

Spis treści

1. Opis świata rzeczywistego	2
1.1 Charakterystyka problemu	2
1.2 Charakterystyka problematyki	3
1.3 Analiza rynku.....	5
1.3.1 Analiza istniejących rozwiązań	5
1.3.2 Analiza potrzeb – podsumowanie	8
2. Koncepcja systemu	9
2.1 Koncepcja rozwiązania problemu.....	9
2.2 Diagram przypadków użycia.....	11
2.2.1 Scenariusze dla poszczególnych przypadków użycia w systemie	12
Przypadek użycia: usuń kandydata.....	12
Przypadek użycia: dodaj kandydata	12
Przypadek użycia: wyszukaj kandydata	12
Przypadek użycia: edytuj kandydata	12
Przypadek użycia: powrót do strony głównej.....	13
Przypadek użycia: wyświetl szczegóły kandydata	13
Przypadek użycia: dodaj notatki z rozmów rekrutacyjnych.....	13
Przypadek użycia: wgrywa pliki (import danych).....	13
Przypadek użycia: wykonuje export danych	13
Przypadek użycia: loguje się	14
Przypadek użycia: zarządza bazą danych.....	14
Przypadek użycia: zakłada konto Rekrutera.....	14
Przypadek użycia: usuwa konto Rekrutera.....	14
Przypadek użycia: zarządza systemem.....	15
2.3 Diagram aktywności dla przypadku użycia „wyszukaj kandydata”.	15
2.4 Diagram sekwencji dla wybranych procesów	16
2.5 Wymagania funkcjonalne	17
2.6 Wymagania нефункционалне	18
3. Środowisko programistyczne.....	19
3.1 Systemy operacyjne	19
3.2 Środowisko Programistyczne	19
3.3 Języki programowania i biblioteki i inne narzędzia.....	20
4. Prezentacja warstwy użytkowej projektu	20
5. Opis techniczny projektu	27
6. Testy jednostkowe i użytkownika	27

6.1	Testy użytkownika	27
6.2	Testy jednostkowe komponentów VUE	32
6.2.1	Przykładowe testy jednostkowe.....	32
6.2.2	Wyniki testów jednostkowych	34
7.	Repozytorium, system kontroli wersji	34
8.	Podsumowanie i wnioski	34
9.	Literatura i źródła:.....	35

1. Opis świata rzeczywistego

1.1 Charakterystyka problemu

Rekruter to specjalista do spraw rekrutacji, przedstawiciel branży HR, czyli Human Resources. Do najważniejszych zadań rekrutera należy zaspokajanie potrzeb organizacji w zakresie zasobów ludzkich, poprzez rekrutację i selekcję najlepszych kandydatów do zatrudnienia na określone stanowiska¹. Prowadzi on rekrutację poprzez przygotowywanie ofert, albo z wykorzystaniem banków danych lub też przez rekrutację bezpośrednią. W tej pracy nie będziemy się skupiać na wykorzystaniu banków danych, ale skupimy się na pozostałych aspektach wymienionych wyżej.

Przeprowadzono rozmowy z kilkoma pracownikami działów HR w rzeszowskich firmach IT na temat organizacji specyfiki ich pracy i obowiązków jakie w zakresie tej pracy wykonują. Pracownicy działów HR (rekruterzy) opowiedzieli o swoich problemach organizacji pracy w takim dziale. Te problemy dotyczą większości firm, jednak specyfika badanego problemu dotyczy głównie branży IT.

Pracownicy na stanowisku rekrutera niejednokrotnie zmagają się z problemami natłoku napływających dokumentów CV i listów motywacyjnych (zarówno w formie papierowej jak i elektronicznej) oraz e-maili. Wszystkie te dokumenty są skrupulatnie, sortowane, przechowywane i chronione zgodnie z obowiązującymi aktualnie przepisami RODO. Część rekruterów uczestniczy często w ogólnopolskich targach pracy², organizowanych co pół roku na terenie całej Polski, na których są przedstawicielami swoich firm, a czasami również prezydentami na tego typu wydarzeniach. Potencjalni kandydaci (programiści, informatycy, testerzy) mają możliwość bezpośredniej interakcji z pracodawcami i przedsiębiorcami. Przy zapisie na takie targi każdy kandydat uzupełnia ankietę, która zawiera szczegółowe pytania odnośnie ich umiejętności, wykształceniu i doświadczeniu zawodowym. Po zakończonych targach każdy z rekruterów uczestniczący w targach, dostaje od organizatorów targów pracy pliki Excel z danymi o potencjalnych kandydatach na stanowisko pracy. W tych plikach znajdują się informacje których udzielili w ankietach potencjalni kandydaci do pracy (uczestnicy targów). W jednym takim pliku znajdują się informacje o około 500 różnych osobach. Na tym jednak nie koniec, ponieważ każdy z pracowników działu HR w obecnych czasach buduje sieć kontaktów na portalach takich jak LinkedIn i GoldenLine, gdzie osobiście kontaktuje się z kandydatami wysyłając im oferty pracy. Takie zachowanie może sugerować, jak duże jest zapotrzebowanie na pracowników IT, skoro rekruterzy podejmują wysiłki składania ofert personalnych z wieloma kandydatami naraz poprzez portale biznesowe. Kandydaci coraz chętniej odpowiadają na oferty pracy kierowane personalnie, zatem jest to skuteczny sposób pozyskiwania pracowników. Jednak nadal pojawia się problem, gdzie te dane o tych kandydatach gromadzić.

Nietrudno sobie wyobrazić jak dużo może być tych zebranych danych po 2 latach pracy na stanowisku rekrutera. Aby praca rekrutera była skuteczna, konieczne jest opracowanie systemu, który scentralizuje i ustandaryzuje zgromadzone dane. Jest to ważne choćby z tego powodu, że zapotrzebowanie na dobrych programistów jest bardzo duże a na rynku pracy branży IT stale brakuje kandydatów, dlatego każde CV i każda informacja jest bardzo istotna. Celem każdego rekrutera jest szybko i efektywnie znaleźć odpowiedniego kandydata na dane stanowisko pracy (również spośród kandydatów, którzy złożyli CV np. pół roku wcześniej), zatem dobra organizacja pracy i systemy do zarządzania danymi kandydatów są niezbędne w pracy rekrutera.

Obecnie jest na rynku kilka systemów informatycznych wspomagających pracę pracowników HR takich jak KARO HRMS, enova365, Sage Hrcloud, Macrologic Merit, HCM Deck, tomHRM, Softlab HR by Asseco, HRsys, elevato³. Wiele z tych systemów jest bardzo drogie, dlatego też

¹ <https://www.randstad.pl/strefa-pracownika/rozwoj-zawodowy/czym-zajmuje-sie-rekruter-i-jak-mozna-nim-zostac/> dostęp z dnia 15 stycznia 2022.

² <https://targikarierait.pl/targi-marzec-2022> dostęp z dnia 5 lutego 2022

³ <http://www.jakisystemhr.pl/> Dostępne systemy - dostęp z dnia 5 lutego 2022.

niewiele firm decyduje się na ich zakup, dodatkowo nie każdy z systemów zaspokaja wszystkie potrzeby pracownika HR. Biorąc pod uwagę fakt, że żaden z systemów nie jest idealny, a preferencje rekruterów w różnych branżach się różnią, to nadal jest zapotrzebowanie na coraz to nowsze i lepiej dostosowane systemy do specyfiki danej branży.

Celem tej pracy jest stworzenie takiego systemu, który spełni oczekiwania jednego z rzeszowskich pracowników na stanowisku Human Resources, usprawni wyszukiwanie potencjalnych kandydatów spośród danych, które posiada już pracownik HR, ułatwi gromadzenie takich danych w jednym miejscu i zarządzanie nimi, tym samym skróci znacząco jego czas pracy i poprawi efektywność wyszukiwania najbardziej dopasowanych kandydatów do danego stanowiska pracy. System będzie dedykowany dla pracownika działu HR branży IT. Specyfika filtrowania danych w systemie będzie zatem bardziej dopasowana do kandydatów takich jak programiści, testerzy, web designerzy, informatycy. Obecnie na rynku nie ma systemu, który jest tak bardzo dopasowany do preferencji pracowników działów HR branży IT⁴, dlatego rozwiązanie tego problemu jest bardzo istotne z punktu widzenia rekruterów tej branży. Aby jeszcze bardziej szczegółowo przedstawić dostępne systemy oferowane pracownikom HR branży IT i porównać je, przeprowadzono analizę rynku.

1.2 Charakterystyka problematyki

Jeden z rekruterów spośród trzech z którymi przeprowadzono wywiad, posiada już w tej chwili dane o około 2000 różnych kandydatach. Część danych posiada w postaci elektronicznej, część w formie papierowej. Wyszukiwanie odpowiedniego kandydata na dane stanowisko z pośród tak dużej ilości danych w różnej formie jest dla niego bardzo problematyczne. Dodatkowym problemem jest to, że nie wiemy tak naprawdę jakich kandydatów z danymi kwalifikacjami na dane stanowiska posiada, zaś wyszukiwanie tych informacji jest czasochłonne.

W dzisiejszych czasach dąży się do tego, aby przechowywać wszystko w jednym miejscu, najlepiej w wersji elektronicznej, dąży się też do kategoryzacji danych. W ten sposób można usprawnić pracę pracowników HR, dostając jednocześnie bardziej dopasowanych kandydatów na dane stanowiska a dodatkowo skrócić czas wyszukiwania takich potencjalnych pracowników.

Naszym celem jest przede wszystkim skrócenie czasu wyszukiwania takich pracowników na dane stanowisko, ponieważ zapotrzebowanie na pracowników branży IT jest w tej chwili tak duże, że ciężko znaleźć jest dobrego pracownika. Pracownicy HR są premiovani przez swoich zleceniodawców za szybkość i efektywność podczas niejednokrotnie skomplikowanego procesu rekrutacji. Biorąc pod uwagę sytuację, gdzie dany kandydat złoży CV do kilku firm na raz, to od momentu przeczytania tego CV przez naszego rekrutera do momentu skontaktowania się z kandydatem, ten sam kandydat może przyjąć już ofertę u innego pracodawcy. Jest to pewne ryzyko, które również wiąże się z czasem i sprawną organizacją pracy rekrutera.

Wyobraźmy sobie sytuację: Firma potrzebuje pracownika na już, zaś wszyscy którzy złożyli CV do firmy odmówili z powodu znalezienia innej pracy. Co robić? Mając dużą bazę kandydatów w jakimś systemie informatycznym, rekruter ma możliwość skontaktowania się w takiej sytuacji z inną osobą o podobnych kwalifikacjach.

Kolejnym problemem z jakim zmagają się rekruterzy jest fakt, że niejednokrotnie zdarza się, że ta sama osoba, która złożyła wcześniej CV do firmy „X” i nie została zakwalifikowana do kolejnego etapu rekrutacji, składa je po roku czasu ponownie do tej samej firmy. Ludzie w branży IT często zmieniają pracę, głównie z powodów finansowych, ale także pięć się po coraz to wyższych szczeblach kariery. Jeśli taki kandydat złoży CV po roku to jego kwalifikacje są już nieco większe niż przed rokiem z uwagi na to, że nabył już doświadczenie w firmie „Y” lub zdobył dodatkowe kwalifikacje i certyfikaty poświadczające wzrost nabytych umiejętności. Zatem CV które składał rok temu może być już nieaktualne. Jak sprawdzić czy ktoś już takie CV składał? Bez dobrego systemu

⁴ Informacja zaczerpnięta po przeprowadzeniu rozmów z pracownikami działów HR

wyszukiwania, ciężko jest znaleźć stare CV, nawet gdy jest ono w postaci elektronicznej, a pamiętajmy, że część CV nadal jest składane w formie papierowej co utrudnia wyszukiwanie. W dziale HR nie ma na to czasu, ale weryfikacja jest konieczna z uwagi na fakt, że część firm stosuje zasadę: „jeśli kandydat złożył CV do naszej firmy „X” do poł roku wcześniej to aplikacja jest automatycznie odrzucana”. Zasada ta pozwala uniknąć rozpatrywania tych samych CV, które zostały wcześniej odrzucone i pozwala zaoszczędzić czas pracy pracownikowi HR. Zatem potrzebny jest system aktualizacji danych takich pracowników i aktualizacji ich CV.

Użytkownik (rekruter) często zbiera dane o kandydatach za pomocą portalu LinkedIn i GoldenLine. Są to portale biznesowe podobne do Facebooka. Każdy z użytkowników tworzy tam swój własny profil biznesowy, który zawiera informacje o tym czym się dana osoba zajmuje, o umiejętnościach, jakie ma wykształcenie, gdzie obecnie pracuje. Przez te portale istnieje też możliwość również składania aplikacji kandydatów w odpowiedzi na oferty rekruterów. Jako rekruter, użytkownik systemu potrzebuje, by prowadzone przez niego rozmowy na czatach przez te portale dawały mu możliwość zbierania informacji o treści ostatnich rozmów na tym portalu i podstawowych danych o kandydacie, a następnie eksportowania danych do jakiegoś pliku, który będzie można zaimportować później do aplikacji (systemu informatycznego HR). Portal LinkedIn nie umożliwia w tej chwili takiej opcji. API LinkedIn posiada rozszerzone możliwości tylko dla użytkowników „premium dla rekruterów”, jednak i tam nie ma możliwości takich jak życzył by sobie użytkownik, dla którego dedykowany jest system „Pomocnik Rekrutera”. Dlatego tutaj potrzebna jest integracja naszej aplikacji z portalem LinkedIn biorąc pod uwagę rozwiązania inne niż API LinkedIn np. specjalistyczne wtyczki do przeglądarki zintegrowane z naszym systemem.

Obecnie nie istnieje na rynku system, który ułatwił by w ten sposób pracę pracownika działu HR. Po przejrzeniu wielu portali i projektów na GitHub oraz rozmowach z ludźmi w branży, dowiedzieliśmy się, że jest jeden programista, który tworzy taki system jednak nie jest on jeszcze dostępny dla wszystkich (obecnie jest to projekt prywatny), a nawet jeśli niedługo będzie dostęp, to z pewnością taki system nie będzie tani.

Automatyzacja wszystkich tych wyżej opisanych procesów i rozwiązanie wszystkich problemów jest celem niniejszej pracy.

Jak wspomniałam powyżej pliki z danymi o kandydatach wpływają na bieżąco w różnych formatach. CV wpływają zazwyczaj w formacie .pdf, zaś dane z targów pracy w plikach .xlsx. Potrzebna jest zatem jakaś standaryzacja danych oraz sposób przetworzenia tych danych, aby pokategoryzować pracowników. Zatem program będzie musiał umieć przetworzyć pliki z formatu .xlsx lub alternatywnie .csv i zapisać je w bazie danych.

Pliki CV w postaci elektronicznej będą przechowywane jako oryginały bezpośrednio na serwerze, ponieważ będziemy chcieli stare CV zastępować nowym lub przechowywać obie wersje. Założono, że CV papierowe będą skanowane do wersji elektronicznej. Zajdzie tu konieczność dodania funkcjonalności wgrywania takich plików na serwer. Trzeba będzie to odpowiednio obsłużyć po stronie serwera.

Skoro zależy nam na czasie i chcemy bardzo szybko zatrudnić pracownika na dane stanowisko to potrzebujemy bardzo dobry system wyszukiwania kandydatów z pośród już posiadanych danych. Da nam to możliwość wyszukiwania nawet kilku alternatywnych pracowników na raz, oszczędzając przy tym na czasie i dając większy wybór.

Centralizacja - zależy nam również na likwidacji przechowywania danych w różnych miejscach na komputerze i zlikwidowaniu przetrzymywania papierowych CV. Przechowywane dane powinny znajdować się na serwerze tak aby za pośrednictwem Internetu przez przeglądarkę można było dostać się do tych danych z dowolnego miejsca.

Chcąc dobrze się przygotować na rozmowę kwalifikacyjną z danym kandydatem potrzebujemy również możliwości dostępu do aplikacji z poziomu przeglądarki internetowej (w celu odnalezienia informacji o pracowniku) oraz możliwości edycji informacji o takim kandydacie. Zapisanie ewentualnych drobnych notatek z rozmów kwalifikacyjnych, edycje numerów telefonu, edycje umiejętności lub po prostu możliwości dodania nowego kandydata do naszej bazy w trakcie takiej rozmowy.

Biorąc pod uwagę, że użytkownik może nie chcąc dodać jakiegoś kandydata do bazy a nie było to jego celem, to musi być też możliwość usuwania takiego kandydata.

Integracja z LinkedIn – jest to jeden z najtrudniejszych problemów do rozwiązania, założono, że w ramach aplikacji będzie możliwość doinstalowania wtyczki do przeglądarki, która pobierze pożądane dane z rozmów na chatach przez portal LinkedIn w postaci .csv lub .json. Tutaj też będzie konieczność odpowiedniego obsłużenia tych danych. Niestety użytkownik systemu (rekruter) nie chce płacić za API, które nie posiada wszystkich funkcjonalności których on potrzebuje, dlatego zrezygnowano z wykorzystania API LinkedIn.

1.3 Analiza rynku

1.3.1 Analiza istniejących rozwiązań

Tabela 1 Zestawienie dostępnych aplikacji i narzędzi dla pracowników działów HR w branży IT

Producent	Nazwa narzędzia	Posiadane wyróżniki/braki	Istotne cechy i funkcje
KONKURENCJA BEZPOŚREDNIA			
ENNOVA	tomHRM ⁵	<ul style="list-style-type: none"> + przyjazny Layout dla użytkownika + możliwość udostępniania danych do menadżerów, + możliwość oznaczania etapów rekrutacji, przenoszenie między rekrutacjami + planowanie spotkań i rozmów + możliwość oceny kandydata + integracja z Google i Outlook, Office365 + obsługa klauzuli RODO (notyfikacje) + baza szablonów e-mail i możliwość wysyłania spersonalizowanych e-maili + monitorowania źródeł pozyskania kandydatów + posiada kilka wersji językowych i motywy na podstawie wybranego przez użytkownika koloru - brak informacji o możliwości wgrywania plików z danymi w formatach .csv lub .xlsx o określonej strukturze danych 	<ul style="list-style-type: none"> + posiada funkcje przechowywania bazy kandydatów i filtrowania w własnym Talent pools, + dodawanie i usuwanie kandydatów, dodawanie notatek do kandydata + częściowa integracja z LinkedIn (umożliwienie udostępniania ofert pracy jednym kliknięciem na portalach społecznościowych) + możliwość tworzenia własnego basenu kandydatów i zarządzania nimi

⁵ WWW <https://tomhrm.com/system-ats/#1564087623891-9bb9b31a-b65b> dostęp z dnia 4.03.2022

		- cena 12 938,04 PLN / 12 miesięcy lub 1299,00 PLN / 1 m-c przy limicie 30rekrutacji/m-c	+ automatyczne odczytywanie danych kandydata z pliku CV.pdf + posiada możliwość połączenia z własną stroną firmową lub dowolnym systemem poprzez API
Elevato S.A.	elevato ⁶	+ przyjazny Layout dla użytkownika + Sourcing kandydatów – wyszukiwanie kandydatów na LinkedIn, Stack Overflow i GitHub + Integracja z OLX, Indeed, Adzuna, Jooble, Fachpraca + konfiguracja własnych etapów rekrutacji + szybki podgląd CV +statystyki dotyczące każdego projektu rekrutacji + zarządzanie klauzulami RODO (notyfikacje) - automatyczne usuwanie kandydatów po zamknięciu rekrutacji - cena: 1. wersja LITE 99 zł / mc – limit 1 rekruter i max 1 projekt rekrutacyjny 2. wersja PRO od 469 zł / mc – limit liczby rekruterów dopasowany do klienta, bez limitów projektów	+ jedna wspólna baza CV z kandydatami z wielu źródeł + możliwość oceniania kandydatów, dodawania notatek, tagowania + dodawanie i usuwanie kandydatów, ofert pracy +export danych do PDF i Excel
KONKURENCJA POŚREDNIA			
Tradiss	enova365	+ system modułowy do zarządzania zasobami ludzkimi + obsługa delegacji, czasu pracy, harmonogramu zadań + możliwość korzystania na urządzeniach mobilnych + posiada system ocen pracowniczych oraz planowania ścieżki rozwoju zawodowego + pozwala na planowanie i prowadzenie rekrutacji	

⁶ WWW <https://www.elevatosoftware.com/pl/funkcje/> dostęp z dnia 4.03.2022

		<ul style="list-style-type: none"> - brak integracji z LinkedIn, - brak możliwości wgrywania plików z danymi w formatach .csv lub .xlsx - brak filtrowania pracowników po požądanych przez nas kategoriach - system nie jest przygotowany typowo pod branże IT <p>Cena to około 540zł za moduł</p>	
<p>Macrologic S.A.</p> <p>W 2018 roku firma została przejęta przez Asseco Bussiness Solutions⁷</p>	<p>Macrologic Merit -> Softlab HR by Asseco⁸⁹</p>	<ul style="list-style-type: none"> + system do zarządzania zasobami ludzkimi przeprowadzający pracowników od rekrutacji do rozwiązania umowy + posiada możliwość wprowadzania i usuwania danych szczegółowych o zatrudnionych pracownikach, posiada zarządzanie płacami, umowami, - brak integracji z LinkedIn, - brak możliwości wgrywania plików z danymi w formatach .csv lub .xlsx o określonej strukturze danych - brak filtrowania pracowników po požądanych przez nas kategoriach - bardzo rozbudowany system - koszt modułu HR za miesiąc to 1300zł <p>Aplikacja dostosowana raczej do wspomagania zarządzania zasobami ludzkimi już pracujących pracowników a nie kandydatów na stanowiska w IT</p>	<p>+ bardzo przyjazny interfejs dla użytkownika, wiele wykresów i ikon</p>
<p>HCM Deck sp. z o.o.</p>	<p>HCM Deck¹⁰</p>	<ul style="list-style-type: none"> + program przeznaczony głównie do zarządzania rozwojem już pracującego pracownika + platforma do przeprowadzania szkoleń - brak oczekiwanych funkcjonalności 	<p>+ platforma do oceny pracowników i tworzenia profili kompetencyjnych</p>
<p>DMZ</p>	<p>KARO HRMS¹¹</p>	<ul style="list-style-type: none"> + system modułowy pozwala na dostosowanie działania programu do potrzeb rekrutera + możliwość korzystania na urządzeniach mobilnych 	<p>+ system przeznaczony głównie dla branży IT</p>

⁷ WWW <https://www.erp-view.pl/rynek-it/22696-asseco-business-solutions-s-a-przejmuje-macrologic-s-a.html> dostęp z dnia 4.03.2022

⁸ WWW <https://assecobs.pl/macrologic/hr/> dostęp z dnia 4.03.2022

⁹ WWW <https://assecobs.pl/businesscloud/portal-hr/> dostęp z dnia 4.03.2022

¹⁰ WWW <https://hcmdeck.com/pl/> dostęp z dnia 4.03.2022

¹¹ WWW <https://dmz.pl/rozwiązania/zarządzanie-zasobami-ludzkimi/portal-pracownika> dostęp z dnia 4.03.2022

		<ul style="list-style-type: none"> + posiada obsługę elektronicznego obiegu dokumentów wraz z możliwością ich monitorowania na różnych etapach rekrutacji + możliwość budowania grafików, rozliczania delegacji, przeprowadzania szkoleń + zgodność z przepisami RODO - system bardzo rozbudowany - brak integracji z LinkedIn, - brak możliwości wgrywania plików z danymi w formatach .csv lub .xlsx - brak filtrowania pracowników po požądanych przez nas kategoriach 	+ możliwość oceny kompetencji
FPHU Hard-Soft S.C.	Sage HrcLOUD	<ul style="list-style-type: none"> + pozwala na przeprowadzenie szkoleń, rekrutacji wewnętrznych i zewnętrznych, + rozwiązanie chmurowe + posiada obsługę elektronicznego obiegu dokumentów i możliwość komunikacji + definiowanie kalendarza spotkań z wybranymi kandydatami wraz z rejestracją oceny + system modułowy elastycznie dopasowujący się do potrzeb rekrutera + dostęp do procesu rekrutacji menedżerów, którzy mogą monitorować jej przebieg, a także aktywnie w niej uczestniczyć poprzez ocenę kandydatów, dodawania notatek + analiza i raportowanie rekrutacji i kandydatów na poszczególnych etapach procesu. - brak integracji z LinkedIn, - brak możliwości wgrywania plików z danymi w formatach .csv lub .xlsx - brak informacji o sposobie filtracji 	<ul style="list-style-type: none"> + tworzenie i zarządzanie profilem kandydata + wyszukiwanie w bazie aplikacji konkretnej lub wszystkich przeprowadzonych rekrutacji,

1.3.2 Analiza potrzeb – podsumowanie

Porównując wszystkie systemy z tabeli nr 1 można wywnioskować, że istnieją systemy ułatwiające prace pracownika działu HR. Można w nich skorzystać z bardzo wielu funkcjonalności, również takich które nie zawsze są potrzebne w mniejszych firmach zajmujących się tylko rekrutacją.

- Wiele z tych aplikacji jest bardzo rozbudowanych o moduły dla działu płac lub innych. Producenci oferują wykupywanie poszczególnych modułów jednak nie są one tanie. Część z nich w ramach

danego modułu ma również limity nałożone na ilość procesów rekrutacji co nie jest pożądane. Opcje bez limitów wiążą się z wyższymi kosztami.

- Część z tych systemów jak na przykład HCM Deck, Sage Hrccloud, Softlab HR by Asseco, enova365, nie są przeznaczone głównie dla branży IT, lecz są bardziej uniwersalne, co powoduje, że są słabiej przystosowane dla pracowników działu HR w branży IT
- Aplikacje takie jak enova365, Softlab HR by Asseco, HCM Deck, KARO HRMS, Sage Hrccloud nie posiadają też filtrowania po żądanych przez nas kategoriach takich jak technologie programistyczne, znajomość języków obcych i inne
- Aplikacja tomHRM to jedna z najbardziej elastycznych i konkurencyjnych aplikacji. Posiada bardzo dużo funkcjonalności dostosowanych do pracownika działu HR w branży IT. Posiada zautomatyzowaną konwersję plików CV co jest bardzo przydatnym narzędziem, ponadto posiada wyszukiwanie po wybranych kategoriach w stworzonym przez rekrutera „basenie talentów” (własnej bazie danych)
- W większości przeanalizowanych systemów nie ma też integracji z LinkedIn,
- Bardzo dużym plusem jest system oceny kandydata w procesie rekrutacji.

Po przeanalizowaniu wszystkich zalet i wad systemów z tabeli nr 2 można stwierdzić, że żaden z tych systemów nie jest idealny i żaden z nich nie spełnia na dzień dzisiejszy wszystkich oczekiwań firm zajmujących się tylko rekrutacją. Systemy są:

- albo bardzo drogie i małych firm zajmujących się rekrutacją nie stać na taki wydatek
- albo nie posiadają możliwości wyszukiwania po językach programowania i poziomie języka
- albo nie posiadają integracji z LinkedIn
- albo nie posiadają możliwości wgrywania plików Excel do budowania własnej bazy
- albo posiadają funkcje przechowywania kandydatów tylko w zakresie danego procesu rekrutacji, zaś po zakończonym procesie dane kandydata są z automatu usuwane, co jest bardzo dużym minusem w niektórych sytuacjach
- wadą jest też często limit na prowadzone procesy rekrutacyjne w obrębie danego abonamentu

2. Koncepcja systemu

2.1 Koncepcja rozwiązania problemu

Aplikacja jest dedykowana dla pracownika działu HR w jednej konkretnej firmie. Nie planowano sprzedaży aplikacji innym firmom, dlatego zrezygnowano na razie z funkcjonalności rejestracji nowych użytkowników aplikacji. Zatem aplikacja będzie posiadała logowanie tylko dla 1 użytkownika utworzonego na stałe w bazie danych. Nie jest to rozwiązanie idealne, ale istnieje możliwość, że aplikacja zostanie rozszerzona o możliwość rejestracji innych użytkowników w przyszłości. Użytkownik (pracownik HR) będzie mógł tworzyć nowych kandydatów, edytować informacje o kandydatach, usuwać kandydatów z bazy, wprowadzać informacje na temat wyników rozmów rekrutacyjnych (technicznych i nietechnicznych), importować z poziomu interfejsu pliki .csv (.xlsx) lub .json, eksportować całą bazę danych o kandydatach do pliku .csv. Będzie miał również możliwość wyszukiwania kandydatów po nazwisku, miejscowości, języku obcym (wraz z jego poziomem), znajomości technologii programistycznych i znajomości systemów operacyjnych. Będzie miał możliwość zalogowania i wylogowania się.

Aplikacja zostanie podzielona na 2 różne moduły: Pierwszy z nich będzie odpowiadał za obsługę napływających CV w .pdf i obsługę plików .csv z danymi, czyli import i export tych danych. Drugi moduł będzie dotyczył głównie poruszania się po interfejsie użytkownika w tym odczyt, edycja danych i filtrowanie danych po odpowiednich kategoriach. Kategorie te to m.in.: Bootstrap, Adobe, Agile, Android, Angular, AWS, Bash, CSS, C#, C++, C, Delphi, HTML, IOS, Java, JavaScript,

jQuery, Kanban, Less, Linux, .NET, Node.js, Oracle, Perl, Photoshop, PHP, Powershell, Python, React, Ruby, Sass, Scala, Scrum, Spring, SQL, Swift, Vue, VBA, Windows, język angielski, język niemiecki, miejscowość, nazwisko, specjalność (np. tester).

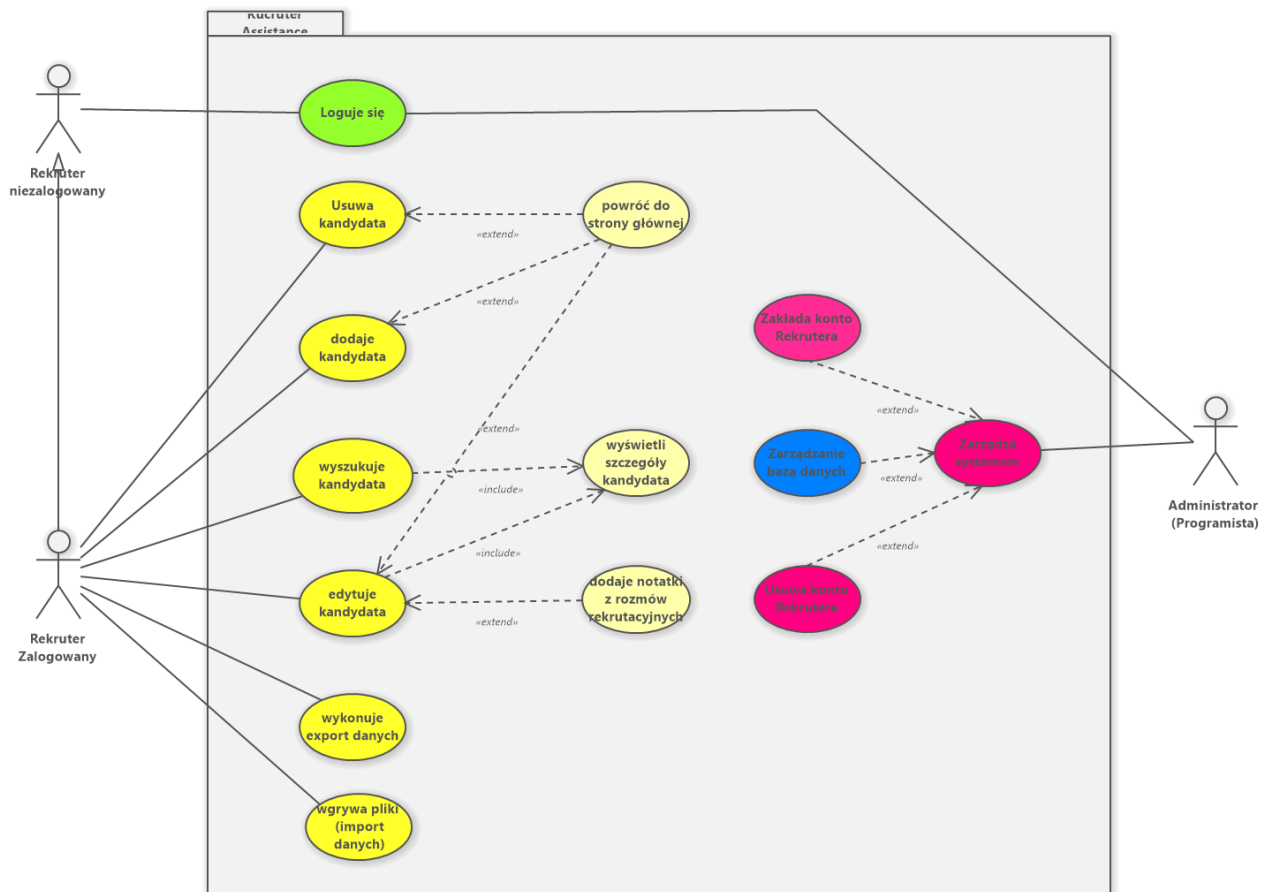
Po pobraniu danych z bazy i wyświetleniu ich użytkownikowi, będzie możliwość edycji każdego z kandydatów z osobna. Po kliknięciu aplikacja przeniesie nas na nową podstronę lub okienko modalne, gdzie można będzie dokonać edycji użytkownika i dokonać aktualizacji w bazie danych. Funkcjonalność usuwania kandydata będzie działała na podobnej zasadzie z tym, że zamiast edycji danych aplikacja zapyta nas czy na pewno chcemy usunąć dany rekord z bazy. Zapobiegnie to przypadkowemu kliknięciu przycisku „usuń”. Korzystając z funkcjonalności edycji danego kandydata, będziemy mieli możliwość jako użytkownik wprowadzić wyniki rozmów technicznych i nietechnicznych a następnie zapisu tych danych w bazie. W razie, gdyby użytkownik kliknął na edycję nie tego kandydata co chciał, to na każdej z podstron będzie możliwość powrotu do strony głównej aplikacji. Użytkownik będzie mógł skorzystać z aplikacji tylko po wcześniejszym zalogowaniu się za pomocą loginu i hasła, zabezpieczy to aplikację przed niepowołanym dostępem do danych przez osoby trzecie. Dodatkowym zabezpieczeniem będzie walidacja danych na backendzie.

Zdecydowano się to przygotować na bazie danych NoSQL (MongoDB) i języka node.js, ponieważ jest to najprostszy typ bazy danych, zaś sposób przechowywania danych przypomina programistyczny słownik (mapę) w postaci JSON, który świetnie się sprawdzi w połączeniu z językiem node.js. Bazy danych NoSQL są idealnym wyborem do aplikacji korzystających z ogromnych ilości nieustrukturyzowanych lub częściowo ustrukturyzowanych danych, które ulegają gwałtownym zmianom. Ten rodzaj baz danych jest bardziej skalowany niż SQL. Nasza baza danych będzie posiadała bardzo dużą ilość danych więc NoSQL będzie lepszym wyborem, poza tym w naszej aplikacji nie zachodzi potrzeba stosowania relacji bazodanowych, dlatego wybrano bazę nierelacyjną.

Dlaczego wybrano język programowania node.js? Ponieważ jest niesamowicie szybki i tak samo jak baza danych MongoDB zyskuje on cały czas na popularności przy tworzeniu aplikacji webowych. Jego szybkość ma wpływ na działanie całej aplikacji. Część jego bibliotek jest napisana w języku C++ z tego też powodu jest dużo szybszy niż inne języki serwerowe, zaś biorąc pod uwagę, że składnia ma podobną do języka JavaScript, bardzo łatwo się go nauczyć.

Inne technologie z których aplikacja będzie korzystać to: JavaScript, npm, express.js (biblioteka node.js), Bootstrap (warstwa wizualna) i scss.

2.2 Diagram przypadków użycia



Rysunek 1 Diagram przypadków użycia.

2.2.1 Scenariusze dla poszczególnych przypadków użycia w systemie

Tabela 2 Scenariusz dla przypadku użycia "usuń kandydata"

Przypadek użycia: usuń kandydata
Aktorzy: Rekruter zalogowany
Scenariusz główny: <ol style="list-style-type: none">1. Rekruter usuwa kandydata.2. Następuje przejście na podstronę.3. Wyświetlony zostaje mu komunikat z zapytaniem czy na pewno chce usunąć kandydata z bazy.4. Rekruter wciska przycisk usuń potwierdzając usunięcie.5. Po usunięciu kandydata, zostaje wyświetlony komunikat o statusie realizacji procesu oraz wyświetlenie przycisku „Powrót do strony głównej”.6. Jeśli Rekruter jednak się rozmyśli może nie usuwać kandydata tylko wcisnąć przycisk „Powrót do strony głównej” i anulować proces usuwania.
Rozszerzenia: powrót do strony głównej

Tabela 3 Scenariusz dla przypadku użycia "dodaj kandydata"

Przypadek użycia: dodaj kandydata
Aktorzy: Rekruter zalogowany
Scenariusz główny: <ol style="list-style-type: none">1. Rekruter klika przycisk „dodaj” kandydata.2. Następuje przejście na podstronę.3. Wyświetlony zostaje formularz wprowadzania nowego kandydata.4. Rekruter wypełnia formularz i wciska przycisk „Dodaj”5. Po dodaniu kandydata, zostaje wyświetlony komunikat o statusie realizacji procesu oraz wyświetlenie przycisku „Powrót do strony głównej”.6. Jeśli Rekruter jednak się rozmyśli może nie dodawać kandydata tylko wcisnąć przycisk „Powrót do strony głównej” i tym samym anulować proces dodawania.
Rozszerzenia: powrót do strony głównej

Tabela 4 Scenariusz dla przypadku użycia "wyszukaj kandydata"

Przypadek użycia: wyszukaj kandydata
Aktorzy: Rekruter
Scenariusz główny: <ol style="list-style-type: none">1. Rekruter wprowadza do formularza dane szukanego kandydata.2. Rekruter klika przycisk „Wyszukaj”.3. W interfejsie zostają wyświetlone wyniki wyszukiwania wraz z szczegółowym opisem kandydatów.4. Proces odbywa się na stronie głównej aplikacji.
Rozszerzenia: wyświetl szczegóły kandydata

Tabela 5 Scenariusz dla przypadku użycia "edytuj kandydata"

Przypadek użycia: edytuj kandydata
Aktorzy: Rekruter zalogowany
Scenariusz główny: <ol style="list-style-type: none">1. Rekruter edytuje wybranego kandydata.2. Następuje przejście na podstronę.3. Rekruter wprowadza zmiany w danych kandydata.4. Rekruter zapisuje zmiany, zmiany zostają wysłane na serwer a następnie do bazy danych.

5. Rekruter zostaje powiadomiony o statusie realizacji procesu (zapisano pomyślnie lub wystąpił błąd).
6. Po edycji kandydata Rekruter może wrócić na stronę główną.
7. Jeśli rekruter jednak się rozmyśli może nie zapisywać zmian tylko powrócić do strony głównej i tym samym anulować proces edycji.
Rozszerzenia: powrót do strony głównej

Tabela 6 Scenariusz dla przypadku użycia "powrót do strony głównej"

Przypadek użycia: powrót do strony głównej
Aktorzy: Rekruter zalogowany
Scenariusz główny: <ol style="list-style-type: none"> 1. Rekruter znajduje się na jednej z podstron. 2. Rekruter klika przycisk „Powrót do strony głównej” 3. Rekruter zostaje przekierowany do strony głównej aplikacji.
Rozszerzenia: -

Tabela 7 Scenariusz dla przypadku użycia "wyświetl szczegóły kandydata"

Przypadek użycia: wyświetl szczegóły kandydata
Aktorzy: Rekruter zalogowany
Scenariusz główny: <ol style="list-style-type: none"> 1. Po akcji wyszukaj lub edytuj, Rekruterowi zostają wyświetlone szczegóły o kandydacie. 2. Rekruter odczytuje interesujące dane.
Rozszerzenia: -

Tabela 8 Scenariusz dla przypadku użycia "dodaj notatki z rozmów rekrutacyjnych"

Przypadek użycia: dodaj notatki z rozmów rekrutacyjnych
Aktorzy: Rekruter zalogowany
Scenariusz główny: <ol style="list-style-type: none"> 1. Podczas edycji kandydata rekruter ma możliwość wprowadzenia w formularz notatek z rozmowy rekrutacyjnej 2. Proces kończy się kliknięciem „Zapisz” lub „Powrót do strony głównej”
Rozszerzenia: -

Tabela 9 Scenariusz dla przypadku użycia "wgrywa pliki"

Przypadek użycia: wgrywa pliki (import danych)
Aktorzy: Rekruter zalogowany
<ol style="list-style-type: none"> 1. Rekruter klika przycisk „Wybierz plik”. 2. Pojawia się okienko systemowe, gdzie można wybrać interesujący nas plik. Rekruter wybiera i klika „Wybierz”. 3. Następnie rekruter klika na przycisk „Prześlij plik”. 4. Plik zostaje przetworzony przez aplikację (sprawdzony zostaje format) i wgrany na serwer, następnie dane z pliku trafiają do bazy danych. Proces jest zautomatyzowany.
Rozszerzenia: -

Tabela 10 Scenariusz dla przypadku użycia "export danych"

Przypadek użycia: wykonuje export danych
Aktorzy: Rekruter zalogowany

Scenariusz główny: <ol style="list-style-type: none"> 1. Rekruter klika przycisk „export”. 2. Zostaje wysłane zapytanie do bazy danych i wygenerowany plik w postaci formatu .xlsx. 3. Pobranie pliku następuje automatycznie do domyślnego folderu na komputerze. Proces jest zautomatyzowany.
Rozszerzenia: -

Tabela 11 Scenariusz dla przypadku użycia "loguje się"

Przypadek użycia: loguje się
Aktorzy: Rekruter niezalogowany, Administrator
Scenariusz główny: <ol style="list-style-type: none"> 1. Rekruter wchodzi na stronę aplikacji. 2. Pojawia się formularz logowania 3. Rekruter wprowadza hasło i login. 4. Rekruter przyciska przycisk „Zaloguj” i zostaje automatycznie przekierowany do strony głównej aplikacji.
Rozszerzenia: -

Tabela 12 Scenariusz dla przypadku użycia "zarządza bazą danych"

Przypadek użycia: zarządza bazą danych
Aktorzy: Administrator (Programista)
Scenariusz główny: <ol style="list-style-type: none"> 1. Aplikacja serwerowa przechowuje dane o kandydatach w bazie. 2. Aplikacja pobiera dane z bazy danych. 3. Aplikacja zapisuje dane do bazy danych. 4. Aplikacja aktualizuje dane w bazie danych. 5. Bazą danych zarządza Programista
Rozszerzenia: -

Tabela 13 Scenariusz dla przypadku użycia "zakłada konto rekrutera"

Przypadek użycia: zakłada konto Rekrutera
Aktorzy: Administrator (Programista)
Scenariusz główny: <ol style="list-style-type: none"> 1. Administrator zakłada konto użytkownika, zapisując dane logowania bezpośrednio w bazie danych. 2. Login i hasło Administrator przekazuje osobiście Rekruterowi na podstawie podpisanej umowy papierowej użytkownika aplikacji między Rekruterem a Administratorem.
Rozszerzenia: -

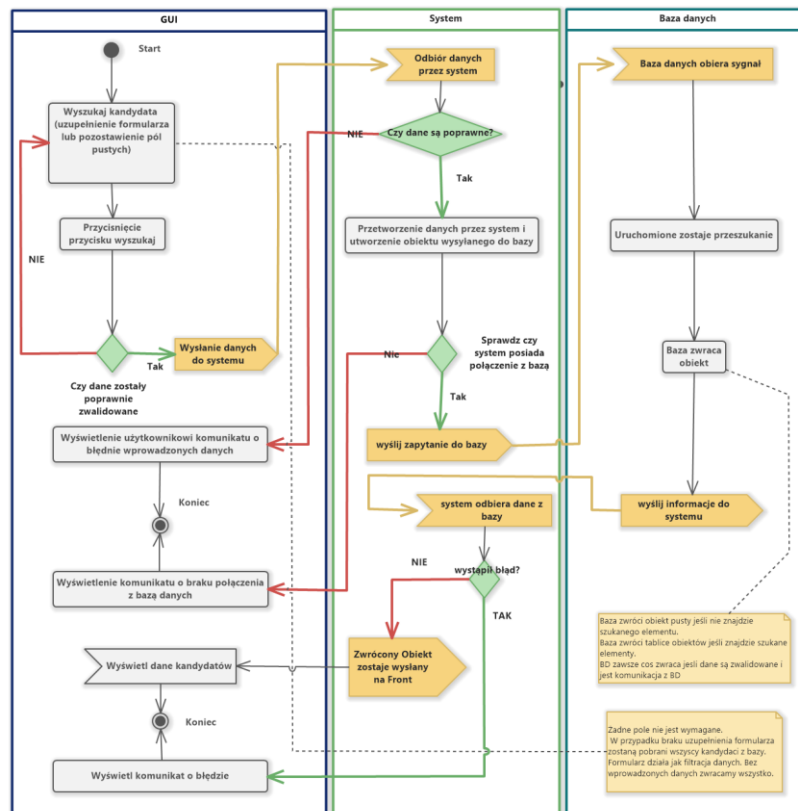
Tabela 14 Scenariusz dla przypadku użycia "usuwa konto rekrutera"

Przypadek użycia: usuwa konto Rekrutera
Aktorzy: Administrator (Programista)
Scenariusz główny: <ol style="list-style-type: none"> 1. Administrator usuwa konto użytkownika usuwając dane logowania z bazy, na podstawie rozwiązania umowy.
Rozszerzenia: -

Tabela 15 Scenariusz dla przypadku użycia "zarządza systemem"

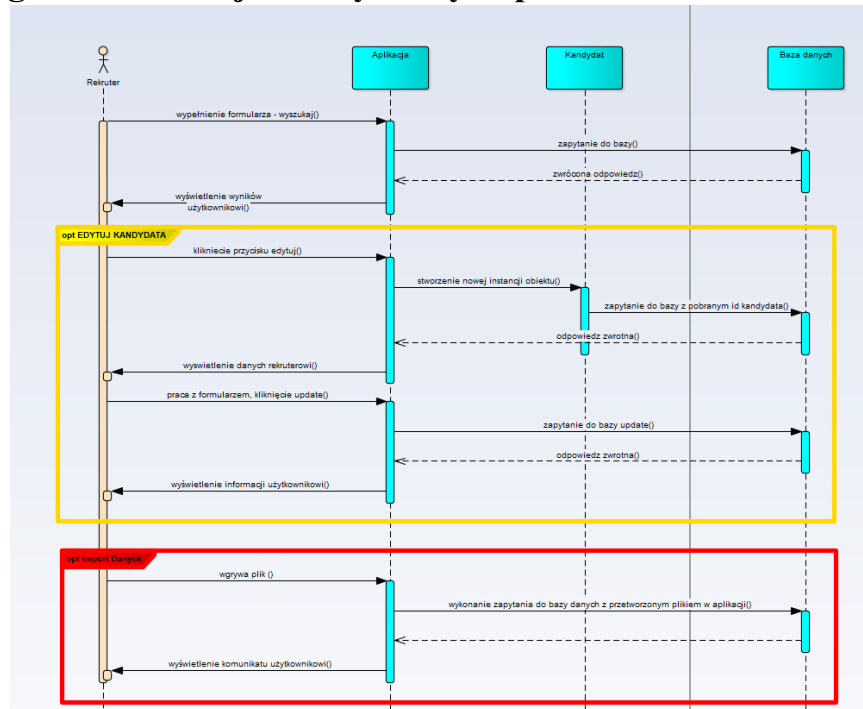
Przypadek użycia: zarządza systemem
Aktorzy: Administrator (Programista)
Scenariusz główny: <ol style="list-style-type: none"> Administrator zarządza kontem użytkownika na podstawie umowy pomiędzy Administratorem i Rekruterem. Zarządzanie odbywa się z poziomu interfejsu bazy danych i dostępu do kodu źródłowego Założono, że nie będzie więcej jak jednego użytkownika aplikacji. W skład tego przypadku użycia wchodzi również naprawa zgłaszanych błędów i support
Rozszerzenia: Usuwa konto Rekrutera, Zarządzanie bazą danych, Usuwa konto Rekrutera

2.3 Diagram aktywności dla przypadku użycia „wyszukaj kandydata”.



Rysunek 2 Diagram sekwencji dla procesu Importu danych z pliku oraz procesu edycji kandydata.

2.4 Diagram sekwencji dla wybranych procesów



Rysunek 3 Diagram sekwencji dla procesu Importu danych z pliku oraz procesu edycji kandydata.

2.5 Wymagania funkcjonalne

Tabela 16 Wymagania funkcjonalne systemu

Nazwa	Opis	Priorytet
Logowanie	Logowanie Rekrutera za pomocą loginu i hasła	wysoki
Wyszukiwanie kandydata	Na stronie głównej będzie możliwość filtrowania i wyszukiwania kandydatów po około 40 różnych kategoriach (filtrach). Między innymi po językach obcych, miejscowości, specjalności, językach programowania. W wyniku wyszukiwania pokażą się nam szczegółowe informacje o kandydacie.	wysoki
Usuwanie kandydata	W aplikacji będzie możliwość wybrania kandydata i usunięcia go z bazy danych.	wysoki
Dodawanie kandydata	W aplikacji będzie możliwość dodawania kandydata do bazy wpisując w formularz co najmniej imię i nazwisko (pola wymagane).	wysoki
Edycja kandydata	W aplikacji będzie możliwość edycji już istniejącego kandydata wraz z dodawaniem notatek z rozmów rekrutacyjnych. Przy edycji będzie możliwość podejrzenia szczegółowych informacji o kandydacie.	wysoki
Import plików .csv	Aplikacja będzie zawierała możliwość importu plików z danymi o kandydatach w formacie .json lub .csv (format Excel)	wysoki
Import plików PDF	Aplikacja będzie umożliwiała import plików CV w formacie .pdf	niski
Export danych	Aplikacja umożliwi export danych o kandydatach z bazy do formatu .xlsx	średni
Nadawanie statusu aplikacji rekrutacyjnej	Aplikacja będzie umożliwiała nadawanie kandydatom statusów aplikacyjnych typu: „1 etap rekrutacji”, „2 etap rekrutacji” „odrzucono”, „zaakceptowano”, „do rozpatrzenia”	niski
Export plików CV	Aplikacja będzie umożliwiała export wcześniej zaimportowanego CV z bazy o wybranym kandydacie w formacie .pdf.	niski
Weryfikacja konta za pomocą kont Google	Wymaganie opcjonalne, logowanie się za pomocą zweryfikowanego konta Google.	niski
Integracja z LinkedIn	Aplikacja umożliwi komunikację z API LinkedIn lub umożliwi integrację danych (za pomocą wtyczki do przeglądarki) konta Rekrutera LinkedIn z systemem HR (nasza aplikacja).	średni
Zakładanie nowego konta	Dodawanie do bazy konta dla Rekrutera z poziomu panelu Administratora.	średni
Usuwanie konta	Usuwanie z bazy danych o koncie użytkownika z poziomu panelu Administratora.	średni
Wyświetlanie statystyk na wykresach o kandydacie	Wyświetlenie za pomocą wykresów danych o umiejętnościach kandydata np. o poziomie języka angielskiego. Oraz wyświetlenie zbiorowych statystyk na	Wysoki

	wykresie o wszystkich kandydatach zwróconych aktualnie przez bazę danych. Funkcjonalność ma być wizualizacją danych o potencjalnym pracowniku lub pracownikach.	
Zarządzanie kontem	Zarządzanie danymi konta użytkownika (rekrutera) z poziomu panelu Administratora.	średni

2.6 Wymagania niefunkcjonalne

Tabela 17 Wymagania niefunkcjonalne systemu

Cecha	Opis	Miara
Bezpieczeństwo	Poufność i stopień zabezpieczenia danych w aplikacji. Zabezpieczenia związane z przechowywaniem danych (RODO)	Algorytmy szyfrowania hasel i plików
Wydajność	Czas reakcji na zapytania ze strony użytkownika, innych aplikacji i systemów. Maksymalna ilość użytkowników korzystających jednocześnie z aplikacji. Struktura aplikacji wraz z danymi nie będzie większa niż 300 GB. Same dane rekordów bez załączników.	Do 500 milisekund Liczba osób 1 Rozmiar aplikacji
Kompatybilność	Współpraca z innymi aplikacjami i systemami	Integracja z LinkedIn
Użyteczność	Łatwość korzystania z aplikacji, estetyka, szybkość uczenia się obsługi aplikacji Aplikacja będzie zgodna z wzorcem User Interface Design. Aplikacja używa czcionki 14 pikseli. Przy rozdzielczości monitora full HD 1980x1020. W aplikacji pola obowiązkowe są oznaczone w inny sposób niż pola nieobowiązkowe. Maksymalny czas odpowiedzi aplikacji na zapytanie użytkownika nie może być dłuższy niż 1 sekunda. Instrukcja dla użytkownika aplikacji zostanie stworzona w formie tutoriala, który będzie dostępny z poziomu strony głównej.	liczba godzin szkolenia, po którym użytkownik może sam korzystać w pełni z aplikacji Wygląd Piksele Czerwone tooltipy Sekundy Tekst informacyjny
Skalowalność	Łatwość zmiany lokalizacji aplikacji, środowiska, exportu danych.	Działanie bez względu na rodzaj komputera, systemu operacyjnego
RWD	Aplikacja będzie responsywna. Jest to w tej chwili standard.	Piksele
Funkcjonalność	Kolory aplikacji to: biały, niebieski, zielony, czerwony i ciemnoszary (kolory loga firmy). Logo firmy jest umieszczone na stronie logowania do aplikacji oraz na stronie głównej w nagłówku strony.	Kolor interfejsu Rozmieszczenie elementów Obsługa przeglądarek

	Aplikacja działa na przeglądarkach internetowych nie starszych wersji niż Mozilla 61, Chrome 68.	
Niezawodność	<p>Czas naprawy wad i błędów, dostępność korzystania z wszystkich cech aplikacji</p> <p>Aplikacja będzie dostępna dla 1 użytkownika na czas trwania umowy korzystania z aplikacji.</p> <p>Aplikacja będzie dostępna 24/7/365 średnio 99,9% czasu.</p> <p>Aplikacja będzie dostępna dla użytkownika w ciągu średnio 99,97% ich czasu pracy w ciągu całego okresu umowy (max 5 lat).</p>	<p>liczba błędów na liczbę godzin użytkowania.</p> <p>Czas trwania umowy</p> <p>Procent czasu</p> <p>Procent czasu pracy</p>
Wsparcie	<p>Wszystkie błędy krytyczne aplikacji zostaną skutecznie naprawione w ciągu maksymalnie 48 h roboczych.</p> <p>Zgłaszanie błędów będzie odbywać się drogą mailową.</p> <p>Wszystkie błędy aplikacji będą monitorowane.</p>	<p>Godziny</p> <p>Komunikacja elektroniczna</p> <p>Rejestrowanie zgłoszonych błędów</p>

3. Środowisko programistyczne

W tym podrozdziale zostaną przedstawione system operacyjny na których będzie zainstalowana aplikacja, środowisko programistyczne wykorzystywane do budowy aplikacji, oraz języki programowania oraz biblioteki wykorzystywane przy jej budowie.

3.1 Systemy operacyjne

Aplikacja została zaprojektowana z myślą o instalacji w dwóch systemach operacyjnych. Nie oznacza to jednak, że nie będzie można z niej korzystać na urządzeniach mobilnych, ponieważ aplikacja będzie wykorzystywać dostęp do Internetu i będzie dostępna z poziomu przeglądarek internetowych również tych na telefonach komórkowych.

Aplikacja może zostać zainstalowana na systemie Linux i Windows. Zdecydowano, że aplikacja musi być kompatybilna z dwoma systemami operacyjnymi, ponieważ:

- na systemie Windows powstała aplikacja. Uruchomienie i zainstalowanie aplikacji w tym systemie jest więc niezbędne do developmentu i testowania aplikacji.
- na systemie Linux aplikacja będzie wdrażana na produkcję. Zdecydowano się na ten krok, ponieważ większość Prywatnych serwerów wirtualnych (VPS) wykorzystuje system operacyjny Linux Ubuntu.

3.2 Środowisko Programistyczne

Do budowy aplikacji wybrano następujące narzędzia:

- Visual Studio Code – wybrano ten edytor, ponieważ jest on w tej chwili najlepszym i najpopularniejszym darmowym edytorem do pisania aplikacji webowych. Jego płatnym konkurentem może być WebStorm firmy JetBrains. Edytor jest bardzo popularny, ponieważ jest lekki, bardzo elastyczny (konfigurowalny) dzięki dużej ilości darmowych i płatnych wtyczek dostępnych w sklepie Marketplace, edytor posiada wbudowane polecenia narzędzia Git co ułatwia pracę, posiada wbudowany debbuger, wyróżnianie i autouzupełnianie składni wybranego języka, posiada wsparcie wdrażania aplikacji na serwery dzięki platformie Microsoft Azure.
- MongoDBCompass – jest to interaktywne narzędzie do optymalizacji i analizowania danych w bazie MongoDB. W tej pracy zostanie wykorzystane do przyspieszenia obsługi bazy danych, które skróci czas pracy developera

- rozproszony system kontroli wersji Git – jest to narzędzie do wersjonowania kodu, niezbędne w tworzeniu oprogramowania a także niezbędne do pracy w zespole. Jako serwer hostingowy gita zostanie wykorzystany GitHub, gdzie będą przetrzymywane wszystkie wersje projektu a następnie z jego pomocą będzie wdrażany projekt na serwer VPS.

3.3 Języki programowania i biblioteki i inne narzędzia

- Do budowy aplikacji od strony backendowej wykorzystany zostanie język serwerowy node.js (czasami nazywany backendowym frameworkiem JavaScript). Wybrano ten język, ponieważ Node jest wykorzystywany do tworzenia skalowalnych aplikacji. Node to wieloplatformowe środowisko uruchomieniowe co jest plusem, posiada także bardzo dobrą dokumentację i wiele bibliotek. Ponadto Node został napisany w języku C++ i JavaScript w oparciu o silnik V8 co czyni go niesamowicie szybkim.

- Wybrana baza danych to baza nierelacyjna MongoDB. Wybrano ten rodzaj bazy, ponieważ w aplikacji nie ma konieczności korzystania z relacji bazodanowych. Ponadto MongoDB wykorzystuje kolekcje w formatach JSON co umożliwia łatwą integrację z Node. MongoDB staje się coraz bardziej popularny, ponieważ jest znacznie szybszy niż bazy MySQL. Duży atut to również skalowalność horyzontalna i wertykalna. Świetnie się sprawdza w połączeniu z językami wykorzystującymi JavaScript takimi jak Node

- do budowy widoków zostanie użyta biblioteka Handlebars lub framework Vue w wersji 3.

- od strony UI/UX zostanie wykorzystany framework CSS3 o nazwie Scss oraz biblioteka Bootstrap w wersji 5.0. W niektórych przypadkach zostanie wykorzystana również technologia flexbox lub grid, oraz biblioteka FontAwesome.

- do budowy wykresów wykorzystano bibliotekę apexchart.js w wersji 3.7 która posiada bardzo duży wachlarz typów wykresów.

4. Prezentacja warstwy użytkowej projektu

Na rysunku 4 przedstawiono główne okno aplikacji. Użytkownik po uruchomieniu aplikacji otrzymuje ekran startowy na którym może:

- Wprowadzić Nazwisko, specjalność i miejscowość szukanego kandydata
- Zaznaczyć technologie, które posiada szukany kandydat
- Zaznaczyć znajomość języka niemieckiego bądź angielskiego oraz wybrać poziom znajomości tych języków, które posiada szukany kandydat
- Wprowadzić inne języki jak np. Hiszpański, które zna szukany kandydat
- Dodać plik w formacie .json lub .csv w celu dodania danych z kandydatami do bazy
- Dodać nowego kandydata do bazy
- Pobrać ponownie dane z bazy po wprowadzeniu w formularz wymaganych umiejętności posiadanych przez kandydatów
- Powyżej i poniżej wyników wyszukiwania zaimplementowano paginację. Użytkownik może przełączać się pomiędzy stronami
- Użytkownik może przejść na podstronę „Wykresy” bądź „O nas”.

The main window features a search interface with the following elements:

- Search Fields:**
 - WPISZ NAZWISKO: (Text input)
 - WPISZ SPECJALNOŚĆ: (Text input)
 - WPISZ MIEJSCOWOŚĆ: (Text input)
 - WPISZ INNY JĘZYK: (Text input)
- Language Selection:**
 - Angielski: A2+
 - Niemiecki: A1+
- Buttons:**
 - Dodaj kandydata
 - Dodaj plik
 - Zapisz dane
 - Pobierz dane
- Skills Grid:** A grid of checkboxes for various skills including Adobe, CSS, JAVA, Node.js, Python, SQL/MySQL, Agile, C#, JavaScript, Oracle, React, Scss, Android, C++, jQuery, Photoshop, Ruby, Swift, Angular, C, Kanban, Vue, VB/VBA, AWS, Delphi, Less, Scala, Bash, HTML, Linux, .NET, Powershell, Spring, and Bootstrap, iOS.
- Pagination:** A row of numbers 1 through 71, with '1' highlighted.
- Candidate Cards:** Three cards are visible at the bottom:
 - ŁUKASZ SĘKOWSKI:** Email: luk.sekowski@gmail.com, Telefon: 507771671, Miejsowość: Rzeszów, Angielski: B1, Niemiecki: -, Inne języki: -, Wiek: 32, Relokacja: nie.
 - KAMILA DREZKA:** Email: dreczkos@gmail.com, Telefon: 697864521, Miejsowość: Rzeszów, Angielski: B1, Niemiecki: -, Inne języki: -, Wiek: 34, Relokacja: tak.
 - KONRAD MAC:** Email: konradmac@o2.pl, Telefon: 506251556, Miejsowość: Siedliska, Angielski: B1, Niemiecki: -, Inne języki: -, Wiek: 35, Relokacja: nie.

Rysunek 4 Główne okno aplikacji, Źródło: opracowanie własne

- Ponadto z poziomu strony głównej można przejść do edycji kandydata lub usuwania kandydata (Rysunek 5) oraz przeglądać dane kandydatów.

The edit view for candidate ROKSOLANA YAKYMYSHYN displays the following information:

- NAME:** ROKSOLANA YAKYMYSHYN
- EMAIL:** yakymyshynroksolana@gmail.com
- TELEFON:** 788717900
- MIEJSCOWOŚĆ:** Rzeszów
- ANGIELSKI:** Angielski B1
- NIEMIECKI:** -
- INNE JĘZYKI:** Rosyjski, Hiszpański
- WIEK:** 20
- RELOKACJA:** nie
- WSPÓŁPRACA Z FIRMAMI:** -
- DOŚWIADCZENIE:** 03-ty lat
- WYMAGANIA FINANSOWE:** < 3000
- TECHNOLOGIE:** -
- KOMENTARZE:** -
- LINKEDIN:** -
- SPECJALNOŚĆ:** Tester
- OSTATNI KONTAKT:** 01-01-2000
- STATUS:** false
- Rozmowa techniczna:** -
- Rozmowa nietechniczna:** -

At the bottom of the form are two buttons: **Edytuj** and **Usuń**.

Rysunek 5 Widok pojedynczego kandydata, Źródło: opracowanie własne

Na rysunku 6 przedstawiono widok edycji kandydata. Okno dodawania kandydata wygląda identycznie z tą różnicą, że wszystkie dane użytkownik musi wprowadzić lub co najmniej wprowadzić nazwisko.

Na ekranie edycji użytkownik może:

- Edytować obecne dane kandydata
- Dodawać informacje w puste pola, które nie były wcześniej uzupełnione
- Dodawać sobie informacje z rozmów rekrutacyjnych technicznych i nietechnicznych

- Dodawać komentarze
- Zmieniać datę ostatniego kontaktu z kandydatem
- Zapisywać wymagania finansowe podczas rozmowy rekrutacyjnej
- Przyciśnięcie przycisku Zapisz powoduje zaktualizowanie danych.
- Przyciśnięcie przycisku anuluj powoduje powrót do strony startowej aplikacji

Edytuj kandydata: Roksolana Yakymyshyn

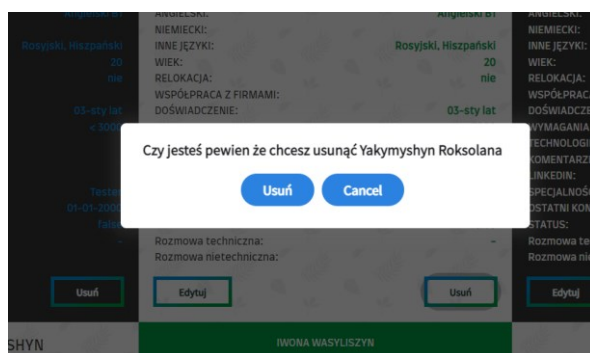
IMIE: Roksolana	NAZWISKO: Yakymyshyn
EMAIL: yakymyshynroksolana@gmail.com	TELEFON: 788717900
MIEJSCOWOŚĆ: Rzeszów	ANGIELSKI: Angielski B1
NIEMIECKI:	INNY JĘZYK: Rosyjski, Hiszpański
WIEK: 20	RELOKACJA: nie
DOSWIADCZENIE: 03-sty	WYMAGANIA FINANSOWE:
DATA KONTAKTU 1.01.2000	LINK DO PROFILU:
TECHNOLOGIE:	
SPECJALNOŚĆ: Tester	
WSPÓŁPRACA Z FIRMAMI:	
KOMENTARZE:	
ROZMOWA TECHNICZNA:	ROZMOWA NIETECHNICZNA:

Zapisz Cancel

Rysunek 6 Widok edycji kandydata, Źródło: Opracowanie własne

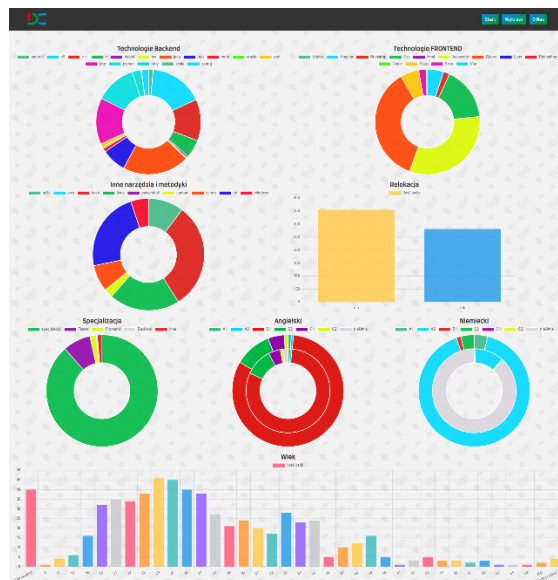
Na rysunku 7 przedstawiono widok usuwania kandydata. Na ekranie usuwania kandydata użytkownik może:

- Potwierdzić przyciskiem usuń, usunięcie kandydata z bazy
- Zrezygnować z usuwania i kliknąć anuluj



Rysunek 7 Usuwanie kandydata, Źródło: Opracowanie własne

Na rysunku 8 przedstawiono widok wykresów. Strona została stworzona w celu wizualizacji obecnie posiadanych kandydatów w bazie. Podzielono te informacje na Backend, Frontend, metodyki inżynierii oprogramowania oraz znajomość systemów operacyjnych, stosunek do Relokacji, specjalizację posiadana przez kandydatów, poziomy języka Angielskiego i niemieckiego oraz wiek kandydatów. Na ekranie usuwania kandydata użytkownik może: jedynie przeglądać wykresy i odczytywać dane. Dzięki temu można wywnioskować jakie umiejętności są niszowe wśród posiadanych informacji o kandydatach. Ta funkcjonalność jest funkcjonalnością dodatkową.



Rysunek 8 Widok wykresów, Źródło: Opracowanie własne

Kolejno przedstawione zostanie działanie wyszukiwania kandydatów.

Na rysunku 9 przedstawiono widok wyniku przeszukiwania po technologiach programistycznych takich jak język C. Wynikiem filtracji są 2 kandydaci, którzy posiadają tę umiejętność. Następnie na rysunku 10 postanowiono zawęzić filtrację do osób, które znają język C i dodatkowo SQL. Baza zwróciła już tylko 1 wynik.

Rysunek 9 przedstawia interfejs wyszukiwania kandydatów. Na górze znajdują się pola do wpisania nazwiska, specjalności i miejscowości, oraz wybrane języki (Angielski, Niemiecki). Poniżej jest lista technologii z zaznaczonymi opcjami. W dolnej części widoczni są dwaj kandydaci: KRZYSZTOF MALETA i KRZYSZTOF MALENKO, z ich danymi kontaktowymi i kwalifikacjami.

Rysunek 9 Widok wyniku przeszukiwania po języku C, Źródło: Opracowanie własne

WPISZ NAZWISKO:

WPISZ SPECJALNOŚĆ:

WPISZ MIĘJSCOWOŚĆ:

Angielski

A2+

Niemiecki

A1+

WPISZ INNY JĘZYK:

☐ Adobe
☐ CSS
☐ JAVA
☐ Node.js
☐ Python
☒ SQL/MySQL

☐ Agile
☐ C#
☐ JavaScript
☐ Oracle
☐ React
☐ Scss

☐ Android
☐ C++
☐ jQuery
☐ Perl
☐ Ruby
☐ Swift

☐ Angular
☒ C
☐ Kanban
☐ Photoshop
☐ Sass
☐ Vue

☐ AWS
☐ Delphi
☐ Less
☐ PHP
☐ Scala
☐ VB/VBA

☐ Bash
☐ HTML
☐ Linux
☐ Powershell
☐ Scrum
☐ Windows

☐ Bootstrap
☐ IOS
☐ .NET
☐ Spring

KRZYSZTOF MALENKO

EMAIL:

a.menteski@gmail.com

TELEFON:

60700009

MIĘJSCOWOŚĆ:

Kraków (Kraków-Podgórze)

ANGIELSKI:

Angielski B2

NIEMIECKI:

Niemiecki A2

INNE JĘZYKI:

Włoski A1

WIEK:

33

RELOKACJA:

tak

WSPÓŁPRACA Z FIRMAMI:

Sakumus, Alfavox

DOŚWIADCZENIE:

03-ty lat

WYMAGANIA FINANSOWE:

C, SQL, Prince2, Scrum, AgilePM, UML

TECHNOLOGIE:

UML

KOMENTARZE:

UML

LINKEDIN:

www.linkedin.com/in/krzysztof-malenski-595437111

SPECJALNOŚĆ:

specjalnosc

OSTATNI KONTAKT:

null

STATUS:

-

Rozmowa techniczna:

-

Rozmowa nietechniczna:

-

Edytuj

Usuń

Rysunek 10 Widok wyniku przeszukiwania po języku C i SQL, Źródło: Opracowanie własne

Na rysunku 11 przedstawiono wyniki wyszukiwania osób które znają język Hiszpański.

WPISZ NAZWISKO:

WPISZ SPECJALNOŚĆ:

WPISZ MIĘJSCOWOŚĆ:

Angielski

A2+

Niemiecki

A1+

WPISZ INNY JĘZYK:

Hiszpański

☐ Adobe
☐ CSS
☐ JAVA
☐ Node.js
☐ Python
☐ SQL/MySQL

☐ Agile
☐ C#
☐ JavaScript
☐ Oracle
☐ React
☐ Scss

☐ Android
☐ C++
☐ jQuery
☐ Perl
☐ Ruby
☐ Swift

☐ Angular
☐ C
☐ Kanban
☐ Photoshop
☐ Sass
☐ Vue

☐ AWS
☐ Delphi
☐ Less
☐ PHP
☐ Scala
☐ VB/VBA

☐ Bash
☐ HTML
☐ Linux
☐ Powershell
☐ Scrum
☐ Windows

☐ Bootstrap
☐ IOS
☐ .NET
☐ Spring

AGATA WYTRWAŁ

EMAIL:

lukasz_intepw.pl

TELEFON:

4896977950

MIĘJSCOWOŚĆ:

Sokolów (Mazowiecki)

ANGIELSKI:

Angielski B1

NIEMIECKI:

Niemiecki A2

INNE JĘZYKI:

Hiszpański

WIEK:

30

RELOKACJA:

tak

WSPÓŁPRACA Z FIRMAMI:

PQS, GZA, Deloitte, Mobica

DOŚWIADCZENIE:

03-ty lat

WYMAGANIA FINANSOWE:

3000 - 5000

TECHNOLOGIE:

C++, Java

KOMENTARZE:

UML

LINKEDIN:

UML

SPECJALNOŚĆ:

01-01-2000

OSTATNI KONTAKT:

false

STATUS:

-

Rozmowa techniczna:

-

Rozmowa nietechniczna:

-

Edytuj

Usuń

KAMIL OŻÓG

EMAIL:

kajetan.sliutka@kaytech.net

TELEFON:

729625153

MIĘJSCOWOŚĆ:

Kraków (Kraków-Śródmieście)

ANGIELSKI:

Angielski B1

NIEMIECKI:

Niemiecki A2

INNE JĘZYKI:

Hiszpański

WIEK:

38

RELOKACJA:

tak

WSPÓŁPRACA Z FIRMAMI:

AdremSoftware, MedStarSurgical, ABB

DOŚWIADCZENIE:

06-ty lat

WYMAGANIA FINANSOWE:

no answer

TECHNOLOGIE:

Java, Python, C#

KOMENTARZE:

UML

LINKEDIN:

https://www.linkedin.com/in/sliutka-98b285132/

SPECJALNOŚĆ:

specjalnosc

OSTATNI KONTAKT:

01-01-2000

STATUS:

false

Rozmowa techniczna:

-

Rozmowa nietechniczna:

-

Edytuj

Usuń

Rysunek 11 Widok wyniku wyszukiwania osób, które znają język Hiszpański

Na rysunku 12 przedstawiono wyniki wyszukiwania osób które znają język Hiszpański, mieszkają w Krakowie i znają język Python

24

Wpisz nazwisko:

Wpisz specjalność:

Wpisz miejscowość:

Wpisz inny język:

Angielski:

Niemiecki:

☐ Adobe ☐ CSS ☐ JAVA ☐ Node.js ☒ Python ☐ SQL/MySQL
☐ Agile ☐ C# ☐ JavaScript ☐ Oracle ☐ React ☐ Scss
☐ Android ☐ C++ ☐ jQuery ☐ Perl ☐ Ruby ☐ Swift
☐ Angular ☐ C ☐ Kanban ☐ Photoshop ☐ Sass ☐ Vue
☐ AWS ☐ Delphi ☐ Less ☐ PHP ☐ Scala ☐ VB/VBA
☐ Bash ☐ HTML ☐ Linux ☐ Powershell ☐ Scrum ☐ Windows
☐ Bootstrap ☐ IOS ☐ .NET ☐ Spring

KAMIL OŻÓG

EMAIL: kajetan.siodutek@kaytech.net

TELEFON: 729625153

MIEJSCOWOŚĆ: Kraków (Kraków-Śródmieście)

ANGIELSKI: Angielski B1

NIEMIECKI: Niemiecki A2

INNE JĘZYKI: Hiszpański

WIEK: 30

RELOKACJA: tak

WSPÓŁPRACA Z FIRMAMI: AdremSoftware, MedStarsurgical, ABB

DOŚWIADCZENIE: 06-ksi lat

WYMAGANIA FINANSOWE: no answer

TECHNOLOGIE: Java, Python, C#

KOMENTARZE:

LINKEDIN: <https://www.linkedin.com/in/siodutek-680285132/>

SPECJALNOŚĆ: specjalnosc

OSTATNI KONTAKT: 01-01-2000

STATUS: false

Rozmowa techniczna: -

Rozmowa nietechniczna: -

Rysunek 12 Wyniki wyszukiwania osób, które znają język Hiszpański, mieszkają w Krakowie i znają język Python

Na rysunku 13 przedstawiono wyniki wyszukiwania kandydatów które znają język angielski na poziomie minimum B2 i wyżej oraz znają język Niemiecki na poziomie minimum A2 i wyżej.

Wpisz nazwisko:

Wpisz specjalność:

Wpisz miejscowość:

Wpisz inny język:

Angielski:

Niemiecki:

☐ Adobe ☐ CSS ☐ JAVA ☐ Node.js ☐ Python ☐ SQL/MySQL
☐ Agile ☐ C# ☐ JavaScript ☐ Oracle ☐ React ☐ Scss
☐ Android ☐ C++ ☐ jQuery ☐ Perl ☐ Ruby ☐ Swift
☐ Angular ☐ C ☐ Kanban ☐ Photoshop ☐ Sass ☐ Vue
☐ AWS ☐ Delphi ☐ Less ☐ PHP ☐ Scala ☐ VB/VBA
☐ Bash ☐ HTML ☐ Linux ☐ Powershell ☐ Scrum ☐ Windows
☐ Bootstrap ☐ IOS ☐ .NET ☐ Spring

JAKUB PISKORZ

EMAIL: csu3802.pl

TELEFON: 729625153

MIEJSCOWOŚĆ: Kraków (Kraków-Śródmieście)

ANGIELSKI: Angielski C1

NIEMIECKI: Niemiecki A2

INNE JĘZYKI: Francuski A2

WIEK: 31

RELOKACJA: tak

WSPÓŁPRACA Z FIRMAMI: IBM, Google

DOŚWIADCZENIE: 03-ty lat

WYMAGANIA FINANSOWE: no answer

TECHNOLOGIE: python, js, SQL, C# (as unity scripting)

KOMENTARZE: noanswer, A2

LINKEDIN: <https://www.linkedin.com/in/siodutek-680285132/>

SPECJALNOŚĆ: specjalnosc

OSTATNI KONTAKT: 01-01-2000

STATUS: false

Rozmowa techniczna: -

Rozmowa nietechniczna: -

KAMIL NOWEK

EMAIL: krzysiek86@gmail.com

TELEFON: 726447539

MIEJSCOWOŚĆ: Kraków (Kraków-Śródmieście)

ANGIELSKI: Angielski C1

NIEMIECKI: Niemiecki A2

INNE JĘZYKI: Rosyjski A2, Bułgarski A2

WIEK: 31

RELOKACJA: nie

WSPÓŁPRACA Z FIRMAMI: Transactor, Infusion

DOŚWIADCZENIE: 06-ksi lat

WYMAGANIA FINANSOWE: no answer

TECHNOLOGIE: C#, JavaScript, TypeScript, .NET, Azure

KOMENTARZE: jestem zainteresowany wykładami, a nie zmianą pracy

LINKEDIN: <https://www.linkedin.com/in/lelto-51979237/>

SPECJALNOŚĆ: specjalnosc

OSTATNI KONTAKT: 01-01-2000

STATUS: false

Rozmowa techniczna: -

Rozmowa nietechniczna: -

KRZYSZTOF MALENKO

EMAIL: a.mentel@gmail.com

TELEFON: 691041018

MIEJSCOWOŚĆ: Kraków (Kraków-Śródmieście)

ANGIELSKI: Angielski B2

NIEMIECKI: Niemiecki A2

INNE JĘZYKI: Włoski A1

WIEK: 35

RELOKACJA: tak

WSPÓŁPRACA Z FIRMAMI: Salomani, Alfavox

DOŚWIADCZENIE: 03-ty lat

WYMAGANIA FINANSOWE: C, SQL, Princez, Scrum, AgilePM, UMLT

TECHNOLOGIE: C, SQL, Princez, Scrum, AgilePM, UMLT

KOMENTARZE: <https://pl.linkedin.com/in/adrmentel>

LINKEDIN: <https://pl.linkedin.com/in/adrmentel>

SPECJALNOŚĆ: specjalnosc

OSTATNI KONTAKT: null


STATUS: null

Rozmowa techniczna: -

Rozmowa nietechniczna: -

Rysunek 13 Wyniki wyszukiwania osób, które znają język angielski na poziomie minimum B2 i znają język Niemiecki na poziomie minimum A2, Źródło: Opracowanie własne

Poniżej przedstawiono również funkcjonalność wgrywania plików do bazy danych. Użytkownik klika w przycisk „Dodaj plik” (Rysunek 14). Następnie pojawia nam się okienko systemowe

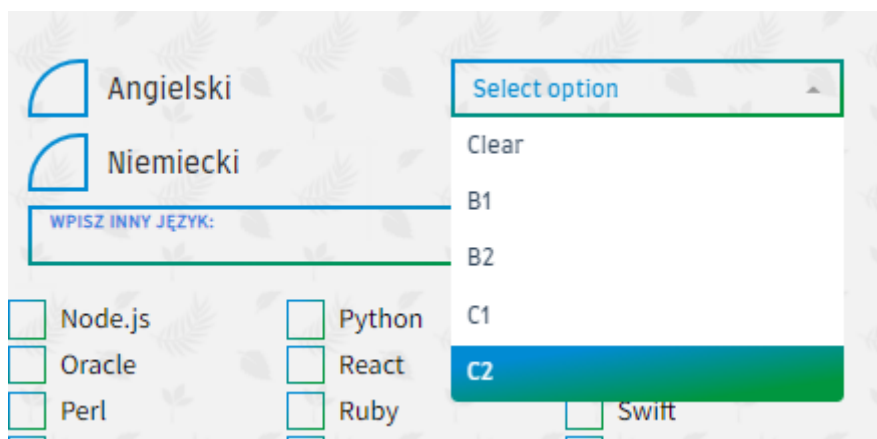


The screenshot displays a web application interface with a light gray background featuring a faint, repeating pattern of small, stylized leaves. In the center, there is a vertical stack of four rectangular buttons. Each button has a blue-to-green horizontal gradient and a subtle drop shadow. The buttons are labeled in white, sans-serif font as follows: 'Dodaj kandydata' (top), 'Dodaj plik' (second from top), 'Zapisz dane' (third from top), and 'Pobierz dane' (bottom). A red arrow, originating from the right side of the 'Dodaj plik' button and pointing towards the right edge of the image, highlights this specific button.

[illegible]

w wyszukiwania została zaimplementowana w ten sposób, że użyto również z paginacji i stronicowania. Jednak paginacja pojawia się tylko i wyłącznie wówczas, gdy liczba wybranych kandydatów przekracza 9 osób. Założono, że przy 1 stronie wynik nie potrzebuje stronicowania w aplikacji.

Przy wyborze poziomu języka jest również dostępna opcja „Clear” wyczyszczenia poziomu języka. Wyczyszczenie tej opcji spowoduje zresetowanie wartości do domyślnych ustawień startowych (Rysunek 16).



26

5. Opis techniczny projektu

W tym projekcie wykorzystano następujące języki i narzędzia:

- Vue3 - frontend
- Node.js - backend
- Express - backend
- MongoDB (Cloud) i Mongoose – baza danych NOSQL
- TypeScript - typowanie
- ChartJs – do wykresów
- Axios – do zapytań API
- Bootstrap - design
- Vuestic UI - design
- Lodash – biblioteka wykorzystywana do pisania logiki
- VueMultiselect – design

Przed uruchomieniem aplikacji należy mieć zainstalowany Git (wersja 2.21) i Node (wersja minimum 16, a najlepiej 16.15.1).

Środowisko programistyczne użyte do budowy projektu to: Visual Studio Code. Aplikacja testowana jedynie na systemie Windows 10. Możliwe, że będzie działać również na systemie operacyjnym Linux z uwagi na to, że była budowana w taki sposób by wdrożyć ją potem na systemy Linux Ubuntu na serwerze VPS. Dokumentacja uruchomieniowa została zamieszczona w pliku README.md w repozytorium pod linkiem https://github.com/bialka104b/recruiters_assistant#readme.

6. Testy jednostkowe i użytkownika

6.1 Testy użytkownika

W tym rozdziale zostanie opisany sposób przeprowadzenia testów akceptacyjnych z realnymi użytkownikami aplikacji. Testy zostaną przeprowadzone na grupie użytkowników nie mających wcześniej styczności z tego typu aplikacją, umiejących posługiwać się komputerem i Internetem. Z testów wykluczono osoby starsze (powyżej 55 roku życia) ponieważ aplikacja jest przeznaczona dla osób w wieku produkcyjnym lub wchodzących na rynek pracy w IT w działach HR. Testy przeprowadzone zostaną na 3 użytkownikach. Pierwsza osoba jest w wieku 17 lat. Druga osoba jest w wieku 26 lat. Trzecia osoba jest w wieku 35 lat. Testy zostaną przeprowadzone na dwóch różnych urządzeniach:

- Laptop HP z przeglądarką Google Chrome w wersji 88.0.4324.190 oraz FireFox Developer Edition, system Windows 10 Pro (użytkownik w wieku 17 lat).

- Komputer PC z przeglądarką Opera GX, Google Chrome w wersji 103.0.5060.66, system Windows 10 Pro (użytkownicy w wieku 26 i 35 lat).

Użytkownicy będą testować wszystkie funkcjonalności, które posiadały priorytet „Wysoki”. Zostaną im zadane konkretne pytania, na które udzielą odpowiedzi w postaci oceny w skali od 1 do 5. Dodatkowo po całym teście każdy z użytkowników udzieli swoich sugestii i opowie o wrażeniach związanych z odbiorem i użytkowaniem aplikacji.

Testy będą miały za zadanie sprawdzić, czy aplikacja jest czytelna, czy dane można łatwo odczytać, czy wszystkie funkcjonalności działają poprawnie, ewentualnie jak długo użytkownicy czekają na odpowiedź serwera o zapytane dane (czy jest to zbyt długi czas dla nich).

W tabeli nr 18 zostały przedstawione przypadki użycia, które będziemy testować w przypadku użytkownika pierwszego w wieku 35 lat w skali 1 do 5.

Tabela 18 Wyniki z przeprowadzonych testów z użytkownikiem nr 1 Sławomir na urządzeniu Laptop HP z systemem Windows 10 Pro

LP	Przypadek użycia	Google Chrome	FireFox Developer Edition	Uwagi
1a	Wyszukiwanie kandydata po technologiach programistycznych	Ocena:5	Ocena:5	
1b	Wyszukiwanie kandydata po nazwisku „WOJCIECHOWSKI”	Ocena:5	Ocena:5	
1c	Wyszukiwanie kandydata po miejscowości „Jarosław”	Ocena:5	Ocena:5	
1d	Wyszukiwanie kandydata po języku angielskim i jego poziomie	Ocena:5	Ocena:5	
1e	Wyszukiwanie kandydata po języku niemieckim i jego poziomie	Ocena:5	Ocena:5	
1f	Wyszukiwanie kandydata po dowolnym języku obcym	Ocena:5	Ocena:5	
1g	Wyszukiwanie kandydata po specjalności	Ocena:5	Ocena:5	
1h	Wyszukiwanie kandydata po wielu kategoriach na raz	Ocena:5	Ocena:5	
2	Usuwanie kandydata	Ocena:5	Ocena:5	
3	Dodawanie kandydata	Ocena:5	Ocena:5	
4	Edycja kandydata	Ocena:5	Ocena:5	
5	Import plików .csv	Ocena:5	Ocena:5	
6	Poprawne wyświetlanie statystyk na wykresach o kandydacie (ocena ogólna)	Ocena:5	Ocena:5	
6a	Odczytanie, ile osób posiada umiejętność języka Python (łatwość odczytania danych z wykresu „Technologie Backend”)	Ocena:5	Ocena:5	
6b	Odczytanie, ile osób posiada umiejętność języka Angular (łatwość odczytania danych z wykresu „Technologie Frontend”)	Ocena:5	Ocena:5	
6c	Odczytanie, ile osób posiada umiejętność pracy w Scrum (łatwość odczytania danych z wykresu „Inne narzędzia i metodyki”)	Ocena:5	Ocena:5	
6d	Odczytanie, ile osób jest gotowa na relokację (łatwość odczytania danych z wykresu „Relokacja”)	Ocena:5	Ocena:5	
6e	Odczytanie, ile osób posiada specjalizację Tester (łatwość odczytania danych z wykresu „Specjalizacja”)	Ocena:5	Ocena:5	
6f	Odczytanie, ile osób posługuje się językiem angielskim na poziomie C1 a ile wcale nie posiada tej umiejętności (łatwość odczytania danych z wykresu „Angielski”)	Ocena:5	Ocena:5	
6g	Odczytanie, ile osób posługuje się językiem niemieckim na poziomie B2 (łatwość odczytania danych z wykresu „Niemiecki”)	Ocena:5	Ocena:5	
6g	Odczytanie w jakim wieku mamy najwięcej kandydatów (łatwość odczytania danych z wykresu „Wiek”)	Ocena:5	Ocena:5	

W tabeli nr 19 zostały przedstawione przypadki użycia, które będziemy testować w przypadku użytkownika drugiego w wieku 26 lat w skali 1 do 5.

Tabela 19 Wyniki z przeprowadzonych testów z użytkownikiem nr 2 Mateusz Laptop HP z systemem Windows 10 Pro

LP	Przypadek użycia	Google Chrome	Firefox Developer Edition	Uwagi
1a	Wyszukiwanie kandydata po technologiach programistycznych	Ocena: 5	Ocena: 5	
1b	Wyszukiwanie kandydata po nazwisku „WOJCIECHOWSKI”	Ocena:5	Ocena:5	
1c	Wyszukiwanie kandydata po miejscowości „Jarosław”	Ocena:5	Ocena:5	
1d	Wyszukiwanie kandydata po języku angielskim i jego poziomie	Ocena:5	Ocena:5	
1e	Wyszukiwanie kandydata po języku niemieckim i jego poziomie	Ocena:5	Ocena:5	
1f	Wyszukiwanie kandydata po dowolnym języku obcym	Ocena:5	Ocena:5	
1g	Wyszukiwanie kandydata po specjalności	Ocena:5	Ocena:5	
1h	Wyszukiwanie kandydata po wielu kategoriach na raz	Ocena:5	Ocena:5	
2	Usuwanie kandydata	Ocena:5	Ocena:5	
3	Dodawanie kandydata	Ocena:5	Ocena:5	
4	Edycja kandydata	Ocena:4	Ocena:4	
5	Import plików .csv	Ocena:5	Ocena:5	
6	Poprawne wyświetlanie statystyk na wykresach o kandydacie (ocena ogólna)	Ocena:5	Ocena:5	
6a	Odczytanie, ile osób posiada umiejętność języka Python (łatwość odczytania danych z wykresu „Technologie Backend”)	Ocena:3	Ocena:3	Kolory za bardzo zbliżone do siebie kolorystycznie
6b	Odczytanie, ile osób posiada umiejętność języka Angular (łatwość odczytania danych z wykresu „Technologie Frontend”)	Ocena:3	Ocena:3	Kolory za bardzo zbliżone do siebie kolorystycznie
6c	Odczytanie, ile osób posiada umiejętność pracy w Scrum (łatwość odczytania danych z wykresu „Inne narzędzia i metodyki”)	Ocena:3	Ocena:3	Kolory za bardzo zbliżone do siebie kolorystycznie
6d	Odczytanie, ile osób jest gotowa na relokację (łatwość odczytania danych z wykresu „Relokacja”)	Ocena:5	Ocena:5	
6e	Odczytanie, ile osób posiada specjalizację Tester (łatwość odczytania danych z wykresu „Specjalizacja”)	Ocena:5	Ocena:5	
6f	Odczytanie, ile osób posługuje się językiem angielskim na poziomie C1 a ile wcale nie posiada tej umiejętności (łatwość odczytania danych z wykresu „Angielski”)	Ocena:5	Ocena:5	
6g	Odczytanie, ile osób posługuje się językiem niemieckim na poziomie B2 (łatwość odczytania danych z wykresu „Niemiecki”)	Ocena:3	Ocena:3	Kolory za bardzo zbliżone do siebie kolorystycznie
6h	Odczytanie w jakim wieku mamy najwięcej kandydatów (łatwość odczytania danych z wykresu „Wiek”)	Ocena:4	Ocena:4	

W tabeli nr 20 zostały przedstawione przypadki użycia, które będziemy testować w przypadku użytkownika drugiego w wieku 17 lat w skali 1 do 5.

Tabela 20 Wyniki z przeprowadzonych testów z użytkownikiem nr 3 Sebastian na urządzeniu nowoczesny Komputer PC z systemem Windows 10 Pro

LP	Przypadek użycia	Google Chrome	Opera GX	Uwagi
----	------------------	---------------	----------	-------

1a	Wyszukiwanie kandydata po technologiach programistycznych	Ocena: 5	Ocena: 5	
1b	Wyszukiwanie kandydata po nazwisku „WOJCIECHOWSKI”	Ocena:5	Ocena:5	
1c	Wyszukiwanie kandydata po miejscowości „Jarosław”	Ocena:5	Ocena:5	
1d	Wyszukiwanie kandydata po języku angielskim i jego poziomie	Ocena:5	Ocena:5	
1e	Wyszukiwanie kandydata po języku niemieckim i jego poziomie	Ocena:3	Ocena:3	
1f	Wyszukiwanie kandydata po dowolnym języku obcym	Ocena:5	Ocena:5	
1g	Wyszukiwanie kandydata po specjalności	Ocena:5	Ocena:5	
1h	Wyszukiwanie kandydata po wielu kategoriach na raz	Ocena:5	Ocena:5	
2	Usuwanie kandydata	Ocena:5	Ocena:5	
3	Dodawanie kandydata	Ocena:5	Ocena: 4	
4	Edycja kandydata	Ocena:5	Ocena:5	
5	Import plików .csv	Ocena:5	Ocena: 4	
6	Poprawne wyświetlanie statystyk na wykresach o kandydacie (ocena ogólna)	Ocena:5	Ocena:5	
6a	Odczytanie, ile osób posiada umiejętność języka Python (łatwość odczytania danych z wykresu „Technologie Backend”)	Ocena:5	Ocena:5	
6b	Odczytanie, ile osób posiada umiejętność języka Angular (łatwość odczytania danych z wykresu „Technologie Frontend”)	Ocena:5	Ocena:5	
6c	Odczytanie, ile osób posiada umiejętność pracy w Scrum (łatwość odczytania danych z wykresu „Inne narzędzia i metodyki”)	Ocena:5	Ocena:5	
6d	Odczytanie, ile osób jest gotowa na relokację (łatwość odczytania danych z wykresu „Relokacja”)	Ocena:5	Ocena:5	
6e	Odczytanie, ile osób posiada specjalizację Tester (łatwość odczytania danych z wykresu „Specjalizacja”)	Ocena:5	Ocena:5	
6f	Odczytanie, ile osób posługuje się językiem angielskim na poziomie C1 a ile wcale nie posiada tej umiejętności (łatwość odczytania danych z wykresu „Angielski”)	Ocena:5	Ocena:5	
6g	Odczytanie, ile osób posługuje się językiem niemieckim na poziomie B2 (łatwość odczytania danych z wykresu „Niemiecki”)	Ocena:5	Ocena: 4	
6g	Odczytanie w jakim wieku mamy najwięcej kandydatów (łatwość odczytania danych z wykresu „Wiek”)	Ocena:5	Ocena:5	

Tabela przedstawia ogólne oceny użytkowników na podstawie wymagań niefunkcjonalnych w skali 1 do 5.

Tabela 21 Wyniki z przeprowadzonych testów z użytkownikami na podstawie wymagań niefunkcjonalnych

Cecha	Opis	Użytkownik 1 - Sławomir	Użytkownik 2 - Mateusz	Użytkownik 3 - Sebastian
Wydajność	Czas reakcji na zapytania ze strony użytkownika, innych aplikacji i systemów.	5	5	5
Użyteczność	Łatwość korzystania z aplikacji, estetyka, szybkość uczenia się obsługi aplikacji. Wygląd, estetyka,	5	4	5

RWD	Wyświetlanie się interfejsu na urządzeniach mobilnych	5	4	4
Funkcjonalność	Kolor interfejsu, obsługa przeglądark, rozmieszczenie elementów	5	5	5
Prędkość działania	Opinia o szybkości odpowiedzi aplikacji na działanie użytkownika	5	5	5
Ilość treści	Opinia o ilości treści zwracanej przez użytkownika. Czy jest za dużo treści lub za mało.	5	5	5
Napotkane problemy w działaniu	Opinia o napotkanych problemach w użytkowaniu	Nie działa klawisz Enter przy wyszukiwaniu 4	Dopracowanie kolorów na diagramach	Małe błędy

Podczas testów użytkownicy zgłaszali następujące uwagi:

- Użytkownik 1 Sławomir zwrócił uwagę na brak automatycznego poprawiania literówek podczas wprowadzania danych przy wpisywaniu miejscowości (nadano priorytet niski). Druga uwaga dotyczyła braku wyszukiwania danych na zdarzenie wciśnięcia klawisza „Enter”, zamiast klikania przycisku „pobierz dane” (nadano priorytet wysoki). Trzecia uwaga dotyczyła braku ikonki „X” w input-ach która umożliwiałaby czyszczenie wprowadzonych wcześniej danych w input (nadano priorytet średni).
- Użytkownik 2 Mateusz zwrócił uwagę na pole „Ilość doświadczenia” w widoku edycji kandydata, stwierdzając, że powinno być tu pole do wprowadzania wyłącznie liczb (nadano priorytet średni). Druga uwaga dotyczyła zbyt podobnych kolorów na wykresach dotyczących wizualizacji danych zwłaszcza „Technologii Backend” i „Technologii Frontend”. Użytkownik nie mógł rozróżnić kolorów (nadano priorytet wysoki).
- Użytkownik 3 Sebastian zwrócił uwagę na najwięcej problemów. Są to:
 - Wyszukiwanie po języku niemieckim na poziomie C1 nie działało poprawnie. Pozostałe poziomy zwracały dane poprawnie (nadano priorytet wysoki).
 - W widoku Dodawania kandydata brak oznaczenia pola „Nazwisko” jako obowiązkowe (priorytet średni)
 - W widoku dodawania kandydata wybieranie daty powinno dać się wybrać maksymalnie dzień dzisiejszy. Nie powinno dać się wybrać daty z przyszłości. (priorytet niski)
 - Po poprawnym dodaniu nowego kandydata wraz z polem „Data kontaktu” przy dodanym kandydacie zostaje wyświetlana niesformatowana data z literką „Z” (priorytet średni).
 - Przy wyszukiwaniu kandydata po języku C++ aplikacja nie zwraca poprawnie wszystkich danych które są w bazie (priorytet wysoki)
 - Po wgraniu pliku .csv trzeba odświeżać stronę, aby zobaczyć dodane dane. Brak automatycznego odświeżenia (priorytet wysoki)
 - Wyszukiwarka nie reaguje na zdarzenie „Enter”

Wszystkie powyższe uwagi użytkowników wzięto pod uwagę, nadano im od razu priorytet i zdecydowano o jak najszybszym usunięciu wszystkich błędów a w pierwszej kolejności tych z priorytetem wysokim, ponieważ mogą one utrudniać normalne codzienne użytkowanie. Nie testowano aplikacji na wielu urządzeniach mobilnych ze względu na ograniczenia sprzętowe. Testy te zostaną wykonane dopiero po wdrożeniu aplikacji na serwer, ponieważ testowanie na emulatorach jest mało wydajne. Obecnie brakuje narzędzi do uruchomienia aplikacji na emulatorze z uwagi na

zbyt małą wydajność komputera, konieczne jest więc wystawienie aplikacji na serwerze chmurowym i dopiero przetestowanie na rzeczywistych urządzeniach mobilnych. Testy za pomocą narzędzi developerskich dla urządzeń mobilnych wykazują, że aplikacja wyświetla się poprawnie, jednak pomimo to z doświadczenia wiem, że nie zawsze tak jest na rzeczywistych urządzeniach.

6.2 Testy jednostkowe komponentów VUE

Testy jednostkowe wykonano przy pomocy biblioteki Vitest którą oferuje Vue3 podczas generowania startowej aplikacji. Testami sprawdzano czy dane przekazywane do komponentów są poprawne i czy ich wartość jest poprawnie renderowana w szablonach komponentów. Sprawdzono również czy przyciski w komponentach poprawnie reagują na zdarzenie („click”) kliknięcia.

Testy przeprowadzono na wszystkich komponentach aplikacji oprócz głównego komponentu App.vue w którym są renderowane wszystkie testowane komponenty. Testowane komponenty to: CreatePerson, DeletePerson, EditPerson, FormRecruter, Header, Graphs, Tables. Poniżej przedstawiono przykładowe testy dla dwóch komponentów.

6.2.1 Przykładowe testy jednostkowe

Aby usunąć użytkownika wymagane jest tylko `_id` użytkownika, które jest zawarte w obiekcie Person. Test (Rysunek 17) miał za zadanie sprawdzić czy po kliknięciu przycisku usuń, zostanie wyrenderowany komponent, który wyświetli Imię i Nazwisko usuwanego użytkownika, sprawdzić, czy przyciski są klikalne i reagują na zdarzenie „click” oraz czy ilość nagłówków renderowanych w komponencie jest równa 1. Czy przypadkiem nie renderujemy podwójnych nagłówków z zapytaniem o potwierdzenie usunięcia kandydata.

```
1  import { describe, it, expect } from "vitest";
2  import { mount, shallowMount } from "@vue/test-utils";
3  import DeletePerson from "../DeletePerson.vue";
4  import { Person } from "@classess/person";
5
6  Complexity is 3 Everything is cool!
7  describe("DeletePerson", () => {
8    it("renders properly", () => {
9      const person = new Person("1234", "Marta", "Kulig");
10     const buttonSelectorCancel = ".cancel";
11     const buttonSelectorDelete = ".delete";
12     const wrapper = shallowMount(DeletePerson, {
13       props: {
14         person: person,
15         showModalDelete: true,
16       },
17     });
18     setTimeout(async () => {
19       expect(wrapper.findAll("h2").length).toEqual(1);
20       expect(wrapper.findAll("h2").at(0)?.text()).toMatch(person.Imie);
21       expect(wrapper.findAll("h2").at(0)?.text()).toMatch(person.Nazwisko);
22       expect(await wrapper.find(buttonSelectorCancel).trigger("click")).toBeTruthy();
23       expect(await wrapper.find(buttonSelectorDelete).trigger("click")).toBeTruthy();
24     }, 100);
25   });
26 });
27
28
29
30
```

Rysunek 17 Test jednostkowy komponentu DeletePerson, który usuwa kandydata

```

1  import { describe, it, expect } from "vitest";
2  import { mount, shallowMount } from "@vue/test-utils";
3  import EditPerson from "../EditPerson.vue";
4  import { Person } from "@classess/person";
5
6  Complexity is 5 Everything is cool!
7  describe("EditPerson", () => {
8    const person = new Person("1234", "Marta", "Kulig");
9    const wrapper = shallowMount(EditPerson, {
10      props: {
11        person: person,
12        showModalDelete: true,
13      },
14      global: { ...
31    },
32  });
33
34  it("renders properly", () => {
35    expect(wrapper.vm.$options.emits.at(0)).toMatch("closeEditModal");
36  });
37
38  it("renders properly", () => {
39    expect(wrapper.vm.$options.emits.at(1)).toMatch("successEdit");
40  });
41
42  it("processes valid props data", async () => {
43    wrapper.setProps({
44      person: person,
45      showModalEdit: true,
46    });
47
48    wrapper.vm.$nextTick().then(async function () {
49      expect(wrapper.vm.person.Nazwisko).toMatchObject("Kulig");
50      expect(wrapper.vm.person.Imie).toMatchObject("Marta");
51      expect(wrapper.findAll("h3").at(0)?.text()).toMatch("Edytuj kandydata: Marta Kulig");
52    });
53  });
54 });

```

Rysunek 18 Trzy testy jednostkowe komponentu EditPerson

Do edycji kandydata wymagane jest przekazanie wcześniej stworzonego obiektu Person który zawiera co najmniej właściwości `_id` i nazwisko oraz pobranie dotychczasowych, istniejących danych o kandydacie.

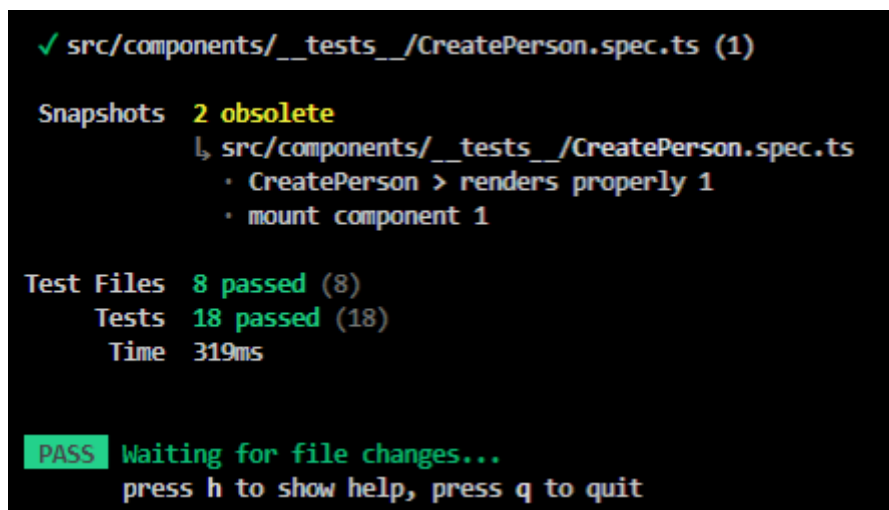
Napisano tutaj trzy testy (Rysunek 18). Celem testu pierwszego w tym komponentcie było przetestowanie czy w opcjach komponentu jest emitowane zdarzenie zamknięcia okienka modalnego które jest wywoływane po kliknięciu przycisku „Anuluj”.

Kolejny test również sprawdzał, czy jest emitowane zdarzenie jednak w tym przypadku chodziło o poprawne zapisanie edytowanych danych na kliknięcie przycisku „Zapisz”. Po emitowaniu tego zdarzenia jest aktualizowany widok kandydatów. Poprawne przejście tego testu jest bardzo istotne z punktu widzenia użytkownika, który oczekuje na GUI aktualizacji danych. To samo tyczy się testu nr 1.

Trzeci test miał za zadanie sprawdzić czy przekazane właściwości do komponentu, wyświetlają się poprawnie w okienku modalnym i czy renderowany tekst nagłówka brzmi „Edytuj kandydata Marta Kulig”. W przypadku niepodania imienia lub nazwiska test by nie przeszedł. Podanie innego imienia bądź nazwiska również spowodowało by że test by nie przeszedł.

6.2.2 Wyniki testów jednostkowych

Wyniki przeprowadzonych testów jednostkowych wykazały, że wszystkie testy przeszły pomyślnie (Rysunek 19). Testowano 8 plików, w których łącznie znalazło się 18 różnych testów. Testowanie trwało 319ms. Na tej podstawie można wyciągnąć wnioski, że komponenty są poprawnie wczytywane, dane przekazywane do komponentów również są poprawne oraz działanie komponentów na różne zdarzenia jest prawidłowe.

A screenshot of a terminal window showing Jest test results. At the top, it says '✓ src/components/_tests_/CreatePerson.spec.ts (1)'. Below that, 'Snapshots 2 obsolete' is shown, followed by a list of snapshots: 'src/components/_tests_/CreatePerson.spec.ts', 'CreatePerson > renders properly 1', and 'mount component 1'. Then, 'Test Files 8 passed (8)', 'Tests 18 passed (18)', and 'Time 319ms' are displayed. At the bottom, a green box with 'PASS' is followed by 'Waiting for file changes...' and 'press h to show help, press q to quit'.

Rysunek 19 Wyniki testów jednostkowych komponentów Vue

7. Repozytorium, system kontroli wersji

Projekt został zrealizowany z wykorzystaniem systemu kontroli wersji git oraz hostingu GitHub. Dokumentacja oraz projekt został umieszczony w repozytorium dostępnym pod adresem: [recruiters_assistant_v2](#). Dokumentacja uruchomieniowa również znajduje się pod tym repozytorium. Starsza wersja tej samej aplikacji znajduje się [RecrAss wersja 1](#)

8. Podsumowanie i wnioski

Podczas implementacji nie udało się zrealizować wszystkich założeń funkcjonalnych, ponieważ brakło czasu, zaś starsza wersja aplikacji została porzucona i przepisana z technologii Handlebars na technologię Vue. Jednak w rozdziale 6 podano link do starszej wersji tej samej aplikacji z zaimplementowanymi niektórymi funkcjonalnościami których nie ma w wersji drugiej. Są to między innymi „export danych”, „logowanie” się, „powrót na stronę główną”. Funkcjonalności związane z aktorem „Administrator” są dostępne nadal tylko z poziomu interfejsu dostępu do bazy danych. W przyszłości planowana jest ich implementacja od strony GUI.

Funkcjonalności, które zostały dodane a nie było ich w założeniach projektowych to między innymi paginacja, ciemny motyw aplikacji, wybór poziomu języka angielskiego i niemieckiego za pomocą Select (wcześniej działało to na zasadzie wpisywania tekstu w input), dodano wizualizację danych z bazy w zakładce „Wykresy”, oraz zakładkę „O Nas” (podstrona o autorze).

Napotkano wiele błędów i problemów związanych z implementacją systemu w Vue3, ponieważ jest to najnowsza wersja Vue i nie wspiera jeszcze wiele bibliotek, ponadto społeczność, która rozwija ten Freamework nie zdołała usunąć jeszcze wszystkich błędów. Są one zgłaszane i powoli rozwiązywane wraz z kolejnymi aktualizacjami Freameworka. W projekcie jest jeszcze wiele do zrobienia, ale na dzień dzisiejszy najważniejsze funkcjonalności, które były zakładane przez klienta, dla którego była pisana aplikacja są zaimplementowane.

Dane w bazie danych są celowo utajnione i nie odzwierciedlają faktycznych osób. Do uruchomienia aplikacji konieczny jest jednak plik z hasłem do bazy danych.

Zgłaszane błędy podczas testów użytkowników zostaną naprawione w dalszej części rozwijania projektu. Plany na rozwój aplikacji to między innymi integracja aplikacji z LinkedIn, implementacja Exportu danych z bazy, implementacja wgrywania plików CV w postaci .pdf, Obsługa logowania i rejestracji, która była zaimplementowana w starszej wersji tego systemu, oraz wybór ilości kafelków z kandydatami na stronie (paginacja).

9. Literatura i źródła:

1. <https://wolski.pro/2015/05/dokumentacja-wymagan-w-oparciu-o-przypadki-uzycia/> z dnia 24.01.2022r.
2. <http://siminskionline.pl/seminarium-inzynierskie/struktura-pracy-inzynierskiej/charakterystykaanaliza-problemu/> z dnia 24.01.2022r.
3. <http://www.jakisystemhr.pl/> z dnia 24.01.2022r.
4. <https://weii.prz.edu.pl/studenci/praca-dyplomowa/jak-napisac-prace-dyplomowa> z dnia 24.01.2022r.
5. https://studiosoftware.pl/blog/wymagania-niefunkcjonalne-aplikacji-czym-sa-i-jak-je-poprawnie-opisac-w-specyfikacji-projektu/#Wymagania_niefunkcjonalne_aplikacji_-_definicja_i_rola_w_procesie_tworzenia_produkту z dnia 24.01.2022r.
6. UML 2.0 w akcji. Przewodnik oparty na projektach - Patrick Graessle, Henriette Baumann, Philippe Baumann, wyd. Helion 2005
7. WWW <http://www.jakisystemhr.pl/> Jaki system HR wybrać? z dnia 4.03.2022r.