання коду: аналіз коду кандидата та надання зворотного зв'язку в реальному часі щодо його якості, правильності та ефективності;
 - інтелектуальне виявлення плагіату: платформа порівнює код кандидата з великою базою даних існуючого коду та позначає будь-які випадки копіювання чи схожості;
 - адаптивне тестування: аналізує відповіді кандидата на попередні запитання та коригує рівень складності наступних запитань на основі їх ефективності;
 - виявлення талантів: аналізує результативність кандидатів у програмуванні та надає рекрутерам ранжований список найефективніших кандидатів на основі їхніх навичок і здібностей кодування.

Загалом технологічний стек платформи створений для забезпечення масштабованого та надійного рішення для найму технічних працівників, а також забезпечує гнучкість та інтеграцію з іншими інструментами та системами.

2.3. Висновки та порівняння ATS

Платформи LinkedIn Talent Solutions, Workable та HackerRank for Work можна порівняти за такими критеріями (див. табл.2.3):

- HackerRank for Work має більш спеціалізовану функціональність для оцінювання технічних навичок кандидатів;
- Workable є більш доступною з точки зору вартості, оскільки має плани для різних розмірів компаній з відповідними ціновими пакетами. LinkedIn Talent Solutions та HackerRank for Work пропонують індивідуальні цінові пропозиції для кожної компанії. Ця ATS також має найкращий функціонал для розміщення оголошень на інших платформах;
- LinkedIn має найбільшу базу даних профілів кандидатів та зв'язків між ними, що дає змогу швидко знайти відповідного кандидата. Однак, HackerRank for Work має спеціалізовану аудиторію програмістів, які можуть бути важливі для компаній, які шукають IT-спеціалістів.
- інтеграції: Workable та LinkedIn Talent Solutions пропонують більш широкі можливості для інтеграції з іншими програмами для рекрутингу, такими як збір резюме з Indeed та ZipRecruiter. HackerRank for Work має менше інтеграцій з іншими програмами, але має можливість інтеграції зі Slack для спілкування з командою.
- користування: всі три платформи мають простий та зрозумілий інтерфейс для користувачів. Проте, Workable має більш сучасний та інтуїтивний дизайн інтерфейсу.

Workable, LinkedIn Talent Solutions і HackerRank пропонують різні підходи до тестування й оцінки кандидатів із різними сильними й слабкими сторонами залежно від конкретних потреб і цілей організації.

LinkedIn Talent Solutions пропонує ряд інструментів для тестування та оцінки кандидатів, рекомендації та оголошення про роботу. Однією з ключових переваг LinkedIn є його величезна мережа професіоналів, яка полегшує пошук кваліфікованих кандидатів і отримання рекомендацій. Однак платформа може бути менш придатною для організацій, які потребують більш індивідуальних методів тестування чи оцінювання.

Workable пропонує набір функцій для тестування кандидатів, включаючи оцінювання навичок, особистісні тести та моделювання роботи. Ці тести розроблені, щоб допомогти рекрутерам оцінити навички, знання та особисті якості кандидатів. Однією з переваг Workable є його гнучкість, оскільки його можна налаштувати відповідно до конкретних вимог конкретної роботи чи організації. Але платформа так само менш придатна для більш спеціалізованих технічних ролей.

HackerRank: HackerRank пропонує низку тестів і оцінок програмування, призначених для оцінки навичок і здібностей кандидатів у програмуванні. Платформа використовує алгоритми штучного інтелекту та машинного навчання для надання зворотного зв'язку в режимі реального часу та підрахунку балів за завдання кодування, що дозволяє рекрутерам швидко визначати найкращих кандидатів. Однією з переваг HackerRank є його зосередженість на технічних навичках і досвіді, завдяки чому він добре підходить для технічних ролей, які вимагають спеціальних знань програмування. Однак платформа може бути менш придатною для нетехнічних ролей або ролей, які потребують ширших навичок і знань.

Підсумовуючи, Workable, LinkedIn Talent Solutions і HackerRank мають різні сильні та слабкі сторони, коли справа доходить до тестування й оцінювання кандидатів. Вибір платформи залежатиме від конкретних потреб і цілей організації, а також характеру посади, яку виконують. Організації, які потребують більш індивідуальних методів тестування або оцінки, можуть віддати перевагу Workable, а ті, хто зосереджений на технічних ролях, можуть віддати перевагу HackerRank. Тим часом LinkedIn Talent Solutions може краще підійти для організацій, які бажають підключитися до великої мережі професіоналів, щоб отримати рекомендації та можливості для спілкування.

На основі підсумків вище та складених з них таблиці (Таб. 2.3.) було виявлено, що кожна з платформ має свої слабкі сторони, що заважає їм бути цілком універсальними.

Таблиця 2.3.

	Мережа професіоналів	Тестування особистих навичок	Тестування технічних навичок	Гнучкість	Призначення інтерв'ю	Оцінка резюме	Публікація вакансій на інших ресурсах	Ціна	Інтеграції	Інтерфейс
LinkedIn Talent Solutions	5	5	2	3	5	5	2	2	5	3
Workable	3	5	2	5	5	5	5	3	5	5
HackerRank	2	4	5	4	5	5	3	2	2	4

2.4. Порівняння існуючих технологій

ATS (Application Tracking Systems) вважається програмною платформою, оскільки це система програмних інструментів, які розроблені, щоб допомогти керувати процесом найму. Ці системи, як правило, є веб-інтерфейсом і надають роботодавцям централізовану платформу для керування оголошеннями про вакансії, відстеження резюме та заяв кандидатів, спілкування з кандидатами та оцінювання претендентів на роботу.

Існує кілька мов кодування та технологій, які зазвичай використовуються для розробки веб-платформ. Найкращий вибір залежить від конкретних вимог проекту, досвіду команди розробників і цілей платформи. Ось деякі з найпопулярніших варіантів:

- Java є однією з найпоширеніших мов програмування для розробки програмного забезпечення. Він відомий своєю масштабованістю, портативністю та простотою використання. Java використовується в різноманітних програмах, включаючи веб-розробку, мобільну розробку та розробку корпоративного програмного забезпечення;
- Python є популярним вибором для розробки програмного забезпечення завдяки своїй простоті та універсальності. Він використовується для веб-розробки, науки про дані, машинного навчання, штучного інтелекту тощо. Python відомий своєю читабельністю, що полегшує його вивчення та використання;

- JavaScript — це мова програмування, яка використовується переважно для веб-розробки. Він використовується для створення динамічних веб-сторінок, інтерфейсів користувача та веб-додатків. JavaScript також використовується для серверного програмування з Node.js;
- C# — сучасна мова програмування, розроблена Microsoft. Вона використовується для створення настільних програм Windows, веб-програм та ігор. C# відомий своєю простотою та легкістю використання, що робить його популярним вибором для розробників;
- PHP — мова сценаріїв на стороні сервера, яка використовується для веб-розробки. Він використовується для створення динамічних веб-сторінок, веб-додатків і платформ електронної комерції. PHP є відкритим кодом і широко використовується;
- Ruby — це динамічна об'єктно-орієнтована мова програмування, популярна для веб-розробки. Вона використовується з інфраструктурою Ruby on Rails для швидкого й ефективного створення веб-додатків;
- Node.js є популярним вибором для розробки програмних платформ, особливо тих, які включають створення веб-додатків або веб-сервісів. Node.js — це середовище виконання JavaScript, створене на движку JavaScript V8 Chrome, що дозволяє розробникам використовувати JavaScript на стороні сервера.

Загалом, найкращий вибір мови та технології для розробки програмної платформи залежить від конкретних потреб проекту. Наприклад, якщо комунікація в реальному часі важлива, Node.js може бути хорошим вибором, а якщо швидкий розвиток є пріоритетом, Ruby on Rails може бути хорошим вибором. Важливо розглянути сильні та слабкі сторони кожного варіанту та вибрати той, який найкраще відповідає вимогам проекту.

РОЗДІЛ 3. ПРОГРАМНЕ РІШЕННЯ

3.1. Постановка задачі

За основу було взяту вже існуючу внутрішню CRM для збору та збереження інформації про кандидатів Recruiter. Платформа використовувалася для ручного перегляду кожного кандидата, зареєстрованого на платформі, і використання їх особистих даних для зв'язку та ручного планування раундів співбесід, включаючи онлайн-тести на кодування, перевірку резюме, порівняння досвіду та результатів тестування. Основним завданням цієї дипломної роботи було покращити існуючий цикл найму у платформі, заощадити зусилля співробітників і гроші компанії, швидше найняти якісних кандидатів. Крім того, зробити цю платформу привабливою для потенційних покупців, якими будуть кадрові агентства та компанії, які хочуть заощадити гроші на внутрішньому підборі персоналу.

Після ретельного аналізу існуючих ATS та порівняння їх функціональних можливостей було зроблено висновок, що немає єдиного типу, що б мав усі кроки на високому рівні, а деякі все ще не мають повного можливого функціоналу. На сьогодні, для того, щоб отримати кращі результати з найму за найкоротший час, існуючі ATS треба інтегрувати між собою та іншими додатковими сервісами, що робить розуміння користування більш ускладненим, а саму ціну найму далеку від меж доступності. Тому було вирішено розробити більш компактну, зручну та доступну за ціною версію, яка може використовуватися малими та середніми компаніями. А головне зробити її універсальною для технічних і нетехнічних посад, та мати повний можливий функціонал.

Також було вирішено з метою заощадження коштів на розробку та швидший вихід на ринок використовувати готові рішення. Рішення також базувалося на логіці використанні перевірених ринком і часом сервісів, які вузько спеціалізуються в певній галузі і можуть забезпечити високий рівень результатів.

Для розробки програмного продукту вибір засобів та інструментів є дуже важливим етапом. Після проведення аналізу існуючих технологій було вирішено, що для написання дипломної роботи буде використано наступні:

- PHP;
- Laravel ;
- HTML;
- JavaScript;
- MySQL.
- SpeechAce
- Affinda

В загальному, вибір засобів та інструментів був здійснений на основі потреб проекту, можливостей розробника та особливостей середовища, на якому буде виконуватись програмний продукт.

3.1.1. PHP

PHP (англ. PHP: Hypertext Preprocessor — PHP: гіпертекстовий препроцесор) - це скриптова мова програмування, яка призначена для розробки веб-рішень. Вона виконується на стороні веб-сервера, і PHP-код інтерпретується в гіпертекстову розмітку HTML, яка передається на сторону клієнта. Оскільки PHP-код виконується на сервері, а не на стороні клієнта, кінцевий користувач системи не може бачити PHP-коду, на відміну від JavaScript.

Ця особливість PHP має свої переваги і недоліки. З одного боку, вона забезпечує більш високий рівень безпеки, оскільки кінцевий користувач не може бачити код. З іншого боку, це може негативно вплинути на інтерактивність сторінок, оскільки вони не можуть взаємодіяти з користувачем на стороні клієнта. Проте, цю проблему можна вирішити, використовуючи PHP для генерації JavaScript-коду, який виконується на стороні клієнта. Це дозволяє створювати більш інтерактивні сторінки, які взаємодіють з користувачем на стороні клієнта,

але при цьому зберігається високий рівень безпеки, оскільки PHP-код залишається на сервері і не доступний для кінцевого користувача.

PHP є однією з найпопулярніших мов програмування, яка використовується для створення веб-додатків. Одна з причин її популярності полягає у тому, що вона може бути легко вбудована в HTML код веб-сторінок, що дозволяє створювати динамічні веб-сторінки з високою швидкістю і ефективністю. Крім того, PHP має велику кількість готових функцій і бібліотек, що дозволяє програмістам ефективно використовувати час для створення веб-додатків.

PHP також є досить зрозумілою для програмістів, що працюють в інших областях, оскільки вона має схожий синтаксис з мовами програмування, такими як C та Perl. Це дозволяє програмістам швидко освоювати PHP та знижує час, потрібний для вивчення мови. Зараз PHP використовується для створення різноманітних веб-додатків, включаючи блоги, електронні магазини, програмних платформ.[12]

Звісно, якщо виникає потреба у виконанні великих проєктів з високою навантаженістю, то можуть виникати питання щодо продуктивності PHP. Але в цьому випадку розробники можуть використовувати інші рішення, такі як компіляція PHP-скриптів, щоб забезпечити високу продуктивність.

Оператори - це конструкції мови програмування, які дозволяють виконувати певні дії над змінними або константами, які називаються операндами. У PHP оператори можуть бути унарні, бінарні або тернарні в залежності від кількості операндів, які вони приймають. Асоціативність оператора визначає порядок виконання операцій, які містять більше одного оператора, і може бути ліво або правоасоціативною. Пріоритетність операторів визначає порядок виконання операцій і її можна змінювати за допомогою використання дужок.

Модульний підхід до побудови програмних комплексів в PHP дозволяє виконувати розділення коду на окремі модулі з різними функціональними можливостями. Це сприяє збільшенню зручності та швидкості розробки, зменшенню кількості помилок та покращенню підтримки коду в майбутньому.

Функції, що містяться у під'єднаних до скрипта файлах модулів, можуть повертати значення для подальшого використання у програмі.

Ось ще деякі переваги використання PHP для розробки ATS:

1. Безкоштовний і відкритий код: PHP — це мова сценаріїв із відкритим вихідним кодом, яка є у вільному доступі. Це означає, що його можна використовувати для розробки ATS, не сплачуючи жодних ліцензійних зборів.
2. Велика спільнота та підтримка: має величезну спільноту розробників, що означає, що ви можете легко знайти допомогу та підтримку, якщо у вас виникнуть проблеми під час процесу розробки.
3. Швидко та ефективно: розроблено для роботи на веб-серверах, що означає, що він оптимізований для веб-розробки. Це також швидка та ефективна мова сценаріїв, яка може обробляти великий обсяг даних.
4. Інтеграція з базами даних: може легко інтегруватися з різними базами даних, включаючи MySQL, Oracle і PostgreSQL. Це робить його ідеальним вибором для розробки ATS, яка вимагає зберігання та отримання великих обсягів даних.
5. Безпека: має вбудовані функції безпеки, які можуть допомогти захистити вашу програму від різних загроз безпеці, включаючи впровадження SQL і міжсайтовий сценарій (XSS).
6. API: Однією з його ключових сильних сторін є здатність PHP інтегруватися з широким спектром API (інтерфейсів прикладного програмування). API дозволяють розробникам зв'язувати свої додатки PHP з іншими службами та платформами, такими як сайти соціальних мереж, платіжні шлюзи та системи керування контентом.

3.1.2. Laravel

Для нашої платформи ми обрали саме Laravel — це безкоштовна веб-платформа PHP із відкритим вихідним кодом, призначена для створення веб-додатків за архітектурним шаблоном Model-View-Controller (MVC). Він був створений Тейлором Отвеллом у 2011 році і став одним із найпопулярніших фреймворків PHP для веб-розробки.

Laravel прагне зробити веб-розробку простішою та швидшою, надаючи елегантний синтаксис, потужний набір інструментів і узгоджену структуру для організації коду. Він включає такі функції, як маршрутизація, проміжне програмне забезпечення, міграції баз даних, механізм створення шаблонів і ORM (Object-Relational Mapping) під назвою Eloquent.

Laravel обробляє запити та відправляє відповіді в послідовності, яка називається життєвим циклом запиту (Рис. 3).

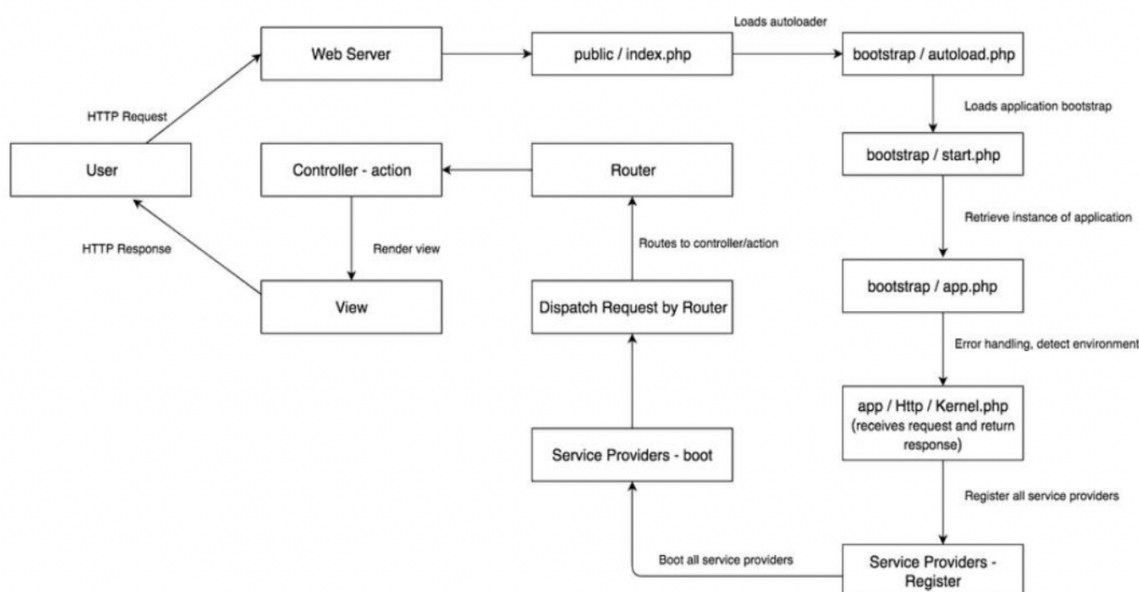


Рис. 3 Цикли запитів Laravel

Основні переваги Laravel:

- Автентифікація та авторизація: Laravel має вбудовані функції, які спрощують реалізацію безпечної автентифікації користувача та керування

доступом на основі ролей. Це важливо для ATS, яка може вимагати різних рівнів доступу для різних користувачів;

- містить інструмент об'єктно-реляційного відображення (ORM) під назвою Eloquent, який спрощує взаємодію з базою даних, дозволяючи розробникам працювати з базами даних за допомогою простого та інтуїтивно зрозумілого синтаксису. Це полегшує виконання типових операцій з базою даних, таких як вставка, оновлення та видалення записів;
- містить інтерфейс командного рядка (CLI) під назвою Artisan, який надає розробникам набір інструментів для генерації шаблонного коду, запуску тестів та виконання інших поширених завдань. Це може допомогти пришвидшити час розробки та підвищити продуктивність;
- містить механізм шаблонів під назвою Blade, який використовується для розробки ієрархічних блоків і макетів із попередньо визначеними блоками, які містять динамічний вміст;
- може легко інтегруватися з іншими технологіями, такими як платіжні шлюзи, служби електронної пошти та хмарне сховище. Це полегшує додавання нових функцій до ATS та покращення його функціональності.

Загалом Laravel є чудовим вибором для розробки ATS завдяки своїм надійним функціям і простоті використання. Його вбудовані функції автентифікації та авторизації, Eloquent ORM і механізм створення шаблонів Blade можуть допомогти прискорити час розробки та підвищити продуктивність програми. Крім того, його інтеграція з іншими технологіями та велика спільнота та підтримка роблять його ідеальним вибором для розробників, які прагнуть створити надійну та масштабовану ATS.

3.1.3. JavaScript

JavaScript — це мова програмування, яка зазвичай використовується у веб-розробці для додання інтерактивності веб-сайтам. Спочатку вона була

створена Бренданом Айхом у Netscape у 1995 році і з тих пір стала однією з найпопулярніших мов програмування у світі.

JavaScript — це мова сценаріїв на стороні клієнта, що означає, що він виконується у веб-браузері користувача, а не на сервері. Вона використовується для створення динамічних ефектів на веб-сторінках, таких як спливаючі вікна, випадаючі меню та анімація. JavaScript також можна використовувати для програмування на стороні сервера за допомогою таких платформ, як Node.js.

JavaScript часто використовується в поєднанні з HTML і CSS для створення інтерактивних і динамічних веб-сайтів. Це універсальна та гнучка мова, яку можна використовувати для широкого спектру програм, включаючи веб-розробку, настільні програми та розробку програм для мобільних пристроїв.

Ось кілька причин, чому JavaScript чудово підходить для розробки програмної платформи:

1. Універсальність: JavaScript — це мова з кількома парадигмами, що означає, що він підтримує як об'єктно-орієнтований, так і функціональний стилі програмування. Це робить її універсальною мовою, яку можна використовувати для широкого спектру додатків, від веб-розробки до розробки мобільних додатків і навіть додатків для настільних ПК.
2. Сумісність між платформами: JavaScript можна виконувати на багатьох платформах, включаючи веб-браузери, сервери, настільні середовища та мобільні пристрої. Це дозволяє легко створювати програмні платформи, які бездоганно працюють на різних платформах і пристроях.
3. Велика екосистема: JavaScript має величезну екосистему бібліотек, фреймворків та інструментів, які можна використовувати для спрощення та прискорення розробки програмного забезпечення. Деякі популярні фреймворки JavaScript для розробки платформ включають Node.js, React, Angular і Vue.js.
4. Підтримка спільноти: у JavaScript є велика та активна спільнота розробників, які постійно створюють і діляться новими інструментами та

ресурсами, щоб зробити розробку програмного забезпечення простішою та ефективнішою. Ця підтримка спільноти може бути неймовірно цінною для розробки платформи, оскільки розробники можуть вчитися та співпрацювати з іншими, хто працює над подібними проектами.

Загалом JavaScript є чудовим вибором для розробки програмної платформи, оскільки він пропонує універсальність, крос-платформну сумісність, велику екосистему інструментів і ресурсів, а також сильну підтримку спільноти.

3.1.4. СКБД MySQL

MySQL є однією з найпоширеніших вільних систем керування реляційними базами даних. Вона була розроблена з метою підвищення швидкодії обробки великих баз даних та являє собою альтернативу комерційним системам управління базами даних. MySQL використовується переважно для створення динамічних веб-сторінок, оскільки має чудову підтримку з боку різноманітних мов програмування.

MySQL характеризується високою швидкістю, стійкістю та простотою використання. Є компактним багатопотоковим сервером баз даних, що дозволяє працювати з базами даних у режимі реального часу. Крім того, MySQL має ряд важливих функцій, таких як підтримка транзакцій, контроль доступу до даних та можливість забезпечення резервного копіювання.

Вона має відкритий код, що дозволяє користувачам вносити внески у розвиток системи та покращувати її функціональність. Також це дозволяє широко використовувати систему в проектах з відкритим кодом.

MySQL є гарним рішенням для малих та середніх застосувань, оскільки вона має низькі вимоги до апаратного забезпечення та є легко налаштованою і простою у використанні. Найбільш повні можливості сервера виявляються в UNIX-системах, де є підтримка багатопоточності. Багатопоточність дозволяє

серверу MySQL обробляти більше запитів одночасно, що підвищує продуктивність системи в цілому.

Крім того, MySQL має ряд інших переваг, таких як можливість встановлення на локальному комп'ютері, використання в хмарному середовищі або віртуальному просторі. Вона також має велику кількість різноманітних додатків та розширень, що дозволяє розширювати її функціональність та пристосовувати до потреб користувача.

Сервер MySQL має багато можливостей, серед яких наступні:

- може бути використаний для зберігання великої кількості даних, що можуть бути легко масштабовані. MySQL підтримує клієнт-серверну архітектуру, що дозволяє використовувати декілька серверів для обробки великих обсягів даних;
- дуже швидкий в роботі та може обробляти велику кількість запитів на секунду. Він підтримує індекси та ряд інших оптимізацій, які допомагають забезпечити високу продуктивність бази даних;
- має різні засоби для забезпечення безпеки бази даних, включаючи внутрішні системи автентифікації та авторизації. Він підтримує різні рівні доступу та може забезпечити захист від SQL-ін'єкцій та інших видів атак на базу даних;
- підтримує транзакції, що дозволяє виконувати групу запитів як одну операцію. Це дозволяє забезпечити цілісність даних та забезпечити відмінність операцій у випадку, якщо сталася помилка в процесі виконання;
- підтримка різних типів даних, включаючи текстові, числові, дати, часи та інші. Він також підтримує роботу з бінарними та географічними даними;
- дозволяє робити резервні копії бази даних та відновлювати їх у разі необхідності. Це допомагає забезпечити безпеку даних та зменшити ризик втрати даних в разі несправності апаратного забезпечення або програмного забезпечення;

- дозволяє створювати репліки бази даних, що копіює дані з одного сервера на інший. Це допомагає зменшити навантаження на основний сервер та забезпечити високу доступність даних;
- підтримує мову SQL (Structured Query Language), що дозволяє виконувати різноманітні запити до бази даних. Це робить його дуже зручним для роботи з даними та забезпечує можливість швидко отримувати необхідну інформацію;
- підтримує різні платформи, включаючи Windows, Linux та Mac OS X. Це дозволяє використовувати його на будь-якій платформі, що зручно для розробників та користувачів;
- підтримує багато різних розширень, які дозволяють розширювати його функціональність. Це може бути корисно для розробників, які хочуть налаштувати сервер під свої потреби або додати нові функції до бази даних.

3.1.5. SpeechAce

SpeechAce — це веб-платформа, яка надає технологію розпізнавання мовлення для тих, хто вивчає мову, викладачів і компаній. Платформа використовує штучний інтелект і алгоритми машинного навчання для аналізу розмовної мови користувачів і надання миттєвого зворотного зв'язку щодо вимови, інтонації, ритму та плавності. [14]

SpeechAce пропонує широкий спектр інструментів і функцій для тих, хто вивчає мову, включаючи вправи з розпізнавання мовлення, інтерактивні уроки та персоналізований відгук. Користувачі можуть практикувати свою вимову та навички вільного мовлення, записуючи свою розмову та миттєво отримуючи відгуки про свою роботу.

Для викладачів SpeechAce надає інструмент для оцінювання навичок розмовної мови учнів. Платформа містить систему, яку можна налаштувати відповідно до певних рівнів володіння мовою та цілей навчання.

Також пропонує бізнес-рішення для компаній, яким потрібно оцінити мовні навички своїх співробітників або клієнтів. Платформу можна налаштувати відповідно до потреб різних галузей і організацій.

Загалом, SpeechAce — це комплексна платформа, яка використовує передову технологію розпізнавання мовлення, щоб допомогти тим, хто вивчає мову, покращити свою вимову та навички вільного мовлення.

SpeechAce може бути хорошим варіантом для перевірки рівня англійської мови в системі відстеження заявників (ATS) з кількох причин:

- точна оцінка навичок розмовної мови: технологія розпізнавання мовлення SpeechAce використовує штучний інтелект і алгоритми машинного навчання для точної оцінки навичок розмовної мови користувачів, включаючи вимову, інтонацію, ритм і плавність. Це може забезпечити точнішу оцінку рівня англійської мови абітурієнта порівняно з традиційними тестами, які оцінюють лише письмові навички;
- забезпечує миттєвий зворотний зв'язок щодо роботи користувачів, дозволяючи їм швидко й ефективно покращувати свої навички. Це може бути особливо корисним для заявників, які хочуть покращити свої навички англійської мови та продемонструвати свій прогрес з часом;
- систему оцінювання SpeechAce можна налаштувати відповідно до певних рівнів володіння мовою та цілей навчання. Це дозволяє організаціям адаптувати оцінку до своїх конкретних потреб і вимог;
- технологію SpeechAce можна інтегрувати в існуючі системи ATS, що робить її масштабованим рішенням для організацій, яким необхідно перевірити знання англійської мови для великої кількості заявників.
- гнучка цінова політика. SpeechAce сервіс має можливість коригувати ціни відповідно до ваших потреб, розміру компанії та кількості запитів до сервісу.

Загалом технологія SpeechAce забезпечує більш точний і ефективний спосіб перевірки знань англійської мови для організацій, які використовують систему

ATS, покращуючи процес відбору кандидатів і допомагаючи організаціям знайти найбільш кваліфікованих кандидатів.

3.1.6. Judge0

Judge0 — це онлайн-платформа для виконання коду з відкритим кодом, яка дозволяє розробникам тестувати, компілювати та запускати фрагменти коду з понад 60 мов програмування. Він був створений некомерційною організацією з метою забезпечення надійної та ефективної платформи для розробників, щоб тестувати свій код без необхідності інсталивати будь-які інструменти програмування чи бібліотеки на своїх локальних машинах.[15]

Judge0 потенційно можна використовувати в поєднанні з ATS для автоматизації та оптимізації певних частин процесу найму за рахунок такої функціональності:

- Judge0 підтримує широкий спектр мов програмування, включаючи PHP, C, C++, Python, Java, JavaScript, Ruby та багато інших. Він також надає ряд корисних функцій для розробників, таких як зворотний зв'язок у реальному часі про помилки коду та підсвічування синтаксису;
- RESTful API Judge0 можна використовувати для інтеграції функції тестування коду в ATS, дозволяючи менеджерам з найму автоматично оцінювати здібності кандидатів до програмування та відфільтровувати невідповідних кандидатів. Це може допомогти спростити процес найму та заощадити час і ресурси;
- розробник може використовувати Judge0 для створення автоматизованого набору тестів, який перевіряє навички програмування претендентів на роботу та їх здатність виконувати конкретні завдання з кодування. Це може бути особливо корисним для технічних посад, де навички кодування є важливою вимогою;
- сервіс є безкоштовним для локально розміщених додатків.

Загалом, Judge0 — це потужний і гнучкий інструмент, який надає розробникам простий спосіб тестування та запуску коду на різноманітних мовах програмування.

3.1.7. Affinda

Affinda — це компанія з аналізу даних, яка надає набір інструментів, які допомагають підприємствам отримувати та аналізувати дані з неструктурованих джерел, таких як веб-сайти, платформи соціальних мереж і оголошення про роботу. Компанія використовує вдосконалені алгоритми обробки природної мови та машинного навчання, щоб отримати цінну інформацію з цих джерел і перетворити її на структуровані дані, які можна легко проаналізувати.[16]

Рішення Affinda для машинного навчання можна використовувати для вилучення понад 100 полів із кожного резюме, упорядковуючи їх у формати файлів із можливістю пошуку. Витягнуті поля містять особисті дані, історію роботи, освіти, сертифікати, навички, резюме, мету, мови, суддів, публікації тощо. Дані кандидатів витягуються та організовуються в форматі Excel, JSON або XML. Дозволяє масово завантажувати резюме в аналізатор. Доступний на 11 мовах, в т.ч. англійська, іспанська, італійська, французька, німецька, португальська, російська, турецька, польська, індонезійська та гінді.

Affinda також має інші корисні функції, які роблять процес найму удвічі легшим, наприклад:

- пошук по всій базі резюме. Фільтрування за досвідом, званням, місцезнаходженням, кваліфікацією тощо. Кожен кандидат отримає бал на основі критеріїв, встановлених рекрутером;
- можливість проаналізувати опис роботи та скласти список кандидатів, які їй підходять;
- уміння проаналізувати резюме та скласти список схожих кандидатів;
- порівняння резюме за окремою роллю та встановлення оцінки профілю.

3.2. Опис модулю виконаного у рамках цієї дипломної роботи

3.2.1. Процес створення профілю кандидата у Recruiter

Після реєстрації та підтвердження електронної пошти кандидат переходить до заповнення особистої інформації (Рис. 3.2), технічних та особистих навичок (Рис. 3.3).

John Smith
Candidate

Candidates > User Profile

PROFILE SKILLS OFFERS CONTRACTS WORKING HOURS RESUME TEST RESULT HISTORY

Avatar

Allowed JPG, GIF or PNG.
Max size of 800kB.

Signature

Allowed JPG, GIF or PNG.
Max size of 800kB.

General

First name *
John

Last name *
Smith

Birthday
01/01/1970

Time zone
(GMT-04:00) New York

Company

Role
Candidate

Status
Verified

Profile Rate

Note

Actions

Change Password

Save

Рис. 3.2 Профіль кандидата у Recruiter

db_150 users	
id : bigint(20) unsigned	
email : varchar(255)	
email_verified_at : timestamp	
password : varchar(255)	
verify_token : varchar(255)	
status : int(11)	
role_id : bigint(20) unsigned	
specialisation_id : bigint(20) unsigned	
remember_token : varchar(100)	
created_at : timestamp	
updated_at : timestamp	
timezone : varchar(255)	
signature : longtext	
avatar : longtext	
skip_resume : tinyint(1)	
resume_id : varchar(255)	
is_tracking_time : tinyint(4)	
tracking_time_check_date : timestamp	
job_board_id : bigint(20) unsigned	

db_150 user_hard_skills	
id : bigint(20) unsigned	
user_id : bigint(20) unsigned	
skill : varchar(255)	
level : varchar(255)	
created_at : timestamp	
updated_at : timestamp	
specialisation_id : bigint(20) unsigned	
skill_id : bigint(20) unsigned	
approved : tinyint(1)	

db_150 user_soft_skills	
id : bigint(20) unsigned	
user_id : bigint(20) unsigned	
skill : varchar(255)	
level : varchar(255)	
created_at : timestamp	
updated_at : timestamp	
approved : tinyint(1)	

Таблиця 3.1

Дані зберігаються у БД, у таблицях user, user_hard_skills, user_soft_skills (див. таб. 3.1).

The screenshot displays a user profile interface for 'Recruter'. At the top, a navigation bar includes links for PROFILE, SKILLS, OFFERS, CONTRACTS, WORKING HOURS, RESUME, TEST RESULT, and HISTORY. The 'SKILLS' section is active, showing a 'Specialization' field with the text 'Community manager, project manager'. Below this, the 'HARD SKILLS LEVEL' section is divided into four columns: Junior, Middle, Senior, and Lead. Each column has an 'Add hard skill' button and a list of skills. The Junior column shows 'Copywriting'. The Middle column shows 'Docker (Software)', 'Linux', 'JavaScript (Programming Language)', 'Bash (Scripting Language)', 'Lua (Scripting Language)', 'Test-Driven Development (TDD)', and 'Cascading Style Sheets (CSS)'. The Senior column shows 'JavaScript', 'HTML', 'HTML5', 'Bootstrap', 'Figma (Design Software)', and 'Github'. The Lead column is empty. Below the hard skills, the 'LANGUAGES' section is divided into 'Conversational' and 'Fluent' columns. The 'Conversational' column shows 'Mandarin' with 'Read', 'Speak', and 'Write' buttons. The 'Fluent' column shows 'English' and 'Russian', each with 'Read', 'Speak', 'Listen', and 'Write' buttons. To the right, the 'SOFT SKILLS' section is divided into 'Middle' and 'Strong' columns. The 'Middle' column has an 'Add soft skill' button. The 'Strong' column shows 'Writing', 'Teamwork', 'Teaching', and 'Planning' buttons.

Рис. 3.3 Список навичок у профілі кандидата у Recruter

Кандидати також мають можливість завантажити резюме (що рекомендується сповіщеннями, щоб мати більше шансів бути найнятим), яке буде проаналізовано за допомогою Affinda, а їхня особиста інформація буде автоматично заповнена. Звертаються за рахунок класу AffindaResumeService, функції uploadResumeToAffinda (Рис. 3.4).

Потім кандидати переходять до оцінки коду. На платформі можна автоматично протестувати понад 60 мов коду. Кандидати кодують рішення завдань, щоб отримати кращий бал (Рис. 3.7). Наша система перевіряє вихідний код за кандидатами та надає оцінку на основі кількості правильних відповідей. Для цього ми використовуємо сервіс Judge0.

```

public function uploadResumeToAffinda() {
    //if file exists
    if(Storage::disk($this->affindaResume->local_disk)->exists($this->affindaResume->disk_filename)) {
        $file_contents = base64_encode(Storage::disk($this->affindaResume->local_disk)->get($this->affindaResume->disk_filename));
        //send request
        $response = Http::withToken($this->api_key)
            ->acceptJson()
            ->attach( name: 'file', $file_contents)
            ->post($this->endpoint);
        //response ok
        if($response->ok()) {
            $this->response = $response->json();
            $this->updateResumeWhenUploadedToAffinda($this->response['meta']);
            (new AffindaIndexService())
                ->indexDocument(config( key: 'affinda.resumes_index_name'), $this->response['meta']['identifier']);
        }
        //response failed
        elseif($response->failed()) {
            \Log::channel()->emergency($response->json());
            $this->clearAffindaDataOnFailure();
        }
        return $this;
    }
}

```

Рис. 3.4 Частина коду класу AffindaResumeService

PHP

Distinct powers

How many distinct terms are in the sequence generated by a^b for $2 \leq a \leq 100$ and $2 \leq b \leq 100$?

Solution? ⓘ

You must use function of output like console, print_r, echo, etc.

Execute

Example

How many distinct terms are in the sequence generated by a^b for $2 \leq a \leq 5$ and $2 \leq b \leq 5$? Consider all integer combinations of a^b for $2 \leq a \leq 5$ and $2 \leq b \leq 5$:

$2^2=4, 2^3=8, 2^4=16, 2^5=32$
 $3^2=9, 3^3=27, 3^4=81, 3^5=243$
 $4^2=16, 4^3=64, 4^4=256, 4^5=1024$
 $5^2=25, 5^3=125, 5^4=625, 5^5=3125$

4, 8, 9, 16, 25, 27, 32, 64, 81, 125, 243, 256, 625, 1024, 3125

The test result

The output is "15".

Action

Previous

Next

Finish Test

Progress

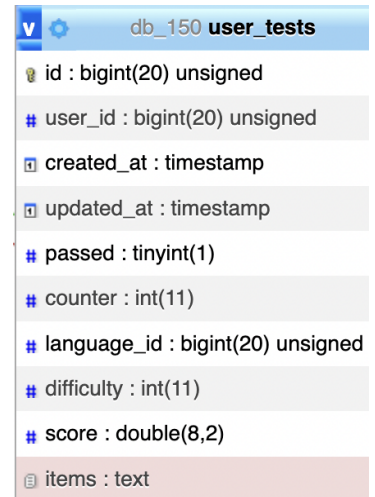
Рис. 3.5 Тестова частина для коду у профілі кандидата у Recruiter

Виконується завдяки класу UserTestController, головною функцією якого є techTestExecute (Рис. 3.6). Ця функція звертається до класу RequestData, що зі свого боку витягує данні з фіктивного клієнта Judge0, який було знайдено у виді контейнера у вільному доступі та розвернуто на приватному сервері.

Фіктивний клієнт можна використовувати, щоб перевіряти функції Judge0. Дані не перевіряються на нашій стороні. Спочатку створюється нове подання, а потім отримується статус подання кожні 1500 мс.

Результати тесту зберігаються у user_test (див таб. 3.2)

Таблиця 3.2



db_150 user_tests	
id	: bigint(20) unsigned
user_id	: bigint(20) unsigned
created_at	: timestamp
updated_at	: timestamp
passed	: tinyint(1)
counter	: int(11)
language_id	: bigint(20) unsigned
difficulty	: int(11)
score	: double(8,2)
items	: text

Наступний крок – оцінювання з англійської, де спочатку роблять відеовступ і розповідають про свій останній відповідний досвід . А потім за відведений час читають два тексти (Рис. 4.4), які будуть автоматично аналізуватися за допомогою API сервісу SpeechAce. Оцінка залежатиме від вимови, швидкості, плавності та акценту мови.

Виконується за допомогою класу UserLanguageTestController, у якому функція langTestSend пересилає данні до опрацювання. Данні оброблюються через сервіс SpeechAce до якого звертаються за допомогою класу ProcessData. Він в свою чергу повертає змінну \$processed_result - оцінку результату.

```

public function techTestExecute(Request $request, int $taskId){
    $task = UserTestTask::find($taskId);

    $currentTest = UserTest::find($task->user_test_id);

    $code = $request->get( key: 'code');

    $testRequest = new RequestData(auth()->user()->id, $taskId, $task->test->language_id);
    $result = $testRequest->sendRequest(['source_code' => $code]);
    $token = $result['token'];
    $isError = false;
    $gottedToken = false;

    while ( !$gottedToken ) {
        if(isset($result['status'])){
            if($result['status']['id'] == 3){
                $gottedToken = true;
                break;
            }
            if($result['status']['id'] > 3){
                $isError = true;
                $gottedToken = true;
                break;
            }
        }
        sleep( seconds: 3);
        $result = $testRequest->getResult($token);
    }
    if(!$isError){
        if ($result['stdout'] == $task->question->answer) {
            $task->passed = 1;
        } else {
            $task->passed = 2;
        }
    } else{
        $task->passed = 2;
    }
    if($isError){
        return response()->json([
            'success' => false,
            'resultObj'=>$result,
            'result' => (int)$task->passed,
        ]);
    }
    return response()->json([

```

Рис. 3.5 Фрагмент коду класу UserTestController

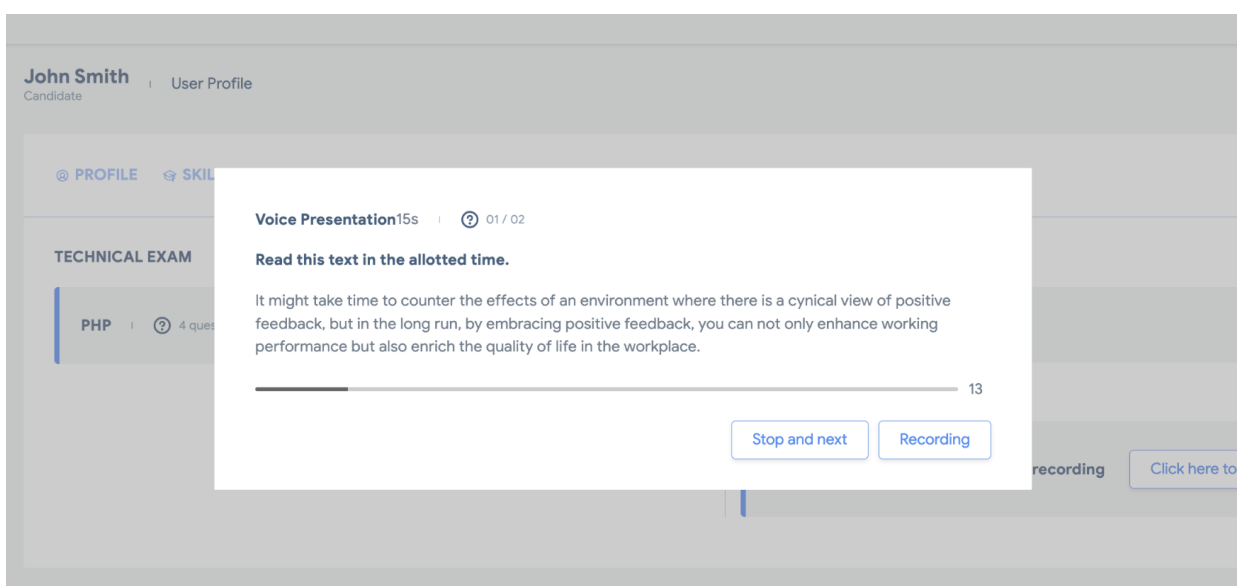


Рис. 3.6 Тестова частина для англійської у профілі кандидата у Recruiter

Таблиця 3.3

db_150 user_language_tests
id : bigint(20) unsigned
user_id : bigint(20) unsigned
test_id : bigint(20) unsigned
video_file : varchar(255)
audio_result : varchar(255)
created_at : timestamp
updated_at : timestamp
passed : tinyint(1)
name : varchar(255)
difficulty : int(11)

Результати англійського тесту зберігаються у таблиці 3.3.

Якщо кандидат пройшов відбір і отримав пропозицію про роботу, він побачить її в такому вигляді на зображенні нижче (Рис. 4.5). Кандидат може прийняти або відхилити пропозицію. За цей етап відповідають функції асерт та decline у класі OfferController.

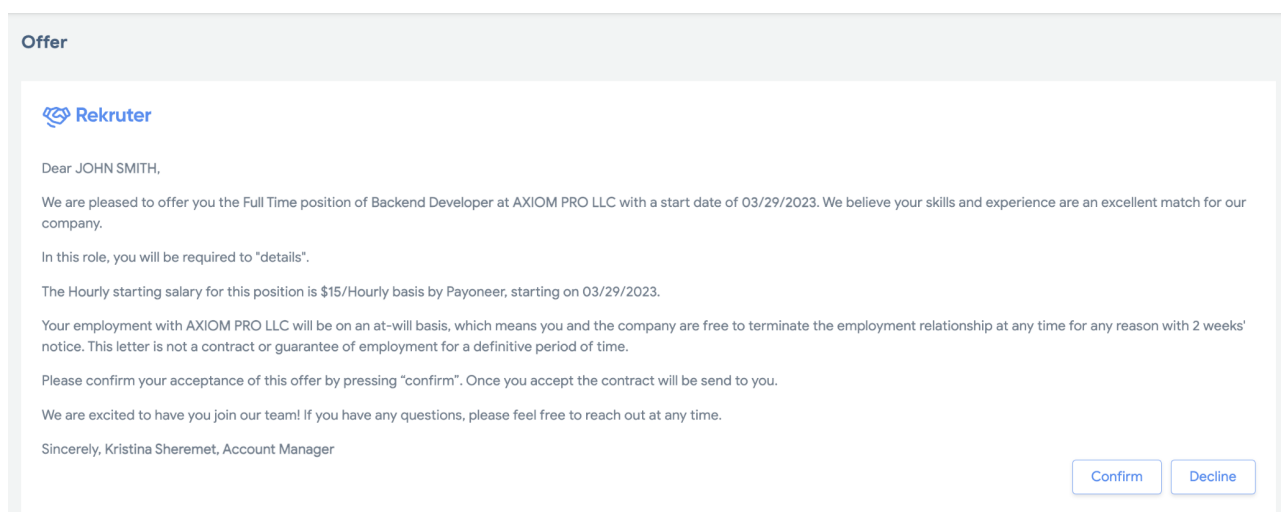


Рис. 3.7 Пропозиція яку отримує обраний кандидат

У разі прийняття пропозиції кандидат отримує контракт, який він може підписати прямо на платформі (Рис. 3.8).

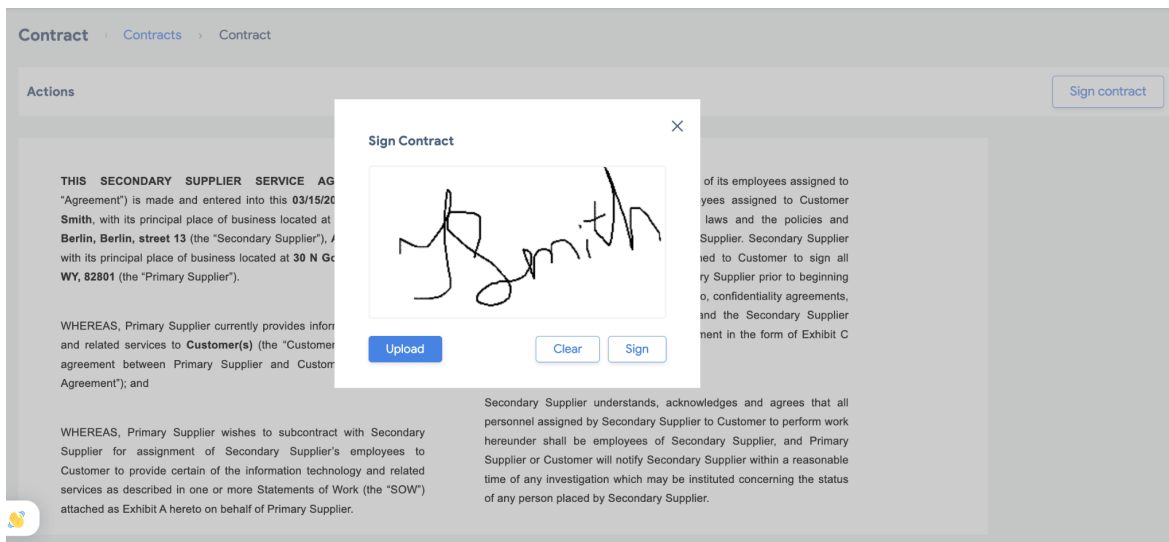


Рис. 3.8 Підписання контракту

3.3. Процес відбору кандидату рекрутером

Створення вакансій на платформі включає заповнення форми опису посади (Рис. 3.9), вибір необхідних навичок і вибір тестів (Рис. 3.10), необхідних для цієї конкретної посади. Відповідальним за цей процес є клас JobsController.

Рис. 3.9 Створення оголошення про посаду

Дані про посаду зберігаються в базі даних у таблиці vacancies (Табл. 3.4)
При публікації вакансії на інших платформах рекрутер додає посилання на сторінку реєстрації на платформі.

Таблиця 3.4

db_150 vacancies	
id	: bigint(20) unsigned
title	: varchar(255)
company_id	: bigint(20) unsigned
project_id	: bigint(20) unsigned
data	: json
skills	: json
min_salary	: int(11)
max_salary	: int(11)
active	: tinyint(1)
unique	: varchar(255)
created_at	: timestamp
updated_at	: timestamp
wp_id	: int(11)
required_tech_test_id	: int(11)

JOB

SKILLS

REQUIRED TESTS

CHOOSE REQUIRED TEST

HARD SKILLS

PHP

JavaScript

LANGUAGES

English

Рис. 3.10 Відмітка тестів необхідних до проходження при подачі на посаду

Відбір з кандидатів, що подалися на позицію здійснюється на основі результатів тестів і найбільш відповідного досвіду у % співвідношені прорахованого через API сервісу Affinda, яка працює за допомогою штучного

інтелекту та машинного навчання. До нього ми звертаємось за допомогою класу `AffindaMatchResumeJobDescriptionService`, функція `affindaMatch` (Рис 3.12). Рекрутер бачить найбільш відповідних кандидатів у верхній частині списку заявників (Рис. 3.11).

List of Apples

☐	CANDIDATE	STATUS	RESPONSIBILITY	APPLIED AT	NOTE	TECH SCORE	ENGLISH SCORE	MATCH SCORE	ACTIONS
☐	AR Amber Remedios	CV under	Axiom	January 11, 2023	Add Note	100	100	65 %	View Edit Delete
☐	WE Wondimu Ejeta	New appli	Axiom	January 13, 2023	Add Note	100	100	53 %	View Edit Delete
☐	MO Michael Orwa	CV under	Axiom	January 10, 2023	Add Note	60	73	65 %	View Edit Delete

Рис. 3.11 Список кандидатів, що подалися на позицію з кращими результатами зверху

В першу чергу відображається інформація про кандидатів у яких найвищі показники по тесту з кодування (Tech Score), англійської мови (English Score), та найвищий процент співпадіння резюме з описом вакансії (Match Score).

```
public function affindaMatch() {
    if($this->candidateCanBeEvaluated()) {
        //request data
        $this->post_data = array_merge([
            'resume' => $this->resume_identifier,
            'job_description' => $this->job_description_identifier,
        ],$this->affinda_weights);
        //send request
        $response = Http::withHeaders([
            'Authorization' => 'Bearer '.$this->api_key,
            'Accept' => 'application/json',
        ])
        ->get($this->endpoint,$this->post_data);
        //response ok
        if($response->ok()) {
            \Log::channel('affindaresumejobmatch')->info($response->json());
            $this->response = $response->json();
            $this->score = $this->response['score'];
            $this->details = $this->response['details'];
            $this->createOrUpdateModel();
        }
        //response failed
        elseif($response->failed()) {
            \Log::channel('affindaresumejobmatch')->info($response->json());
        }
    }
}
```

Рис. 3.12 Код функції `affindaMatch`

Рекрутер має можливість відкрити профіль зацікавившого кандидата, щоб детальніше ознайомитися з результатами тестування (Рис. 3.13, Рис. 3.14) та переглянути резюме, продивитися відео презентацію, якщо це необхідно. Ці дані зберігаються на платформі назавжди та можуть бути використані для майбутніх посад і потреб.

Після вибору найпривабливішого кандидата рекрутер має можливість відправити пропозицію про роботу (Рис. 3.15) зі спеціальної вкладки в профілі кандидата. Це робиться простим заповненням форми пропозиції, яка автоматично надсилається кандидатам на затвердження.

Дані про пропозицію зберігаються у таблиці offers у БД (Табл. 3.4).

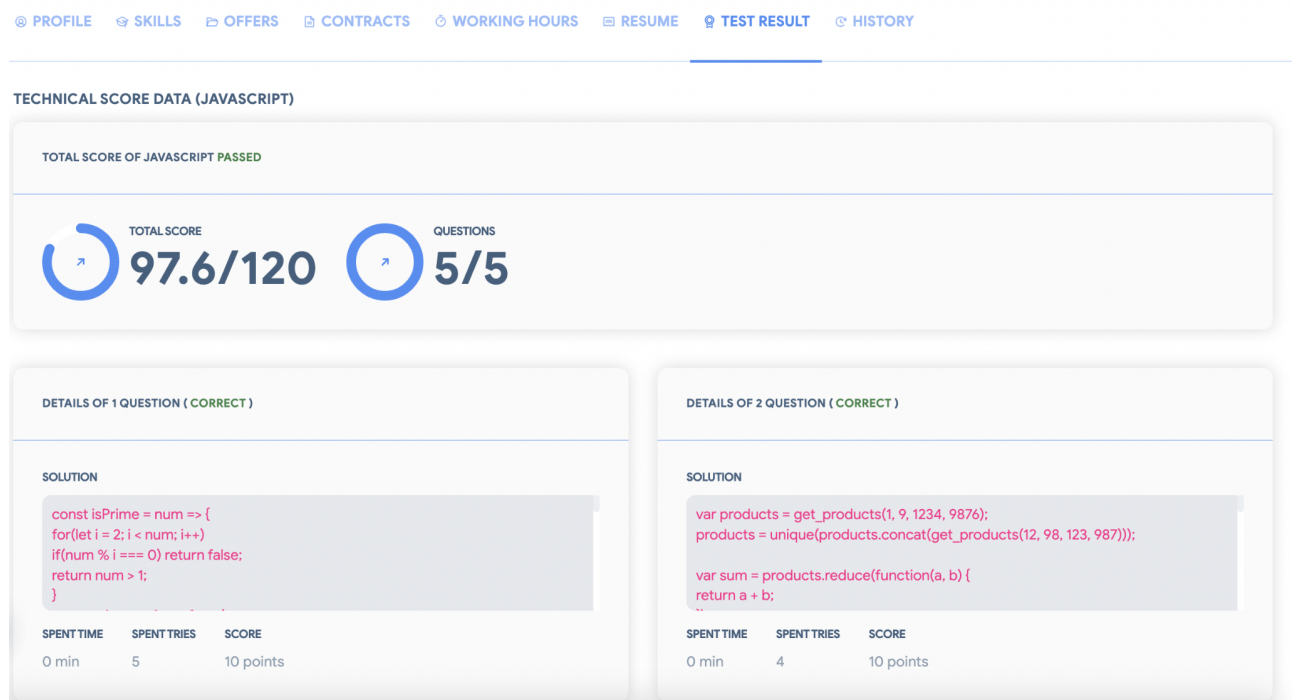


Рис. 3.13 Результати тесту з кодування у профілі кандидата

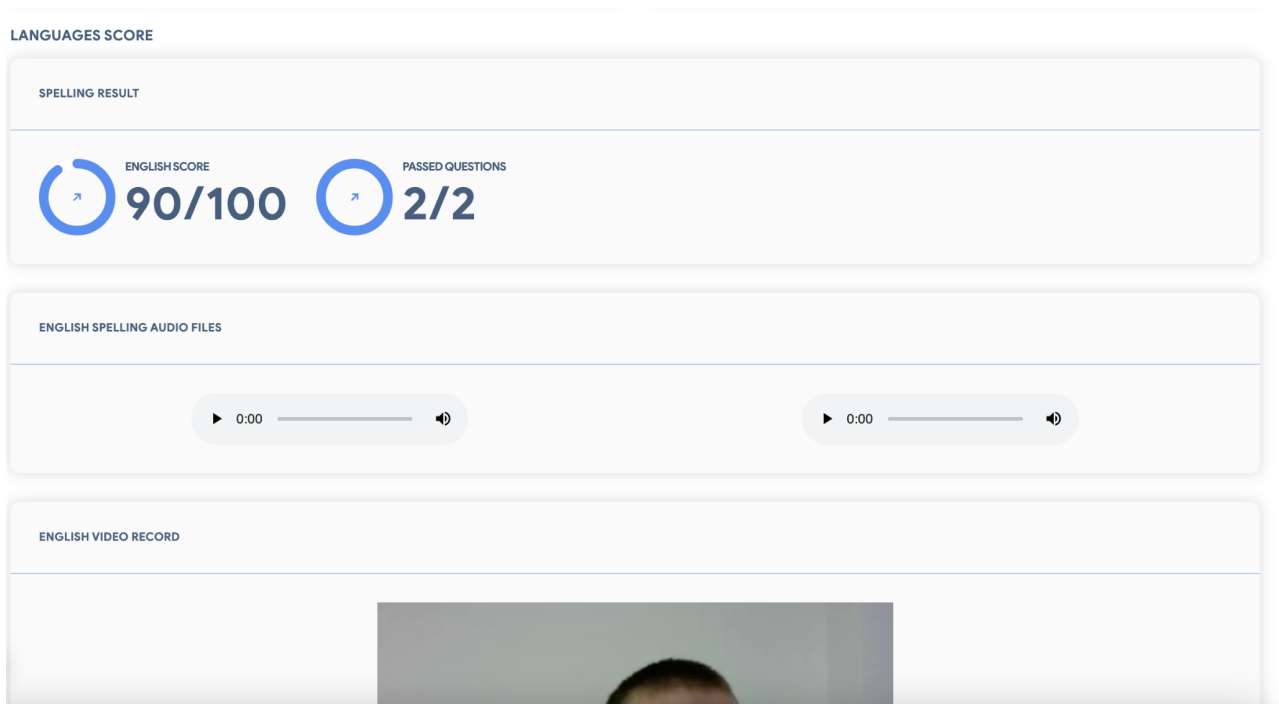


Рис. 3.14 Результати тесту з англійської мови у профілі кандидата

Рис. 3.15 Створення пропозиції для обраного кандидата

У разі згоди кандидата з пропозицією він натискає відповідну кнопку. Контракт генерується автоматично у тому ж класі що й пропозиція, OfferController, функцією offerToContract. Він складається із заздалегідь

написаного шаблону та заповнюється деталями, підтвердженими в пропозиції про роботу.

Таблиця 3.4

v	db_150 offers
🔑	id : bigint(20) unsigned
#	user_id : bigint(20) unsigned
📦	data : json
📄	status : varchar(255)
#	manager_id : int(11)
🕒	created_at : timestamp
🕒	updated_at : timestamp

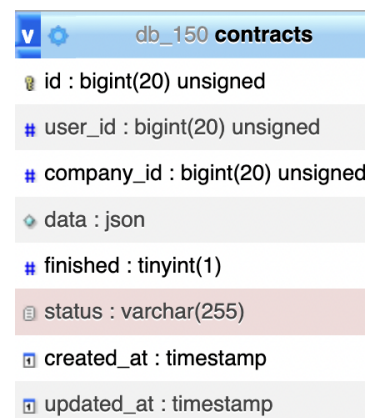
Дані про контракти зберігаються у таблиці contract (таб. 3.5).

Після підписання контракту кандидат переходить зі списку «Кандидати» в список «Співробітники».

```
public function offerToContract($offer): bool {
    # $offer = Offer::find($offerId);
    $user = User::find($offer->user_id);
    $company = Company::where('column', 'title', '=', $offer->data['company'])->first();
    $startDate = Carbon::createFromFormat('format: 'm/d/Y', $offer->data['start']); #->format('Y-m-d');
    $contract = new ContractService(
        $user,
        $company,
        $offer->data['start'],
        $offer->manager_id
    );
    $data = [
        'rate' => $offer->data['rate'],
        'period' => $offer->data['period'],
        'position' => $offer->data['position'],
        'manager' => $offer->manager_id,
        'duration' => $offer->data['duration'],
        'comment' => $offer->data['content'],
        'payment' => $offer->data['payment'],
        'startDate' => $startDate,
    ];
    $contractData = array_merge($data, $contract->prepareData());
    if (Contract::create([
        'user_id' => $offer->user_id,
        'company_id' => $company->id,
        'finished' => false,
        'data' => $contractData,
        'status' => 'New',
    ])) {
        return true;
    }
    return false;
}
```

Рис. 3.16 Код функції offerToContract

Таблиця 3.5



db_150 contracts	
id	: bigint(20) unsigned
user_id	: bigint(20) unsigned
company_id	: bigint(20) unsigned
data	: json
finished	: tinyint(1)
status	: varchar(255)
created_at	: timestamp
updated_at	: timestamp

3.4. Результати роботи Recruiter після додавання модулю

В результаті ми отримали платформу яка автоматизує 80% циклу підбору персоналу. Це насамперед відбувається за рахунок того, що кандидати заповнюють свої особисті дані, завантажують резюме, виконують тести з кодування та англійської мови. Recruiter задіює сервіси що використовують штучний інтелект і машинне навчання для аналізу результатів тестів і визначення найбільш відповідного досвіду в резюме кандидата. (Рис. 3.17) Такий підхід дозволяє рекрутерам приділяти увагу лише найбільш перспективним кандидатам і не витрачати зайвий час на підбір. Recruiter дозволяє заощадити час і гроші на процесі найму персоналу, дозволяючи зосередитися на тому, що справді важливо – розвитку бізнесу. Крім того, ця система зберігає всю інформацію про кандидатів у їхніх профілях, що дозволяє легко відстежувати прогрес і приймати обґрунтовані рішення щодо того, кого та коли найняти.

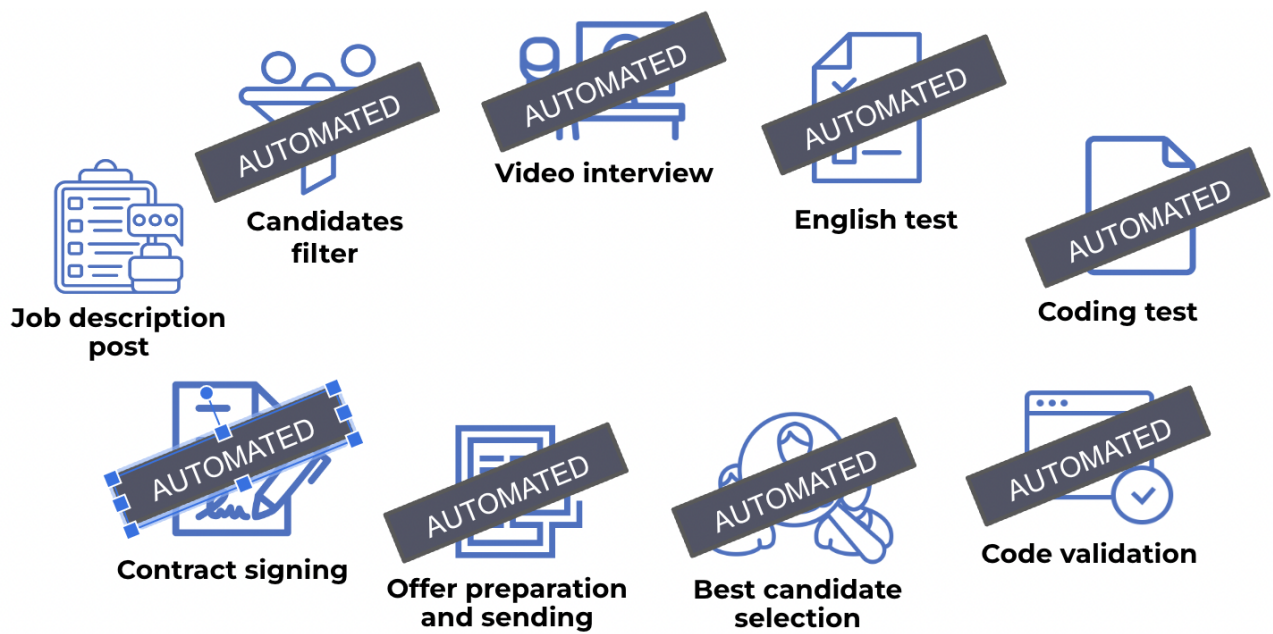


Рис. 3.17 Цикл пошуку та найму працівників з Recruiter

Висновки

Під час цього дослідження було чітко визначено, що технології рекрутингу дуже розвинені і тримаються в цьому напрямку. Процес стає все більш і більш автоматизованим, а рішення здебільшого передаються на машини. Але все ще нема єдиної платформи, що повністю б задовільняла потреби рекрутингу як технічних так і нетехнічних напрямків, мала б можливості розміщення об'яв на різноманітних платформах для пошуку роботи, велику базу кадрів, можливості підписання контракту і управління після найму на роботу.

У цій дипломній роботі ми розробили більш універсальний та бюджетну модуль системи відстеження заявників шляхом оптимізації більшої частини циклу, що покращує процес прийняття рішень і економить значну частину витраченого часу та зусиль.

Також було визначено багато способів потенційних майбутніх покращень для забезпечення найкращого досвіду з найму на платформі. Наприклад, оптимізація для мобільних пристроїв. Зі збільшенням використання смартфонів і планшетів оптимізація систем ATS для мобільних пристроїв є надзвичайно важливою. Обробка природної мови (NLP), що аналізують описи посад і профілі кандидатів і попереджають упередженість у виборі, якщо посада або описаний досвід мають інше формулювання. Інтеграції для автоматичної публікації посад на можливі платформи для пошуку роботи. Також, інтеграції з соціальними медіа: LinkedIn, Facebook і Twitter, є важливими інструментами для залучення найкращих талантів. Завдяки інтеграції соціальних медіа з системами ATS рекрутери можуть легко шукати кандидатів, ділитися вакансіями та переглядати профілі кандидатів. Системи ATS генерують велику кількість даних, які можна використовувати для оптимізації процесу найму. Завдяки інтеграції розширених аналітичних можливостей рекрутери можуть отримати уявлення про свої показники найму та визначити області для вдосконалення.

Підсумовуючи дану дипломну роботу, розробка універсальних ATS яка мала би достатньо широкі можливості та доступну ціну має великий потенціал розвитку і удосконалення.

Література

1. Analysts of the National Estimates Branch. Current Employment Statistics Highlights [Електронний ресурс] / Analysts of the National Estimates Branch – Режим доступу до ресурсу: <https://www.bls.gov/ces/publications/highlights/2022/current-employment-statistics-highlights-05-2022.pdf>.
2. Survey Connects Poor Candidate Experience To Limits In Future Job Applications [Електронний ресурс]. – 2016. – Режим доступу до ресурсу: <https://www.careerarc.com/in-the-news/survey-connects-poor-candidate-experience-to-limits-in-future-job-applications/>.
3. THE TOP 10 CHALLENGES IT TEAMS FACE IN 2023 [Електронний ресурс] – Режим доступу до ресурсу: <https://www.skillsoft.com/blog/the-top-10-challenges-it-teams-face-in-2023>.
4. 5 tech recruiting challenges [Електронний ресурс] – Режим доступу до ресурсу: <https://join.com/recruitment-hr-blog/tech-recruiting-challenges>.
5. Top Seven IT Recruitment Challenges [Електронний ресурс] – Режим доступу до ресурсу: <https://hiring.monster.com/resources/recruiting-strategies/talent-acquisition/challenges-hiring-it-talent/>.
6. See how LinkedIn can help you with your recruitment needs [Електронний ресурс] // LinkedIn – Режим доступу до ресурсу: https://business.linkedin.com/talent-solutions/cx/22/01/uki-recruiter-demo-1?src=go-pa&trk=sem-ga_campid.17500942701_asid.136472491863_crid.643980101250_kw.linkedin%20talent%20solutions_d.c_tid.kwd-362182430527_n.g_mt.e_geo.9044605&mcid=6945090833574424756&cid=&gclid=Cj0KCQjwn9CgBhDjARIsAD15h0AFrN236u9XELz6FQEA0NvBBh6yc1ThqWU6N2PzKML8vIExg6q5M7gaAj0CEALw_wcB&gclsrc=aw.ds.

7. Programming languages used in most popular websites [Електронний ресурс] – Режим доступу до ресурсу: https://en.wikipedia.org/wiki/Programming_languages_used_in_most_popular_websites.
8. How Was LinkedIn Developed? [Електронний ресурс] // GIZELLE LABAY. – 2020. – Режим доступу до ресурсу: <https://wiredelta.com/how-was-linkedin-developed/#:~:text=It%20is%20an%20open%2Dsource,have%20non%2DJava%20based%20clients..>
9. Hire at Scale with Workable [Електронний ресурс] – Режим доступу до ресурсу: <https://www.workable.com/features/automate-and-hire>.
10. Transform Your Remote Hiring End-to-End [Електронний ресурс] – Режим доступу до ресурсу: <https://www.hackerrank.com/solutions/remote-hiring/>.
11. Applicant Tracking Systems Evolve [Електронний ресурс] // Dave Zielinski. – 2011. – Режим доступу до ресурсу: <https://www.shrm.org/resourcesandtools/hr-topics/technology/pages/atsevolves.aspx>.
12. PHP: The Complete Guide for Beginners, Intermediate, and Advanced Detailed Approach to Master PHP Programming by MG Martin 2019 — 238 с.
13. Б. Скотт, Т. Нейл: Проектування веб-інтерфейсів 2016 — 352 с.
14. Speech Recognition for language learners [Електронний ресурс] – Режим доступу до ресурсу: <https://www.speechace.com/#what-we-do>.
15. Free and open-source online code editor [Електронний ресурс] – Режим доступу до ресурсу: <https://judge0.com/>.
16. Enterprise Grade Resume Parser & API [Електронний ресурс] – Режим доступу до ресурсу: <https://www.affinda.com/resume-parser>.