

# PROJEKT ZESPOŁOWY 02fz-pz

Aplikacja do analizowania dokumentów CV

## Projekt aplikacji do analizy CV z wykorzystaniem Flask

### Spis treści

<b>1. WSTĘP .....</b>	<b>2</b>
<b>2. CEL PROJEKTU.....</b>	<b>2</b>
<b>3. GŁÓWNE FUNKCJONALNOŚCI.....</b>	<b>3</b>
REJESTRACJA .....	3
LOGOWANIE .....	6
PRZESYŁANIE DOKUMENTU CV .....	10
ANALIZA DOKUMENTU CV .....	12
<i>Ekstrakcja treści CV .....</i>	<i>12</i>
<i>Wyszukiwanie kluczowych informacji.....</i>	<i>13</i>
<i>Ocena dopasowania do stanowiska .....</i>	<i>14</i>
<i>Prezentacja wyników .....</i>	<i>15</i>
ZARZĄDZANIE STANOWISKAMI .....	16
<i>Dodawanie stanowisk .....</i>	<i>17</i>
<i>Edytowanie stanowisk .....</i>	<i>19</i>
<i>Usuwanie stanowisk .....</i>	<i>24</i>
RANKING KANDYDATÓW.....	25
<i>Wybór stanowiska pracy i ograniczenie liczby wyników .....</i>	<i>26</i>
<i>Prezentacja wyników .....</i>	<i>27</i>
<i>Dodatkowe akcje dla kandydatów.....</i>	<i>29</i>
<b>3. TECHNOLOGIE I BIBLIOTEKI.....</b>	<b>31</b>
JĘZYKI PROGRAMOWANIA .....	31
POZOSTAŁE JĘZYKI.....	32
FRAMEWORKI I BIBLIOTEKI BACKENDOWE .....	32
FRAMEWORKI I BIBLIOTEKI FRONTENDOWE.....	32
BIBLIOTEKI NLP I ANALIZY DANYCH .....	32
BAZY DANYCH.....	32
INFRASTRUKTURA I NARZĘDZIA WDROŻENIOWE .....	32

ZARZĄDZANIE UŻYTKOWNIKAMI I BEZPIECZEŃSTWO .....	33
FUNKCJE POMOCNICZE .....	33
SYSTEMY KONTROLI WERSJI I ORGANIZACJA PRACY .....	33
<b>4. STRUKTURA KATALOGÓW .....</b>	<b>33</b>
<b>5. MOŻLIWE ROZSZERZENIA .....</b>	<b>34</b>
<b>7. PODSUMOWANIE .....</b>	<b>35</b>

## 1. Wstęp

Prezentowana aplikacja została stworzona, aby wspierać procesy rekrutacyjne poprzez automatyczne przetwarzanie i analizę dokumentów aplikacyjnych. Jej głównym zadaniem jest ułatwienie selekcji kandydatów poprzez identyfikację kluczowych informacji i ocenę ich dopasowania do zdefiniowanych wymagań stanowiskowych. Dzięki wykorzystaniu nowoczesnych technologii, takich jak przetwarzanie języka naturalnego (NLP) i zaawansowana analiza danych, system oferuje intuicyjne zarządzanie procesem rekrutacji. Aplikacja umożliwia precyzyjne przypisywanie parametrów do stanowisk pracy, co pozwala na dokładne porównanie kwalifikacji kandydatów. Zastosowanie tej technologii pozwala firmom zoptymalizować pracę zespołów HR, skracając czas potrzebny na selekcję aplikacji i wspierając proces podejmowania obiektywnych decyzji.

## 2. Cel projektu

Podstawowym celem projektu jest stworzenie narzędzia, które w znaczący sposób automatyzuje procesy rekrutacyjne, eliminując konieczność ręcznej analizy CV. System koncentruje się na wykorzystaniu algorytmów przetwarzania języka naturalnego do wydobywania i oceny danych kluczowych, takich jak kwalifikacje, doświadczenie czy umiejętności kandydatów. Funkcjonalności aplikacji obejmują definiowanie stanowisk pracy wraz z wymaganiami oraz tworzenie mechanizmów, które klasyfikują kandydatów na podstawie stopnia spełnienia tych wymagań. Ważnym aspektem systemu jest jego zdolność do automatycznego generowania rankingów kandydatów, co przyspiesza i usprawnia proces decyzyjny. Realizacja tego celu wspiera organizacje w zmniejszeniu kosztów

rekrutacji, poprawie jakości wyboru kandydatów oraz bardziej efektywnym zarządzaniu zasobami ludzkimi.

### 3. Główne funkcjonalności

#### Rejestracja

Moduł rejestracji polegający na założeniu konta użytkownika, aby miał dostęp do swoich indywidualnych danych kandydatów i stanowisk został zaprojektowany w sposób umożliwiający prosty proces zakładania konta użytkownika, bez dodatkowych uwierzytelnień. Rejestracja wymaga podania podstawowych danych, takich jak:

1. Nazwa użytkownika
2. Adres e-mail
3. Hasło

**Rejestracja**

Nazwa użytkownika:

Sprawozdanie

Adres email:

sprawozdanie@gmail.com

Hasło:

....|

Zarejestruj się

Aby zapewnić bezpieczeństwo danych użytkowników, hasła są szyfrowane za pomocą odpowiedniej funkcji przed zapisaniem ich w bazie danych. System wykorzystuje metodę `set_password` z importu biblioteki `werkzeug.security` w modelu użytkownika, która implementuje szyfrowanie haseł.

```
def set_password(self, password):  
    self.password_hash = generate_password_hash(password)  
  
def check_password(self, password):  
    return check_password_hash(self.password_hash, password)
```

W przypadku błędów, takich jak próba rejestracji z już istniejącą nazwą użytkownika, system wyświetla komunikat za pomocą funkcji flash, a użytkownik jest przekierowywany z powrotem na stronę rejestracji.



The image shows a registration form titled "Rejestracja" (Registration). At the top, there is a red error message box that says "Nazwa użytkownika jest już zajęta!" (Username is already taken!). Below this, there are three input fields: "Nazwa użytkownika:" (Username), "Adres email:" (Email address), and "Hasło:" (Password). At the bottom of the form is a button labeled "Zarejestruj się" (Register).

Po pomyślnym utworzeniu konta, nowy użytkownik jest zapisywany w bazie danych, a sesja kończy się przekierowaniem na stronę logowania z odpowiednim komunikatem potwierdzającym.

## Logowanie

Rejestracja zakończona pomyślnie! Teraz możesz się zalogować.

Nazwa użytkownika:

Hasło:

Zaloguj się

Nie masz konta? [Zarejestruj się](#)

Użytkownik po zarejestrowaniu jest widoczny w bazie danych w tabeli user z zaimplementowanymi danymi na kolumnie id, email, password\_hash (szyfrowanie haseł).

Tabela: user				
Filtruj w dowolnej kolumnie				
	id	username	email	password_hash
	Filtr	Filtr	Filtr	Filtr
1	1	Test1	test@gmail.com	scrypt:...
2	2	Test2	test2@gmail.com	scrypt:...
3	3	Prezentacja	prez@gmail.com	scrypt:...
4	4	TestX	testx@gmail.com	scrypt:...
5	5	Sprawozdanie	sprawozdanie@gmail.com	scrypt:...

## Logowanie

Moduł logowania został zaprojektowany, aby umożliwić użytkownikom dostęp do ich indywidualnych kont w systemie, gdzie mogą zarządzać danymi kandydatów i stanowisk. Logowanie wymaga podania dwóch podstawowych danych uwierzytelniających:

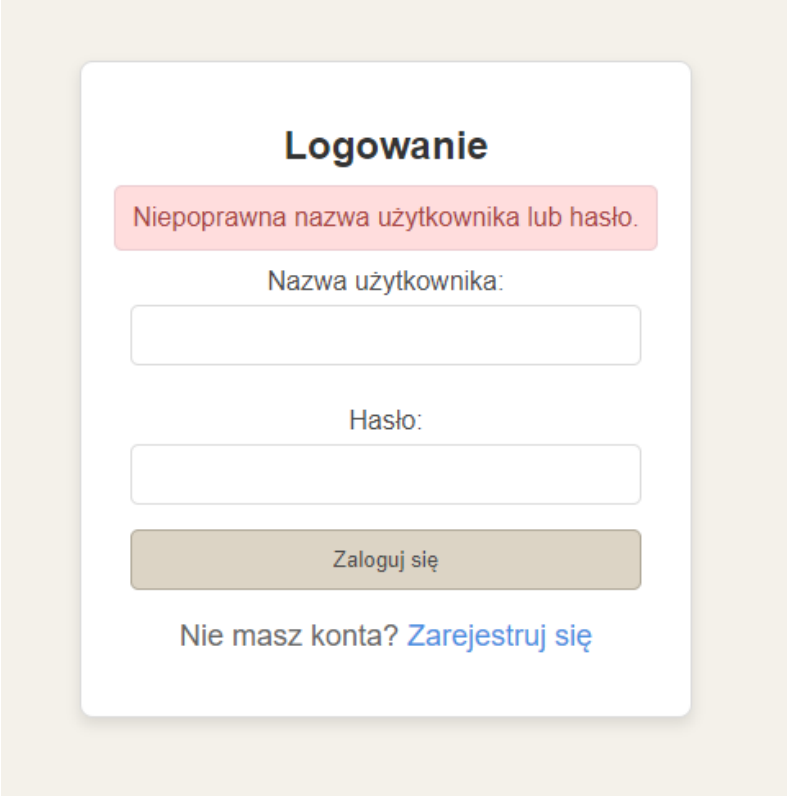
1. Nazwa użytkownika
2. Hasło

The image shows a login form titled "Logowanie". It has a label "Nazwa użytkownika:" followed by a text input field containing the text "Sprawdzanie". Below this is a label "Hasło:" followed by a password input field with masked characters "....". A "Zaloguj się" button is positioned below the password field. At the bottom, there is a link "Nie masz konta? Zarejestruj się".

System weryfikuje podane dane na podstawie zapisanych w bazie informacji. Hasło użytkownika jest sprawdzane za pomocą funkcji `check_password`, która wykorzystuje algorytm porównujący podane hasło z jego zaszyfrowaną wersją przechowywaną w bazie danych.

```
if user and user.check_password(password):
    session["user_id"] = user.id
    flash("Logowanie zakończone sukcesem!")
    return redirect(url_for("home"))
else:
    flash("Niepoprawna nazwa użytkownika lub hasło.")
    return redirect(url_for("login"))
```

W przypadku błędnych danych logowania, takich jak niepoprawna nazwa użytkownika lub hasło, użytkownik otrzymuje odpowiedni komunikat o błędzie za pomocą funkcji flash, a system odświeża stronę logowania. Mechanizm ten zapewnia jasną informację zwrotną bez konieczności przerywania pracy aplikacji.



The image shows a login form titled "Logowanie" (Login) on a light beige background. The form is a white rounded rectangle. At the top, the title "Logowanie" is in bold black font. Below it, a red error message box contains the text "Niepoprawna nazwa użytkownika lub hasło." (Incorrect username or password). Under the error message, there are two input fields: "Nazwa użytkownika:" (Username) and "Hasło:" (Password). Below the password field is a brown button labeled "Zaloguj się" (Login). At the bottom of the form, there is a link that says "Nie masz konta? Zarejestruj się" (Don't have an account? Register).

Po prawidłowym uwierzytelnieniu użytkownik jest logowany do systemu, a jego identyfikator (user\_id) zostaje zapisany w sesji za pomocą mechanizmu Flask Sessions. Umożliwia to autoryzowany dostęp do chronionych zasobów aplikacji oraz personalizację danych. Po zalogowaniu system przekierowuje użytkownika na stronę główną aplikacji.



## Witaj, Sprawozdanie!

Co chciałbyś zrobić dzisiaj?

Prześlij CV

Dodaj nowe stanowisko

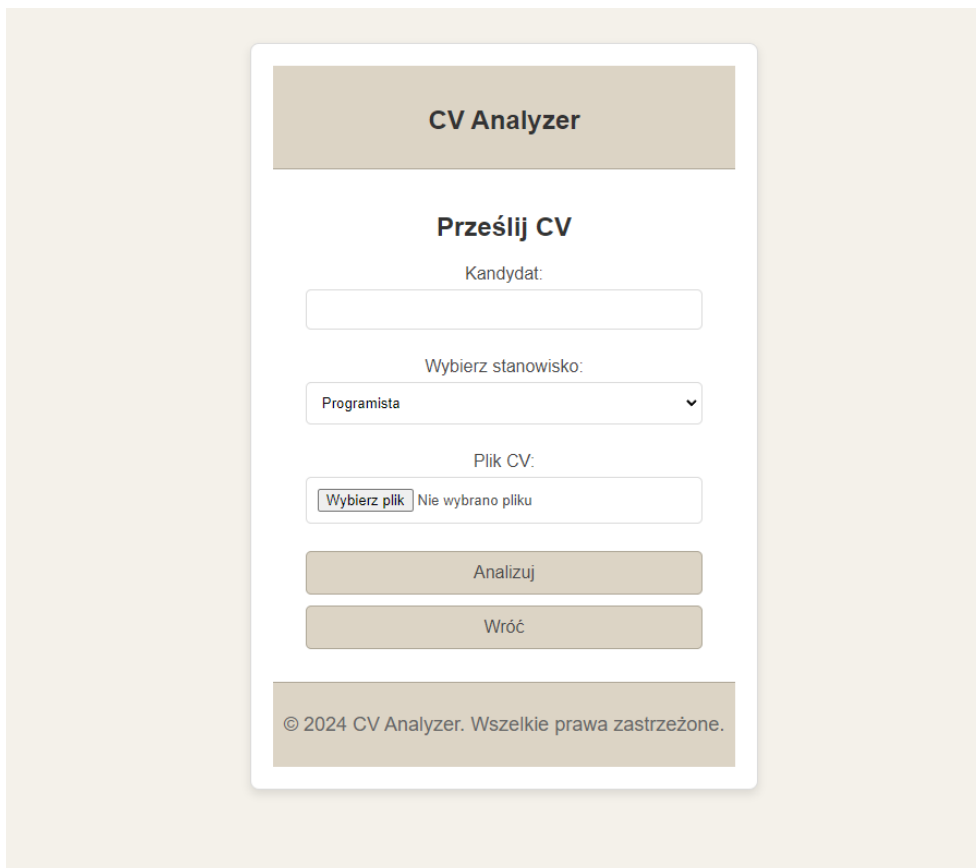
Zobacz swoje stanowiska

Zobacz ranking

Wyloguj się

## Przesyłanie dokumentu CV

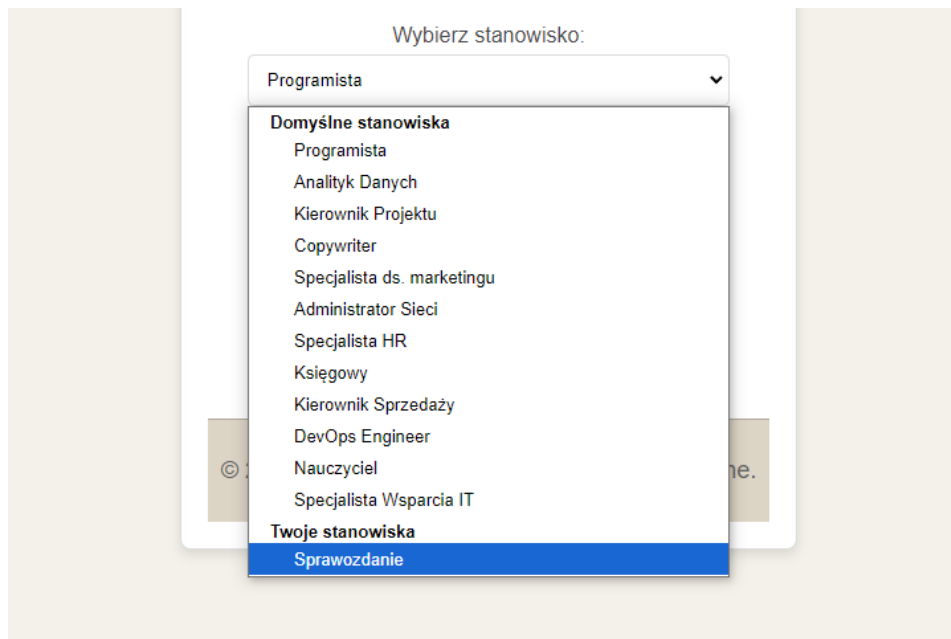
Moduł przesyłania dokumentów CV pozwala użytkownikom na dodanie plików CV kandydatów do systemu w celu przeprowadzenia ich analizy i dopasowania do wybranych stanowisk pracy.



The image shows a web form titled "CV Analyzer" with a sub-header "Prześlij CV". The form contains the following elements: a text input field for "Kandydat:", a dropdown menu for "Wybierz stanowisko:" with "Programista" selected, a file upload area for "Plik CV:" with a "Wybierz plik" button and the text "Nie wybrano pliku", and two buttons at the bottom: "Analizuj" and "Wróć". A footer at the bottom of the form reads "© 2024 CV Analyzer. Wszelkie prawa zastrzeżone."

Formularz przesyłania danych składa się z kilku prostych sekcji:

- a) Pole "Kandydat" umożliwia wpisanie dowolnej nazwy, co pozwala na przypisanie dokumentu do odpowiedniej osoby.
- b) Lista rozwijana "Wybierz stanowisko" pozwala użytkownikowi wybrać stanowisko, dla którego CV będzie analizowane. Lista jest podzielona na kategorie:
  - a. Domyślne stanowiska dostępne globalnie w systemie.
  - b. Stanowiska użytkownika, które zostały wcześniej zdefiniowane przez zalogowanego użytkownika.
- c) Pole „Plik CV” pozwala na przesłanie pliku w formacie .pdf



Po wypełnieniu formularza i wybraniu pliku użytkownik może przestać dane do analizy, klikając przycisk „Analizuj”. Dane zostają wysłane na serwer w celu przetworzenia i oceny dopasowania CV do wybranego stanowiska.

```
<h2>Prześlij CV</h2>
<form action="/analyze_cv" method="post"
  enctype="multipart/form-data">
  <label for="name">Kandydat:</label>
  <input type="text" name="name" required>

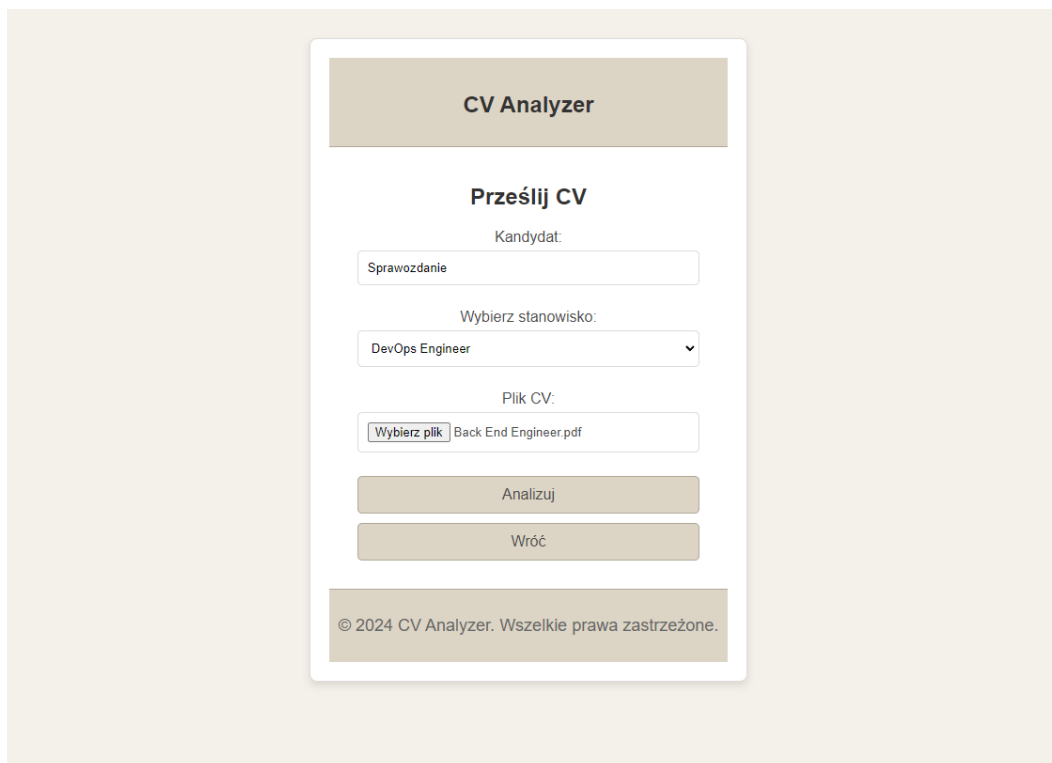
  <label for="position_id">Wybierz stanowisko:</
  label>
  <select name="position_id" required>
    <optgroup label="Domyślne stanowiska">
      {% for position in global_positions %}
      <option value="{{ position.id }}" {% if
        position.id==last_position_id %}selected
        {% endif %}>
        {{ position.title }}
      </option>
      {% endfor %}
    </optgroup>
    <optgroup label="Twoje stanowiska">
      {% for position in user_positions %}
      <option value="{{ position.id }}" {% if
        position.id==last_position_id %}selected
```

W przypadku potrzeby powrotu do poprzedniego ekranu użytkownik może skorzystać z przycisku „Wróć”.

## Analiza dokumentu CV

Proces analizy rozpoczyna się od odebrania danych przesłanych w formularzu:

- Dowolna nazwa kandydata
- Identyfikator stanowiska pracy wybranego z listy
- Plik CV w formacie PDF



The image shows a web form titled "CV Analyzer". The form is titled "Prześlij CV" (Send CV). It contains the following fields and elements:

- A label "Kandydat:" followed by a text input field containing the text "Sprawozdanie".
- A label "Wybierz stanowisko:" followed by a dropdown menu showing "DevOps Engineer".
- A label "Plik CV:" followed by a file upload button labeled "Wybierz plik" and a text input field containing the text "Back End Engineer.pdf".
- Two buttons: "Analizuj" (Analyze) and "Wróć" (Return).
- A footer section containing the text "© 2024 CV Analyzer. Wszelkie prawa zastrzeżone."

Plik CV jest zapisywany w bazie danych pod unikalną nazwą, która łączy imię kandydata z nazwą przesłanego pliku.

## Ekstrakcja treści CV

Dokument CV jest przetwarzany w celu wyodrębnienia zawartości tekstowej. Dla plików PDF, używana jest biblioteka pdf2image do konwersji stron PDF na obrazy, a następnie pytesseract do odczytu tekstu z tych obrazów. W przypadku błędów podczas ekstrakcji system informuje użytkownika o problemie za pomocą komunikatów flash.

```

try:
    pages = convert_from_path(file_path)
    extracted_text = " ".join(image_to_string(page)
    for page in pages)
except Exception as e:
    flash(f"Błąd podczas wyodrębniania tekstu z
    PDF: {str(e)}")
    return redirect(url_for("upload"))

```

## Wyszukiwanie kluczowych informacji

System przeprowadza analizę zawartości tekstowej w celu wyodrębnienia istotnych danych, takich jak:

- Imię i nazwisko kandydata – identyfikowane na podstawie analizy tekstu w pierwszych liniach dokumentu.
- Adres e-mail – wyodrębniany za pomocą wzorca regularnego dopasowującego popularne formaty adresów e-mail.
- Numer telefonu – znajdowany na podstawie wyrażeń regularnych identyfikujących formaty numerów telefonicznych.

```

def extract_email_from_cv_text(text):
    email_pattern = r'[a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}'
    match = re.search(email_pattern, text)
    return match.group(0) if match else None

```

## Ocena dopasowana do stanowiska

Wyszukiwane są wystąpienia słów kluczowych zdefiniowanych dla wybranego stanowiska pracy. Każde słowo kluczowe posiada określoną wagę, która wpływa na końcową punktację. Mechanizm działa następująco:

1. Tekst CV jest normalizowany poprzez usunięcie znaków diakrytycznych oraz zamianę na małe litery.
2. Słowa kluczowe są porównywane z zawartością CV, a każda zgodność zwiększa wynik punktowy zgodnie z wagą przypisaną do danego słowa kluczowego.
3. Wynik dla każdego słowa kluczowego obejmuje liczbę jego wystąpień, wagę oraz łączną liczbę zdobytych punktów.

id	word	position_id	weight
Filtr	Filtr	Filtr	Filtr
1042	C1	18	3
1043	C2	18	4
1061	B1	19	1
1062	B2	19	2
1063	C1	19	3
1064	C2	19	4
1543	English	18	1
1544	Angielski	18	1
1566	English	19	1
1567	Angielski	19	1
2136	programowanie	18	5
2137	Python	18	5
2138	Java	18	5
2139	JavaScript	18	5
2140	C#	18	4
2141	SQL	18	3
2142	bazy danych	18	2
2143	API	18	3

## Prezentacja wyników

Wyniki analizy prezentowane są w nowym oknie z takimi informacjami jak:

- Wprowadzona nazwa kandydata przez użytkownika
- Łączna liczba punktów po analizie dokumentu CV

**CV Analyzer**

**Wyniki analizy dla:**

Wprowadzona nazwa: Sprawozdanie

**Łączna liczba punktów: 38**

Pokaż szczegóły

Wróć

Wyniki analizy po kliknięciu przycisku „Pokaż szczegóły” są prezentowane użytkownikowi w przejrzystej formie tabelarycznej, zawierającej:

- Słowa kluczowe.
- Liczbę ich wystąpień.
- Wagę przypisaną do słowa kluczowego.
- Liczbę punktów zdobytych za każde słowo.

## CV Analyzer

### Wyniki analizy dla:

Wprowadzona nazwa: Sprawozdanie

Łączna liczba punktów: 38

Ukryj szczegóły

Wróć

Słowo kluczowe	Ilość wystąpień	Waga	Punkty
programowanie	0	5	0
Python	5	5	25
Java	0	5	0
JavaScript	0	5	0
C#	0	4	0
SQL	1	3	3
bazy danych	0	2	0
API	0	3	0
framework	0	3	0
debugowanie	0	3	0
kodowanie	0	2	0

Wyniki analizy są zapisywane w bazie danych w tabeli kandydata, która przechowuje m.in.:

1. Pełną treść CV.
  - Wyodrębnione informacje (imię i nazwisko, e-mail, numer telefonu).
  - Całkowity wynik punktowy oraz szczegółowe wyniki dla każdego słowa kluczowego.
  - Ścieżkę do przestanego pliku CV.

Dane są przechowywane w bazie danych, z takiego powodu, że mogą być wykorzystane w aplikacji do dalszego przetwarzania.

## Zarządzanie stanowiskami

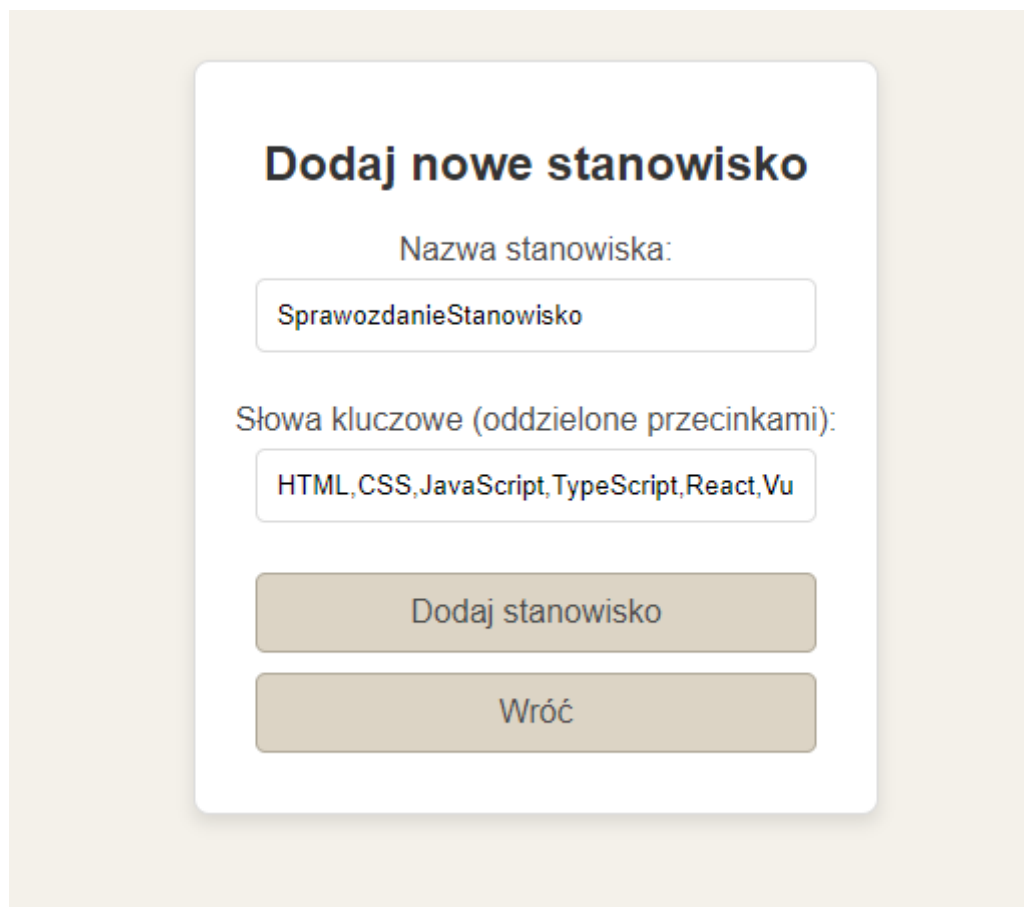
Moduł zarządzania stanowiskami umożliwia użytkownikom dodawanie, edytowanie i usuwanie stanowisk pracy, co stanowi kluczowy element w procesie personalizacji analizy dokumentów CV. Dzięki tej funkcji



użytkownik może definiować wymagania dla poszczególnych stanowisk, co wpływa na skuteczność analizy i dopasowania kandydatów.

## Dodawanie stanowisk

Dodanie nowego stanowiska jest intuicyjne i odbywa się za pośrednictwem dedykowanego formularza. Użytkownik musi podać nazwę stanowiska oraz zdefiniować słowa kluczowe, które mają być uwzględniane podczas analizy CV, oddzielając je od siebie przecinkami.



**Dodaj nowe stanowisko**

Nazwa stanowiska:

SprawozdanieStanowisko

Słowa kluczowe (oddzielone przecinkami):

HTML,CSS,JavaScript,TypeScript,React,Vu

Dodaj stanowisko

Wróć

Funkcja dodawania stanowiska korzysta z formularza HTML oraz mechanizmu obsługi żądań POST w aplikacji, co umożliwia zapisanie danych w tabeli Position w bazie danych. Każde stanowisko może być przypisane do konkretnego użytkownika ze względu kolumnę user\_id w tej tabeli jako klucz obcy, tworząc relację z tabelą User, dzięki czemu dane są dostępne wyłącznie dla jego konta.

Tabela: position

	<u>id</u>	title	user_id	is_default
	Filtr	Filtr	Filtr	Filtr
1	57	SprawozdanieStanowisko	5	0
2	56	Sprawozdanie	5	0

Tabela: user

	<u>id</u>	username	email
	Filtr	Filtr	Filtr
1	5	Sprawozdanie	sprawozdanie@gmail.com

Po dodaniu stanowiska słowa kluczowe zostają zaimplementowane do tabeli Keyword w bazie danych i mają przypisane position\_id tworząc kolejną relację, dzięki czemu słowa kluczowe są przypisane tylko do tego stanowiska.

	word	position_id	weight
1	HTML	57	1
2	CSS	57	1
3	JavaScript	57	1
4	TypeScript	57	1
5	React	57	1
6	Vue	57	1
7	Angular	57	1
8	Sass	57	1
9	Less	57	1
10	Webpack	57	1
11	Babel	57	1
12	Gulp	57	1
13	Node	57	1

Wykonano bez błędów.  
Wynik: Zwrócono 53 wierszy w czasie 15ms  
W wierszu 1:  
SELECT word,position\_id,weight FROM keyword WHERE position\_id=57;

## Edytowanie stanowisk

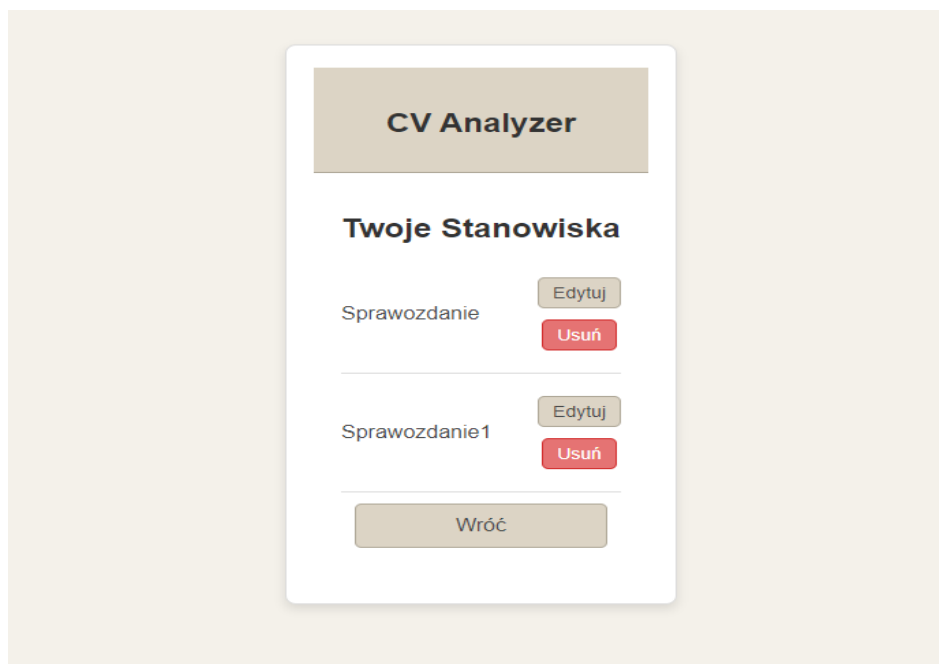
Edytowanie istniejących stanowisk pozwala na aktualizację nazwy oraz powiązanych słów kluczowych, pobranych wcześniej z bazy danych przez konkretny endpoint. Mechanizm edycji stanowiska jest realizowany poprzez dynamiczne załadowanie danych stanowiska do formularza edycyjnego, co pozwala użytkownikowi na wygodne aktualizowanie zawartości.

```
@app.route("/edit_position/<int:position_id>", methods=
["GET", "POST"])
def edit_position(position_id):
    position = Position.query.get_or_404(position_id)

    if request.method == "POST":
        title = request.form.get("title")
        position.title = title

        keyword_ids = request.form.getlist("keyword_ids")
        keyword_words = request.form.getlist
            ("keyword_words")
        weights = request.form.getlist("weights")
        deleted_keywords = request.form.getlist
            ("deleted_keywords")
```

Funkcja ta jest szczególnie przydatna w przypadku zmiany wymagań stanowiskowych lub rozszerzenia ich zakresu. Użytkownik może z łatwością wprowadzać modyfikacje za pomocą interfejsu użytkownika, gdzie każda zmiana jest odzwierciedlana w bazie danych w czasie rzeczywistym.



W oknie edycji stanowiska najważniejszym blokiem jest formularz HTML wspierany przez narzędzie Jinja2 zintegrowane z frameworkiem Flask. Po zatwierdzeniu zmian, dane są zapisywane w bazie za pomocą odpowiedniej funkcji obsługującej żądanie POST.

```
<label>Słowa kluczowe i ich wagi:</label>
<ul class="keywords-list">
  {% for keyword in keywords %}
  <li>
    <input type="text" name="keyword_words"
    value="{{ keyword.word }}" required>
    <input type="hidden" name="keyword_ids"
    value="{{ keyword.id }}">
    <input type="number" name="weights"
    value="{{ keyword.weight }}" min="1"
    max="5" required>
    <button type="button"
    onclick="removeKeyword(this, '{{
    keyword.id }}')">Usuń</button>
  </li>
  {% endfor %}
</ul>
```

Na samej górze formularza wyświetla się etykieta „Tytuł stanowiska” z dotychczasowo pobraną nazwą stanowiska, którą możemy edytować, a pod nią zostaje pobrana lista wszystkich słów kluczowych i wag, dotycząca

danego stanowiska, na której również użytkownik może „manipulować” danymi m.in.:

1. zmienić nazwę
2. zmienić wagę (po utworzeniu stanowiska wszystkie słowa kluczowe mają domyślną wagę 1), zakres wartości wagi to 1-5
3. usunąć słowo kluczowe, przyciskiem Usuń znajdującym się pod danym elementem

**Edytuj stanowisko**

Tytuł stanowiska:

Sprawozdanie1

Słowa kluczowe i ich wagi:

- HTML 1
- CSS 1
- JavaScript 1

Usuń

**Edytuj stanowisko**

Tytuł stanowiska:

Sprawozdanie1

Słowa kluczowe i ich wagi:

- CSS 1
- JavaScript 1
- TypeScript 1

Usuń

W dolnej sekcji formularza, użytkownik ma możliwość dodania słowa kluczowego i przypisanie do niego wagi. Kliknięcie przycisku „Dodaj” spowoduje wywołanie funkcji dodającej słowo kluczowe do formularza.

**Dodaj nowe słowo kluczowe**

Słowo kluczowe:

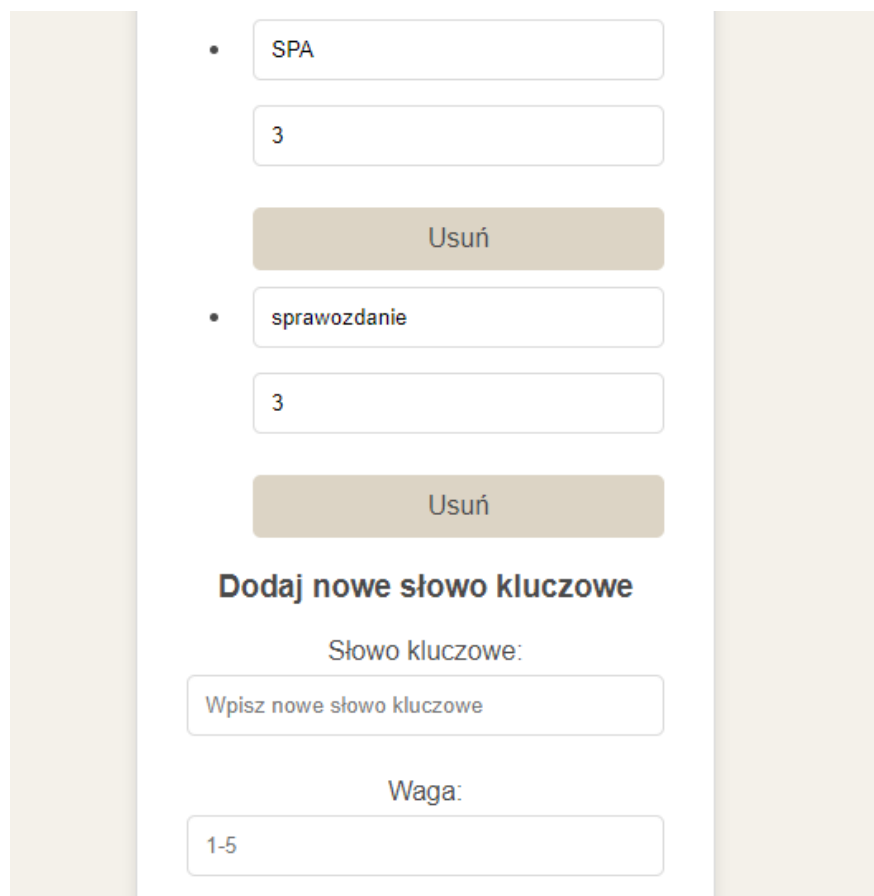
sprawozdanie

Waga:

3

Dodaj

Zapisz zmiany



• SPA

3

Usuń

• sprawozdanie

3

Usuń

**Dodaj nowe słowo kluczowe**

Słowo kluczowe:

Wpisz nowe słowo kluczowe

Waga:

1-5

Za dodawanie i usuwanie słów kluczowych odpowiada skrypt z użyciem języka JavaScript.

Funkcja `removeKeyword(button, keywordId)`, odpowiada za:

1. usuwanie wybranego elementu z listy słów kluczowych `<li>`
2. dodawanie ukrytego pola do formularza, aby oznaczyć które słowa kluczowe zostały usunięte. Dzięki temu backend (Flask) wie, jakie słowa kluczowe usunąć z bazy danych.

```
function removeKeyword(button, keywordId) {
  const li = button.parentElement;
  li.remove();
  const form = document.querySelector('form');
  const input = document.createElement('input');
  input.type = 'hidden';
  input.name = 'deleted_keywords';
  input.value = keywordId;
  form.appendChild(input);
}
```

Funkcja addKeyword(), odpowiada za:

3. dodawanie nowego słowa kluczowego i jego wagi do listy
4. tworzenie nowego elementu <li> w liście słów kluczowych z odpowiednimi polami tekstowymi i liczbowymi.

```
function addKeyword() {
  const keywordInput = document.getElementById(
    'new-keyword');
  const weightInput = document.getElementById(
    'new-weight');

  const keyword = keywordInput.value.trim();
  const weight = weightInput.value.trim();

  if (!keyword || isNaN(weight) || weight < 1 ||
    weight > 5) {
    alert('Podaj poprawne słowo kluczowe i wagę (1-5).');
    return;
  }
}
```

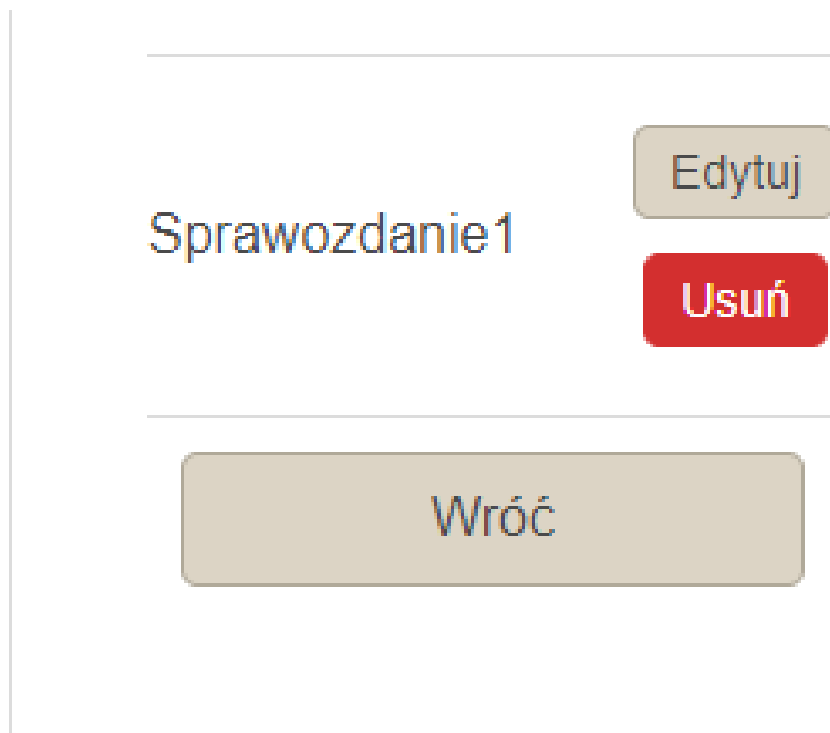
Funkcja `removeKeyword(button)` odpowiada za usuwanie wybranego słowa kluczowego z listy.

```
function removeKeyword(button) {  
    const li = button.parentElement;  
    li.remove();  
}
```

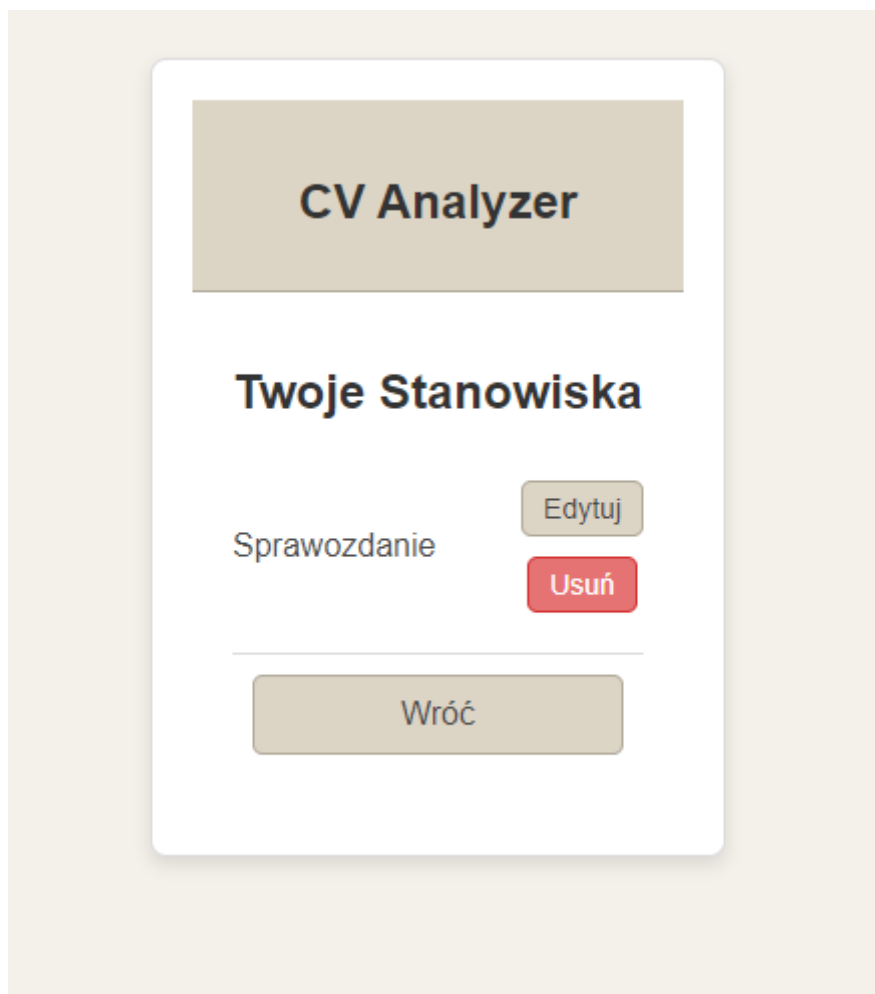
Po zakończeniu edycji stanowiska, użytkownik może zapisać zmiany i wrócić do okna swoich stanowisk.

## Usuwanie stanowisk

Dla stanowisk, które nie są już potrzebne, dostępna jest opcja ich usunięcia. Funkcja ta zapobiega przechowywaniu nieaktualnych danych w systemie, co zwiększa jego przejrzystość i wydajność. Usunięcie stanowiska skutkuje także usunięciem powiązanych z nim słów kluczowych, zapewniając integralność danych.







## Ranking kandydatów

Moduł rankingu kandydatów umożliwia użytkownikowi przeglądanie i analizowanie wyników dopasowania kandydatów do wybranego stanowiska pracy. W przejrzystym układzie użytkownik może z łatwością wybrać stanowisko, określić liczbę wyświetlanych wyników oraz zapoznać się z kluczowymi informacjami na temat każdego kandydata. Dzięki zastosowaniu dynamicznego systemu filtracji i sortowania, użytkownik jako potencjalny rekruter, może łatwo uzyskać listę najlepszych kandydatów na wybrane stanowisko, co znacząco usprawnia proces rekrutacji.

CV Analyzer

Ranking kandydatów

Wybierz stanowisko:

SprawozdanieRanking

Liczba wyników (1-50):

20

Pokaż ranking

Wróć

#	Nazwa	Imię i nazwisko	E-mail	Numer telefonu	Punkty
---	-------	-----------------	--------	----------------	--------

© 2024 CV Analyzer. Wszelkie prawa zastrzeżone.

## Wybór stanowiska pracy i ograniczenie liczby wyników

Formularz umożliwia wybranie jednego z dostępnych stanowisk pracy. Lista stanowisk jest dynamicznie generowana z bazy danych i obejmuje zarówno globalne stanowiska, jak i te dodane przez użytkownika. Wybrane stanowisko określa kryteria oceny kandydatów oraz słowa kluczowe uwzględniane w analizie.

### Ranking kandydatów

Wybierz stanowisko:

SprawozdanieRanking

Testy

Specjalista ds. Sztucznej Inteligencji

Front-end Developer

Programista

Analitik Danych

Kierownik Projektu

Copywriter

Specjalista ds. marketingu

Administrator Sieci

Specjalista HR

Księgowy

Kierownik Sprzedaży

DevOps Engineer

Nauczyciel

Specjalista Wsparcia IT

Sprawozdanie

SprawozdanieRanking

Użytkownik może ustawić maksymalną liczbę wyników wyświetlanych w rankingu (zakres: 1–50). Ta funkcja pozwala na skupienie się na najlepszych kandydatach, co ułatwia podejmowanie decyzji.

Liczba wyników (1-50):

60

⬆️⬆️

!

Wartość nie może być większa niż 50.

## Prezentacja wyników

Ranking jest przedstawiony w formie tabeli, w której wyświetlane są następujące informacje o kandydacie:

1. Pozycja w rankingu (#).
2. Nazwa pliku CV.
3. Imię i nazwisko (lub inne zidentyfikowane dane z dokumentu CV).
4. Adres e-mail.
5. Numer telefonu.
6. Liczba uzyskanych punktów w analizie.

Kandydaci są uporządkowani malejąco według liczby punktów.

CV Analyzer

Ranking kandydatów

Wybierz stanowisko:

SprawozdanieRanking

Liczba wyników (1-50):

20

Pokaż ranking

Wróć

#	Nazwa	Imię i nazwisko	E-mail	Numer telefonu	Punkty		
1	SprawozdanieTestowe	JAN KOWALSKI	sprawozdanie@gmail.com	123456789	30		
2	InnyKandydat1	OLIVIA SCHUMACHER	hello@reallygreatsite.com	123-456-7890	12		
3	InnyKandydat	KAJA DWORCZYK	witaj@naprawdesvietnastrona.pl	123 456 783	10		

© 2024 CV Analyzer. Wszelkie prawa zastrzeżone.

Endpoint pobiera informacje o kandydacie z bazy danych z tabeli Candidate, którzy są tam przechowywani po zakończonej pomyślnie analizie dokumentu CV.







89	89	Ola Dobosz	OLA DOBOSZ...	5	31	OLA DOBOSZ	witaj@naprawdesvietnastrona.pl	NULL	1	NULL
90	90	Kaja Dworczyk	KAJA...	5	7	KAJA DWORCZYK	witaj@naprawdesvietnastrona.pl	123 456 783	1	NULL
91	91	Programista1	KATHY JAMES...	5	66	KATHY JAMES	k.james@email.com	NULL	1	NULL
92	92	Emilia	SMIGIELSKA...	14	8	SMIGIELSKA @ DOSWIACZENIE	emilia.malina904@wp.pl	+48 533 975 853	1	NULL
93	93	TestowyTestowy1	About Me...	9	0	OLIVIA SCHUMACHER	hello@reallygreatsite.com	123-456-7890	1	NULL
94	94	KXKXK	OLA DOBOSZ...	14	23	OLA DOBOSZ	witaj@naprawdesvietnastrona.pl	NULL	1	NULL
95	95	Emilia	SMIGIELSKA...	8	21	SMIGIELSKA @ DOSWIACZENIE	emilia.malina904@wp.pl	+48 533 975 853	1	NULL
96	96	TESTCSS	EDUCATION...	14	5	* ENGLISH *w FRENCH	hello@reallygreatsite.com	123-456-7890	1	NULL
97	97	TESTCSS1	EDUCATION...	14	5	* ENGLISH *w FRENCH	hello@reallygreatsite.com	123-456-7890	1	NULL
98	98	TestySzczegoly	CV...	6	100	CV ADAM KOWALSKI	adamkowalski@email.pl	123 456 789	1	NULL
99	99	TestyCiągle	OLA DOBOSZ...	4	2	OLA DOBOSZ	witaj@naprawdesvietnastrona.pl	NULL	1	NULL
100	100	KandydatTestowy	KAJA...	4	1	KAJA DWORCZYK	witaj@naprawdesvietnastrona.pl	123 456 783	1	NULL
101	101	StopkaStopka	LUKASZ...	4	1	LUKASZ MALINOWSKI	lukasz.malinowski79@gmail.com	762 762 128	1	NULL
102	102	StopkaStopka2	KATHY JAMES...	4	3	KATHY JAMES	k.james@email.com	NULL	1	NULL
103	103	testystopka1	CV...	4	0	CV ADAM KOWALSKI	adamkowalski@email.pl	123 456 789	1	NULL
104	104	NauczycielTest	About Me...	15	0	C 2010-2014) ( GPA:3.85 ) Skill	NULL	123-456-7890	1	NULL
105	105	TestStanowisk	OLIVIA WILSON...	28	3	OLIVIA WILSON	hello@reallygreatsite.com	123-456-7890	1	NULL
106	106	FrontEndKandydat1	FRANCOIS...	44	0	FRANCOIS MERCIER	shell@reallygreatsite.com	123-456-7890	4	NULL
107	117	TestoweCV	Date / Place of birth...	43	119	Nierozpoznane	NULL	NULL	4	Ci...
108	110	TestStanowiska	KATHY JAMES...	47	15	KATHY JAMES	k.james@email.com	NULL	4	Ci...
109	119	TestStanowiska	KATHY JAMES...	47	15	KATHY JAMES	k.james@email.com	NULL	4	Ci...
110	120	TestStanowiska	KATHY JAMES...	47	15	KATHY JAMES	k.james@email.com	NULL	4	Ci...
111	121	Kandydat12	LUKASZ...	43	3	LUKASZ MALINOWSKI	lukasz.malinowski79@gmail.com	762 762 128	4	Ci...
112	122	TestowaSerja	LUKASZ...	48	1	LUKASZ MALINOWSKI	lukasz.malinowski79@gmail.com	762 762 128	4	Ci...
113	123	Sprawozdanie	FRANCOIS...	53	0	FRANCOIS MERCIER	shell@reallygreatsite.com	123-456-7890	5	Ci...
114	124	Sprawozdanie	First Last...	44	38	Nierozpoznane	first.last@gmail.com	+44 1234567890	5	Ci...
115	125	SprawozdanieTestowe	www.reallygreatsite.com...	57	30	JAN KOWALSKI	sprawozdanie@gmail.com	123456789	5	Ci...
116	126	InnyKandydat	KAJA...	57	10	KAJA DWORCZYK	witaj@naprawdesvietnastrona.pl	123 456 783	5	Ci...
117	127	InnyKandydat1	About Me...	57	12	OLIVIA SCHUMACHER	hello@reallygreatsite.com	123-456-7890	5	Ci...

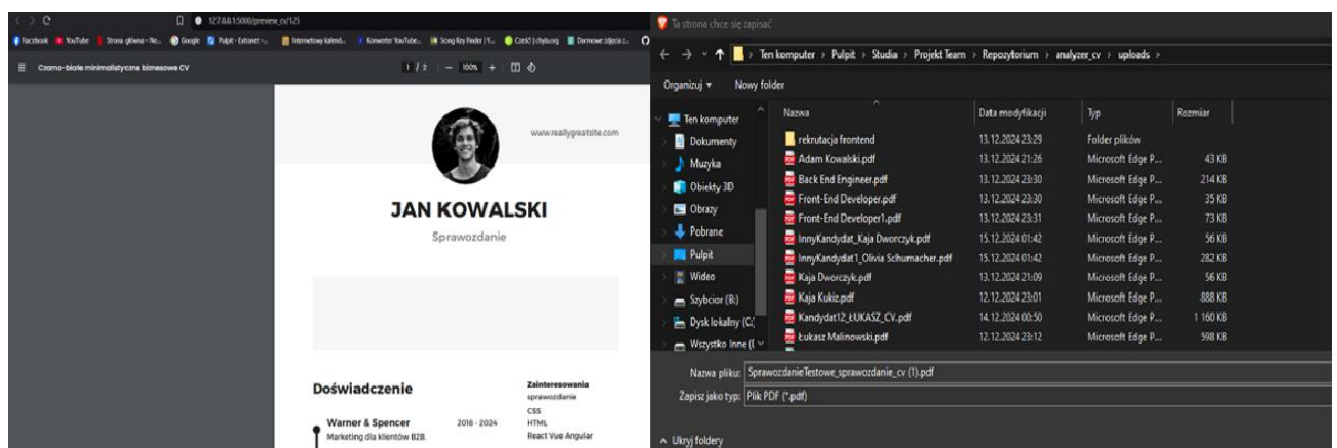
Powyższy zrzut ekranu, przedstawia przykładowe dane testowe, stworzone na rzecz modułu rankingu kandydatów. Każdy zalogowany użytkownik ma swoją bazę kandydatów, z powodu relacyjnego modelu bazy danych, ponieważ w tabeli Candidate znajdują się kolumna user\_id.

## Dodatkowe akcje dla kandydatów

Użytkownik posiada możliwość zarządzania kandydatami z poziomu rankingu:

1. Klikając ikonę „oko” (fa-eye) z biblioteki Font Awesome, użytkownik może otworzyć CV w osobnym oknie przeglądarki
2. Klikając ikonę „pobierz” (fa-download), użytkownik ma możliwość zapisania pliku CV na swoim urządzeniu.
3. Klikając w ikonę „czerwonego kwadratu”, kandydat z zostanie usunięty z listy w rankingu

#	Nazwa	Imię i nazwisko	E-mail	Numer telefonu	Punkty		
1	SprawozdanieTestowe	JAN KOWALSKI	sprawozdanie@gmail.com	123456789	30		
2	InnyKandydat1	OLIVIA SCHUMACHER	hello@reallygreatsite.com	123-456-7890	12		
3	InnyKandydat	KAJA DWORCZYK	witaj@naprawdeswietnastrona.pl	123 456 783	10		



## 4. Baza danych

### Struktura bazy danych

Baza danych aplikacji została zaprojektowana w oparciu o relacyjny model danych, aby efektywnie przechowywać i zarządzać danymi związanymi z procesem analizy CV oraz zarządzaniem użytkownikami i stanowiskami. Schemat bazy danych obejmuje pięć głównych tabel: users, positions, keywords, candidates oraz alembic\_version (służącą do zarządzania migracjami).

### Tabele

#### **Users (Użytkownicy)**

- Przechowuje dane logowania i dane identyfikacyjne użytkowników.
- Pola: id (klucz główny), username, email (unikalne), password\_hash.
- Relacje:
  - Jeden użytkownik może posiadać wiele stanowisk (positions).
  - Jeden użytkownik może mieć przypisanych wiele kandydatów (candidates).

#### **Positions (Stanowiska)**

- Zawiera dane dotyczące stanowisk pracy, na które kandydaci są oceniani.
- Pola: id, title (nazwa stanowiska), is\_default (czy stanowisko jest globalne), user\_id.
- Relacje:
  - Każde stanowisko może mieć przypisane wiele słów kluczowych (keywords).
  - Stanowisko jest powiązane z jednym użytkownikiem, który je utworzył.

#### **Keywords (Słowa Kluczowe)**

- Przechowuje słowa kluczowe przypisane do stanowisk oraz ich wagi.
- Pola: id, word (słowo kluczowe), position\_id (relacja do stanowiska), weight (waga).
- Relacje:
  - Powiązane z tabelą positions.

## Candidates (Kandydaci)

- Przechowuje dane przestanych CV oraz wyniki analizy.
- Pola: id, name, cv\_text, position\_id, points (punkty uzyskane w analizie), first\_words, email\_cv, phone\_number, user\_id, path (ścieżka do przestanego pliku).
- Relacje:
  - Kandydat jest powiązany z określonym stanowiskiem (positions).
  - Kandydat jest przypisany do użytkownika, który przestał jego CV.

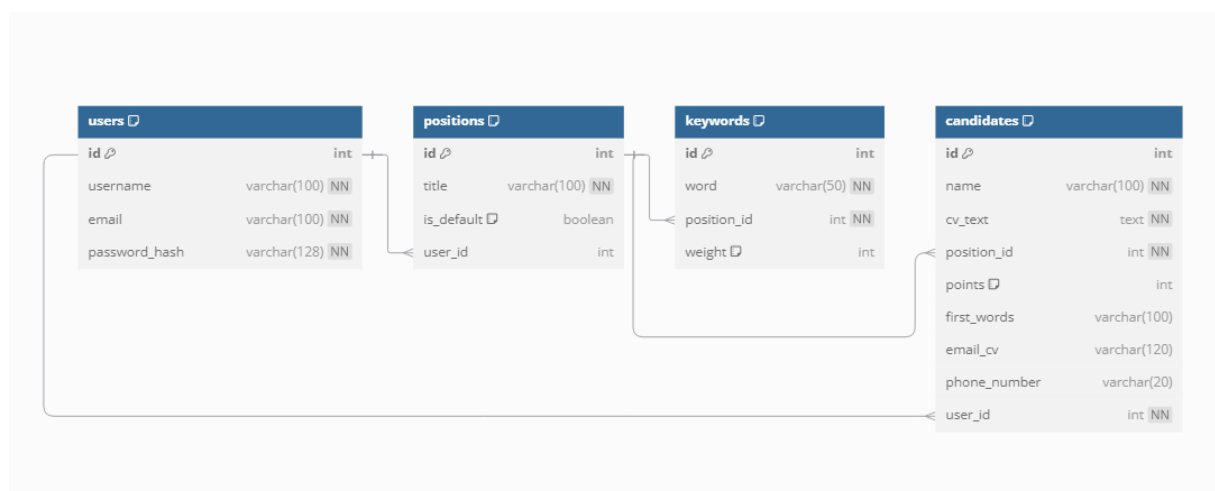
## Alembic\_version

- Przechowuje dane dotyczące wersji migracji bazy danych, co pozwala na zarządzanie zmianami w schemacie bazy.

## Diagram ERD

Struktura bazy danych została przedstawiona w formie diagramu ERD (ang. Entity-Relationship Diagram). Diagram ukazuje relacje między tabelami:

- users ma relację jeden-do-wielu z positions oraz candidates.
- positions ma relację jeden-do-wielu z keywords oraz z candidates.



## 5. Technologie i biblioteki

### Języki programowania

1. Python – główny język backendu.

2. JavaScript – do dynamicznych interakcji (funkcje w `edit_position.html`)

## Pozostałe języki

1. HTML – do tworzenia szablonów stron
2. CSS – do stylizacji i projektowania interfejsu użytkownika

## Frameworki i biblioteki backendowe

- a) Flask - framework webowy do obsługi backendu i integracji z frontendem.
- b) Flask-SQLAlchemy - ORM do zarządzania bazą danych.
- c) Flask-Migrate - narzędzie do migracji schematów bazy danych.
- d) Jinja2 - silnik szablonów w Flask, używany do dynamicznego generowania HTML w szablonie i szkielecie interfejsu aplikacji tworzonemu na języku znaczników
- e) gunicorn - serwer WSGI do produkcyjnego wdrożenia aplikacji.

## Frameworki i biblioteki frontendowe

- f) Bootstrap - stylizacja i responsywność interfejsu.
- g) Font Awesome - ikony używane w elementach interfejsu (np. przyciski).

## Biblioteki NLP i analizy danych

- a) Pytesseract – narzędzie OCR do ekstrakcji tekstu z obrazów PDF
- b) Pdf2image – konwersja plików PDF na obraz\

## Bazy danych

- a) SQLite – lokalna baza danych używana w środowisku deweloperskim.

## Infrastruktura i narzędzia wdrożeniowe

- a) Docker - konteneryzacja aplikacji:
  - a. Obraz oparty na `python:3.8-slim`.
  - b. `requirements.txt` do instalacji zależności.
- b) Heroku - platforma chmurowa do hostowania aplikacji:



- a. Plik heroku.yml do konfiguracji wdrożenia.
- b. Plik Procfile.txt do uruchamiania aplikacji z użyciem Gunicorn.
- c) Alembic - narzędzie do migracji bazy danych (konfiguracja w alembic.ini).

## Zarządzanie użytkownikami i bezpieczeństwo

- a) Werkzeug.security – generowanie i weryfikacja hashów haseł użytkowników

## Funkcje pomocnicze

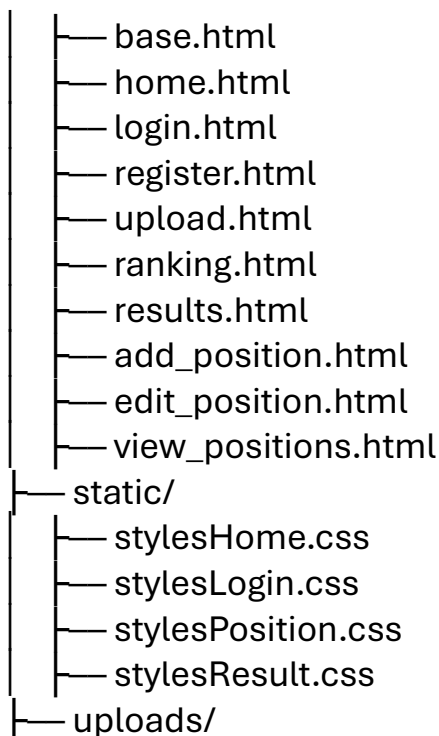
- b) re - do wyszukiwania wzorców w tekście (np. e-maile, numery telefonów).
- c) unicodedata - normalizacja tekstu w celu usunięcia znaków diakrytycznych.
- d) os - operacje na plikach i ścieżkach.
- e) jsonify - zwracanie danych w formacie JSON w odpowiedzi HTTP.

## Systemy kontroli wersji i organizacja pracy

- a) GitHub - repozytorium kodu.
- b) .gitignore - konfiguracja ignorowanych plików (np. środowiska wirtualne, bazy danych, klucze .env).

# 6. Struktura katalogów

```
analyzer_cv/  
├── app.py  
├── models.py  
└── templates/
```



## 7. Możliwe rozszerzenia

### 1. Frontend:

- a. Interaktywne dashboardy i nowoczesny design (np. React/Vue.js).
- b. Tryb ciemny/jasny.
- c. Zaawansowane filtry wyszukiwania kandydatów.

#### a) Analiza CV

- a. Obsługa dodatkowych formatów (np. .doc, .docx).
- b. Lepsza analiza semantyczna z wykorzystaniem zaawansowanych modeli NLP.
- c. Punktowanie doświadczenia i kwalifikacji na podstawie dat i certyfikatów.

#### a) Ranking kandydatów:

- a. Zaawansowane kryteria rankingowe z wizualizacjami wyników (np. wykresy).
- b. Automatyczne rekomendacje stanowisk na podstawie analizy treści CV.
- c. Eksport raportów rankingowych do PDF lub Excel.

- a) Funkcjonalności użytkownika:
- a. Różne role użytkowników (np. administrator, rekruter, kandydat).
  - b. Historia przesłanych CV i analiz.
  - c. Powiadomienia e-mail o wynikach analizy.

## 7. Podsumowanie

Stworzona aplikacja do analizy CV stanowi rozwiązanie dla współczesnych wyzwań związanych z procesami rekrutacyjnymi. Dzięki zastosowaniu zaawansowanych technologii, takich jak przetwarzanie języka naturalnego (NLP) oraz odpowiednio zaprojektowane algorytmy dopasowujące, narzędzie to automatyzuje najbardziej czasochłonne aspekty rekrutacji. Główne funkcje, takie jak ekstrakcja kluczowych informacji z CV, dopasowanie kandydatów do zdefiniowanych stanowisk oraz generowanie rankingów, sprawiają, że aplikacja wspiera zarówno rekruterów, jak i firmy, w szybkim i efektywnym pozyskiwaniu kandydatów.