

УДК 004.773.2

Льовкін В.М.¹, Горобець В.І.²

¹ канд. техн. наук, доц. НУ «Запорізька політехніка»

² студ. гр. КНТ-119 НУ «Запорізька політехніка»

ПРОГРАМНИЙ КОМПОНЕНТ З ТЕСТУВАННЯ В СИСТЕМІ ПОШУКУ ВАКАНСІЙ ДЛЯ ПРАЦЕВЛАШТУВАННЯ

Сучасний світ не ставить перед людьми завдання зменшення актуальності пошуку роботи, а навпаки, з надзвичайною швидкістю розвивається та стає все більш вимогливим. Люди звільняються з роботи за власним бажанням, навпаки відбуваються скорочення, дехто бажає змінити своє місце роботи з якихось причин, а інші просто хочуть знайти щось нове та більш цікаве. Замість класичного пошуку роботи сьогодні використовуються спеціалізовані сайти, де можна знайти відповідну вакансію та відгукнутися на неї або ж розмістити резюме.

Однак, з іншого боку, роботодавці теж стикаються з викликами, пов'язаними зі зростаючим числом кандидатів на роботу. Щоб відібрати найбільш придатних претендентів, треба приділити багато часу. Це додатково ускладнюється з розвитком штучного інтелекту, коли пропозиції можуть генеруватися автоматично і таким чином або спотворювати, або в певних ситуаціях робити неможливим проведення відбору серед всіх поданих анкет, резюме. Тому цей трудомісткий процес необхідно вдосконалювати: його можна реалізувати з підтримкою первинного тестування, яке дозволяє відсіяти кандидатів, які не відповідають вимогам вакансії. Це допомагає ефективно використовувати час та зусилля при наймі нових працівників.

Одним із центральних завдань при розробці вебдодатку для пошуку та публікації вакансій для працевлаштування є первинне тестування кандидатів на відповідність бажаній посаді: варто зазначити, що на сьогоднішній день багато аналогічних вебдодатків не надають подібних інструментів для роботодавців.

Відповідний програмний проєкт, який реалізовувався в роботі, було вирішено будувати на основі фреймворку Django [1] тому, що він має вбудовані інструменти такі, як:

– Django ORM, що дозволяє працювати з базами даних в зручній об'єктно-орієнтованій моделі мови Python;

- Django Template, що представляє систему шаблонів, яка дозволяє відокремлювати вигляд вебсторінок від коду додатку;
- Django Admin, що є вбудованим інтерфейсом для адміністрування вебдодатку.

Для реалізації проєкту було використано шаблон проєктування MVC. Цей підхід до розробки програмного забезпечення, дозволяє розділити відповідальність між різними компонентами програми та зробити код більш зрозумілим та легким для підтримки. Для занесення даних про результати тестування кандидатів було використано технологію звернення без презавантаження сторінки – AJAX.

В ході розробки програмного компоненту з тестування було реалізовано наступні функції:

- створення тестів для вакансій роботодавцем;
- виставлення часових обмежень на проходження тесту;
- можливість проходження тестів кандидатами (рисунок 1);
- автоматичний прорахунок балів за спробу;
- збереження результатів тестування у вигляді таблиць (рисунок 2);
- можливість видалення некоректних результатів роботодавцем.

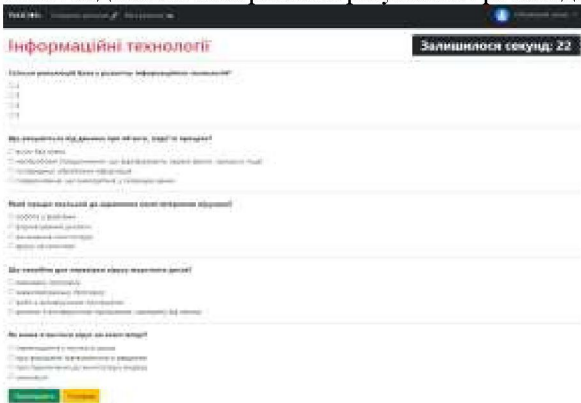


Рисунок 1 – Сторінка тестування кандидата

Користувач	Тест	Вакансія	Оцінка	Додати до мету	Видалити
vladyslav.kovalev	Інформаційні технології	Аналітик	5.0	<button>Додати</button>	<button>Видалити</button>
adrian	Інформаційні технології	Аналітик	4.0	<button>Додати</button>	<button>Видалити</button>
adrian	Інформаційні технології	Аналітик	3.0	<button>Додати</button>	<button>Видалити</button>
adrian	Інформаційні технології	Аналітик	0.0	<button>Додати</button>	<button>Видалити</button>

Рисунок 2 – Сторінка результатів

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. William, S. Vincent Django for Beginners [Текст] : Build websites with Python and Django / S. Vincent William. – WelcomeToCode, 2021. – 296 p.