**Uniwersytet Jagielloński w Krakowie**

Wydział Fizyki, Astronomii i Informatyki Stosowanej

**Daniel Wielgosz**

Nr albumu: 1157140

**Projekt i implementacja usługowego serwisu ogłoszeniowego**

Praca magisterska  
na kierunku Informatyka Stosowana

Praca wykonana pod kierunkiem

dr hab. Barbara Strug

Zakład Projektowania i Grafiki Komputerowej

Kraków 2024

Spis treści

[1. Wstęp 3](#_Toc175077523)

[1.1 Uzasadnienie wyboru tematu 3](#_Toc175077524)

[1.2 Cel zaprojektowania strony internetowej 4](#_Toc175077525)

[1.3 Zakres pracy 5](#_Toc175077526)

[2. Opis istniejących rozwiązań na rynku 6](#_Toc175077527)

[2.1 Fixly 6](#_Toc175077528)

[2.2 Oferteo 7](#_Toc175077529)

[2.3 Homerun 9](#_Toc175077530)

[2.4 Podsumowanie i porównanie 10](#_Toc175077531)

[3. Projekt aplikacji 11](#_Toc175077532)

[3.1 Wymagania funkcjonalne i niefunkcjonalne 11](#_Toc175077533)

[3.2 Przypadki użycia 13](#_Toc175077534)

[4. Użyte technologie informatyczne 13](#_Toc175077535)

[4.1 PostgresDB 13](#_Toc175077536)

[4.2 Golang 15](#_Toc175077537)

[4.3 React 16](#_Toc175077538)

[5. Implementacja 18](#_Toc175077539)

[5.1 Wstęp 18](#_Toc175077540)

[5.2 Użyte środowiska i narzędzia 18](#_Toc175077541)

[5.3 Schemat bazy danych 19](#_Toc175077542)

[5.4 Architektura kodu po stronie backendu 19](#_Toc175077543)

[5.5 Architektura kodu po stronie frontendu 19](#_Toc175077544)

[5.6 Implementacja wybranych funkcjonalności 19](#_Toc175077545)

[6. Instrukcja obsługi 19](#_Toc175077546)

[6.1 Strony ogólnodostępne 19](#_Toc175077547)

[6.2 Instrukcja obsługi dla zalogowanych klientów 19](#_Toc175077548)

[6.3 Instrukcja obsługi dla zalogowanych specjalistów 19](#_Toc175077549)

[7. Podsumowanie 19](#_Toc175077550)

[8. Bibliografia 19](#_Toc175077551)

# Wstęp

Od wielu lat Internet towarzyszy w codzienności każdego z nas, obejmując większość sfer życia społecznego. Jego ciągły rozwój z każdym dniem daje coraz większe możliwości, od komunikacji z bliskimi, przez zapewnienie rozrywki, aż po dokonywanie rezerwacji wszelkiego rodzaju usług lub wizyt. Szczególnie to ostatnie staje się coraz powszechniejsze, obejmujące coraz więcej dziedzin, chociażby branży medycznej, która umożliwia rezerwację wizyt u lekarzy, sfery szeroko pojętej rozrywki, przez rezerwację stolików w restauracjach, czy także branży budowlanej, poprzez rezerwację usług np. hydraulików, elektryków czy stolarzy. Bez względu na obszar, dla wszystkich użytkowników elektroniczna forma rezerwacji stanowi duże ułatwienie zarówno wśród klientów jak i osób oferujących usługi. Przede wszystkim zapewnia dużą oszczędność czasu oraz pieniędzy, jest również możliwa z każdego miejsca na ziemi [1]. Dodatkowo pozwala poszerzyć horyzonty klientom oraz daje im więcej opcji, tak aby mogli oni dokonać najlepszego dla siebie wyboru. Z kolei osoby oferujące swoje usługi, mogą w ten sposób zareklamować się i dotrzeć do większego grona potencjalnych klientów oraz lepiej zarządzać swoim czasem pracy.

## Uzasadnienie wyboru tematu

Głównym powodem, dla którego autor pracy zdecydował się wybrać taki temat pracy, jest aktualność obecnej tematyki w dobie wciąż rozwijającego się Internetu oraz chęć stworzenia rozwiązania ułatwiającego rezerwację usług wszystkim użytkownikom, a zwłaszcza osobom starszym.

Rosnąca z dnia na dzień cyfryzacja obejmuje kolejne dziedziny naszego życia. Coraz więcej spraw, które do tej pory byliśmy w stanie zrealizować jedynie osobiście, teraz można załatwić nie wychodząc z domu. Jest to również pokłosie pandemii Covid, która wymusiła na rynku powstanie odpowiednich stron do rezerwacji usług, jak np. strony z rezerwacją i realizacją wizyt u lekarza czy usług fachowców z danej dziedziny. W tamtym czasie zapewniało to bezpieczeństwo użytkownikom przez unikanie możliwości zarażenia wirusem przez bezpośredni kontakt. Nie wychodząc z domu, mogli oni po kilku kliknięciach umówić się na wizytę lub zarezerwować usługę.

W obecnych czasach, w których rządzi pośpiech i praktyczność, rozwiązanie to powinno idealnie wpasować się w wymagania przeciętnego użytkownika. Stanowi ono bowiem duże ułatwienie oszczędzające przede wszystkim czas spędzony na wyszukanie np. odpowiedniego fachowca do zrealizowania konkretnej usługi. Dzięki niemu, klienci w jednym miejscu mogą wyszukać osobę, która spełni ich wymagania – nie muszą fizycznie spotkać się z danym fachowcem aby zamówić jego usługę. Dodatkowo posiłkując się ocenami innych użytkowników, zamieszczonymi w serwisie, mogą mieć pewność, że wybrana osoba zrealizuje zlecenie zgodnie z ich kryteriami. Z kolei fachowcy oferujący swoje usługi mogą dotrzeć do większego grona zainteresowanych osób oraz umocnić swoją pozycję na rynku. Oprócz tego, mając wszystkie swoje zlecenia na jednej stronie, mogą w łatwy sposób zarządzać swoim czasem i kolejnymi zleceniami.

Możliwość rezerwowania usług przez Internet może stanowić również ułatwienie dla osób starszych. Ci często ze względu na swój wiek oraz stan zdrowia nie są w stanie radzić sobie z drobnymi pracami lub naprawami. Mogą również nie mieć wystarczającej wiedzy o fachowcach działających w okolicy i reklamujących się online, a tym bardziej sił, aby spędzać czas na długotrwałym ich poszukiwaniu poza domem .

Rozwiązaniem w tej sytuacji mogą być serwisy internetowe umożliwiające rezerwację usług, jednak dla osób starszych, które na co dzień nie korzystają z tego typu udogodnień, oraz nie są zaznajomione z nowymi technologiami, poziom ich skomplikowania może okazać się nie do przejścia. Twórcy stron często zapominają o takich osobach, przez co tworzone przez nich narzędzia stają się zbyt trudne do opanowania przez seniorów. Z tego względu, jednym z powodów podjęcia takiego tematu pracy przez autora była chęć stworzenia intuicyjnego i czytelnego rozwiązania, które dzięki łatwości w obsłudze, nie będzie ograniczało osób starszych, a wręcz będzie stanowić dla nich pomoc i ułatwienie w procesie elektronicznej rezerwacji usług fachowców.

Autor przy wyborze tematu pracy kierował się również chęcią stworzenia serwisu przy wykorzystaniu nowych i popularnych technologii oraz języków programowania. Proces tworzenia narzędzia do rezerwacji usług fachowców stworzył przestrzeń do poszerzenia i ugruntowania wiedzy dotyczącej aktualnych trendów w zakresie architektury stron oraz wykorzystywanych do tego narzędzi. Oprócz tego autor mógł rozwinąć umiejętności dotyczące wyszukiwania informacji, samodzielnego rozwiązywania problemów czy właściwego analizowania i planowania dalszej pracy.

## Cel zaprojektowania strony internetowej

Obecne życie każdego człowieka bardzo mocno opiera się na technologii, która pomaga nam w wielu czynnościach naszego codziennego życia [2]. Dlatego prezentowany w pracy serwis, miał na celu przede wszystkim ułatwienie i przyspieszenie procesu rezerwacji usług fachowców, oraz łatwiejsze dotarcie do potencjalnych klientów przez usługodawców.

Za pomocą takiego serwisu użytkownicy, w sposób szybki, mogą z listy osób wybrać tę, która najbardziej odpowiada ich potrzebom, bez konieczności wychodzenia z domu lub wykonywania niepotrzebnych telefonów. Pomocne przy tym są również opinie innych klientów, a ich duża ilość powinna dać pełniejszy obraz danego fachowca oraz pewność jego niezawodności.

Dodatkowo możliwość zarządzania dokonanymi rezerwacjami z jednego miejsca pozwala użytkownikom łatwiej zaplanować swój czas. Przydatne jest to zwłaszcza dla usługodawców, którzy przy natłoku zleceń potrzebują łatwego sposobu do zarządzania nimi, co również było celem opisywanego serwisu.

Łatwość i prostota poruszania się po nim to również jeden z celów, jakie postawił sobie autor produkując te rozwiązanie. Zbyt duży poziom jego skomplikowania mógłby tylko utrudnić korzystanie z niego przez użytkowników i rodzić w nich niechęć z jego użytkowania. Dzięki swojej intuicyjności serwis skraca czas potrzebny na znalezienie fachowca, nawet w kryzysowej sytuacji.

Prostota jest również ważna w kontekście wykorzystywania serwisu przez osoby starsze. Nawet mimo małego obycia z nową technologią, mogą oni z pomocą przystępnego interfejsu wykorzystać jej potencjał i użyć do realizacji swoich potrzeb.

Kolejnym aspektem opisywanego rozwiązania jest budowania własnej marki przez fachowców. Wielu z nich trudno jest dotrzeć do szerszego grona klientów, z kolei tutaj mają szansę zostać znalezionymi przez innych, a rzetelne opinie zadowolonych klientów mogą tylko ugruntować ich pozycję na rynku.

## Zakres pracy

Niniejsza praca zawiera osiem rozdziałów. W pierwszym z nich zawarto podstawowe informacje, takie jak uzasadnienie wybranego tematu i cel pracy. Opis istniejących na rynku rozwiązań z dziedziny rezerwacji usług przedstawiono w rozdziale drugim. Z kolei w trzecim szczegółowo opisano wymagania funkcjonalne stworzonego serwisu, wraz z opisem jego funkcji, warunkami wstępnymi czy danymi wejściowymi. W kolejnym, czwartym rozdziale, zawarto przypadki użycia, w tym interakcje w ramach danych funkcjonalności systemu. W końcu autor w rozdziale piątym opisuje technologie informatyczne, jakie zostały wykorzystane do zaprojektowania aplikacji. Dalej szczegółowo opisana została implementacja poszczególnych części serwisu w rozdziale szóstym. W kolejnym rozdziale znalazła się instrukcja obsługi serwisu, pokazująca zrzuty ekranu i koje kroki, jakie należy wykonać w celu wykonania danej akcji. W rozdziale ósmym, zamykającym pracę, znalazło się miejsce na podsumowanie, wraz z opisem potencjalnego dalszego rozwoju aplikacji.

# Opis istniejących rozwiązań na rynku

Już od wielu lat, Internet traktowany jest przez użytkowników jako przestrzeń do wyszukiwania usługodawców, którzy mogliby wykonać dane zlecenie. Początkowo do tych celów służyły specjalistyczne fora, na których użytkownicy tworzyli odpowiednie wątki związane z ich problemem, oraz grupy w serwisach społecznościowych, gdzie internauci poprzez umieszczanie postów, starali się znaleźć fachowca odpowiadającego ich potrzebom [3]. Z czasem jednak, zaczęły powstawać dedykowane serwisy, umożliwiające tworzenie ofert oraz wyszukiwanie usługodawców według podanych kryteriów. Obecnie jest ich wiele na rynku, a każdy z nich różni się między sobą sposobem użytkowania czy dostępnymi funkcjonalnościami.

Poniżej przedstawiono i opisano najpopularniejsze z nich, wraz z przybliżeniem ich najważniejszych funkcji.

## Fixly

Obraz zawierający tekst, zrzut ekranu, Czcionka, Strona internetowa

Opis wygenerowany automatycznie

Rysunek 1. Strona główna *Fixly* [4]

*Fixly* jest jednym z najpopularniejszych serwisów pozwalających na rejestrowanie wykonawców oraz wyszukiwanie ich przez zainteresowanych klientów. Założony w 2017 roku przez *Grupę OLX* [5], zawiera wiele funkcji oraz narzędzi ułatwiających poruszanie się po kolejnych stronach oraz rezerwowanie usług.

Już na stronie głównej serwisu (rys. 1) podana jest informacja o dużej ilości zarejestrowanych wykonawców, co stanowi sugestię, że każdy powinien znaleźć w serwisie to czego szuka. Wyświetlona jest również lista dostępnych kategorii oraz pole wyszukiwania, za pomocą których, użytkownik może zaraz po wejściu do serwisu rozpocząć poszukiwania. Użytkownicy mogą również wybrać konkretną lokalizację wykonania usługi, co pozwala bardziej spersonalizować wyniki wyszukiwania.

W zależności od wybranej wcześniej przez użytkownika kategorii, wypełnia on odpowiedni formularz ze szczegółami opisującymi usługę jaka go interesuje – np. jej rodzaj, miejsce wykonania czy termin realizacji. Użytkownik może wprowadzić również dodatkowe dane, takie jak szczegółowy opis czy zdjęcia, aby móc jeszcze lepiej zobrazować swój problem.

Następnie usługodawcy wysyłają swoje zgłoszenia na chęć realizacji usługi do użytkownika. Mając dostęp do profili danych wykonawców, może on porównać ich oferty, przeczytać o nich opinie wystawione przez poprzednich klientów, czy sprawdzić zrealizowane przez fachowca do tej pory usługi. Dzięki temu klient może wybrać osobę, która spełni wszystkie jego oczekiwania.

W trakcie dokonywania wyboru, użytkownik może również skontaktować się z wykonawcą za pomocą podanego numeru telefonu, lub poprzez czat, aby jeszcze dokładniej opisać swój problem i w ten sposób uniknąć nieporozumień w dalszej części realizacji usługi.

Tworzenie ofert i poruszanie się po serwisie jest proste i intuicyjne. Jednak w razie problemów, użytkownicy mają możliwość zgłoszenia się o pomoc podając swój numer telefonu lub adres email w odpowiednim okienku.

Serwis posiada również sekcje, które mają zachęcić użytkowników do skorzystania z niego. Wyświetlane są komunikaty informujące np. o ilości wykonawców, którzy zareagowali na ofertę utworzoną przez innego użytkownika. Dostępny jest również blog, na którym umieszczane są wpisy stanowiące porady z różnych kategorii. Istnieje także sekcja z najpopularniejszymi usługami w ostatnim czasie, co pozwala użytkownikom szybciej utworzyć ofertę.

Serwis istnieje również jako aplikacja na telefony – jest ona łatwo dostępna, dzięki czemu użytkownicy mogą korzystać z serwisu w najwygodniejszy dla nich sposób.

## Oferteo

Obraz zawierający tekst, zrzut ekranu, Czcionka, numer

Opis wygenerowany automatycznie

Rysunek 2. Strona główna *Oferteo* [6]

Kolejnym przykładem serwisu umożliwiającego wyszukiwanie usługodawców w różnych dziedzinach jest *Oferteo*. Podobnie jak w przypadku *Fixly*, już na stronie startowej twórcy zaznaczają swój duży wkład w rynek usługowy – podane są informacje o milionach ofert złożonych do tej pory w serwisie oraz tysiącach zarejestrowanych w nim wykonawców usług.

Oprócz sekcji informacyjnej, na stronie startowej (rys. 2) znajduje się również lista kategorii obsługiwanych przez serwis oraz pole do wyszukiwania konkretnych usług. Po wybraniu jednego ze sposobów wyszukiwania, użytkownik proszony jest o uzupełnienie formularza o szczegółowe informacje, w zależności od wybranej kategorii. Dla uzupełnienia opisu, może on również opisać zakres usługi do zrealizowania, oraz dodać odpowiadające jej zdjęcia.

Po przejściu całego procesu, serwis przygotowuje listę wykonawców spełniających podane kryteria, oraz wysyła do nich utworzoną ofertę wykonania usługi. Z poziomu tej listy, klient może wejść w rozmowę z usługodawcą za pomocą czatu, oraz odwiedzić jego profil – gdzie znajdują się szczegółowe informacje o fachowcu: dane kontaktowe, zrealizowane poprzednio usługi, opinie innych czy zakres wykonywanych prac. Wszystkie te elementy w sprawny i szybki sposób pomagają znaleźć odpowiednią osobę, mając dodatkowo pewność dokładnego zrozumienia problemu, z którym zgłasza się klient.

Po utworzeniu zlecenia, w odpowiedniej zakładce, użytkownik może również przeglądać oferty wystawione przez innych klientów, a po założeniu konta wykonawcy, może w łatwy sposób złożyć ofertę realizacji usługi.

Serwis prócz sekcji z ofertami posiada również blog z poradami. W nim użytkownicy mogą znaleźć poradniki z każdej kategorii usługowej obsługiwanej na stronie. Dzięki czemu przed stworzeniem własnej oferty, mogą oni nabyć dodatkową wiedzę z zakresu danej usługi.

W serwisie znajdują się również opinie specjalistów na temat korzystania z *Oferteo* – podkreślają w ten sposób korzyści jakie przyniosła im rejestracja w serwisie, jednocześnie zachęcając do tego nowych specjalistów

Dodatkowo serwis, na jednej z podstron, opisuje korzyści jakie mogą otrzymać zleceniodawcy, podkreślając zalety swojego rozwiązania. Dostępne są również inne podstrony, dające wskazówki, zarówno dla klientów ja i wykonawców, jak poruszać się po serwisie – wszystko po to, aby użytkownicy mogli w łatwy sposób z niego korzystać.

Oprócz wersji przeglądarkowej, serwis dostępny jest również w formie aplikacji na telefon, dzięki czemu użytkownicy mogą wygodnie z niego korzystać bez względu na miejsce czy sprzęt.

## Homerun

Obraz zawierający tekst, zrzut ekranu, Ludzka twarz, ubrania

Opis wygenerowany automatycznie

Rysunek 3. Strona główna *homerun* [7]

Ostatnim z prezentowanych serwisów do tworzenia ofert z danych kategorii, jest *homerun*. Tak jak poprzednie z przytoczonych przykładów, może on pochwalić się dużą ilością zarejestrowanych wykonawców, co potwierdza informacja na stronie głównej (rys. 3). Daje to pewność użytkownikom, że ze znalezieniem specjalisty spełniającego ich oczekiwania nie powinno być problemu.

Tworzenie oferty można zacząć w serwisie na dwa sposoby – poprzez wpisanie szukanej usługi w polu wyszukiwania, lub poprzez wybranie jednej z listy umieszczonej na stronie głównej. W niej znajdują się najpopularniejsze usługi, z których korzystali użytkownicy w ostatnim czasie – stanowi to duże ułatwienie dla nowych osób i pozwala w szybszy sposób znaleźć to czego szukają.

W kolejnym kroku, w zależności od wybranej kategorii, użytkownik wypełnia formularz ze szczegółami odnośnie usługi. W porównaniu do poprzednich serwisów, ilość informacji które musi podać użytkownik jest mniej, dzięki czemu nie traci on dużo czasu na stworzenie oferty.

Po uzupełnieniu formularza, użytkownik ma wgląd do utworzonej oferty. Wyświetlona zostaje również lista wykonawców, którzy zaoferowali wykonanie usługi – z jej poziomu użytkownik ma możliwość zapoznania się z profilem danego fachowca i wybrać tego, który najbardziej przypadnie mu do gustu. Pomocne w tym stają się szczególnie opinie poprzednich klientów, umieszczone na profilu fachowca.

Dla łatwiejszego tworzenia ofert, serwis posiada również sekcje z listą miast, w której po wybraniu jednego z nich wyświetlana jest lista możliwych do zrealizowania w nim usług – stąd klienta dzieli już tylko kilka kroków do skompletowania oferty.

Oprócz podstron służących do składania ofert, *homerun* posiada sekcję „Pomoc”, gdzie umieszczone są odpowiedzi na najczęstsze pytania użytkowników – aby ich doświadczenie korzystania z serwisu było jak najlepsze i bezproblemowe.

## Podsumowanie i porównanie

Na podstawie przytoczonych powyżej przykładów serwisów można stwierdzić, iż każdy z nich cieszy się popularnością wśród klientów i stanowi dla nich wygodne narzędzie do wyszukiwania wykonawców usług.

*Fixly* dzięki szerokiej gamie kategorii oraz bardzo szczegółowym formularzom, pozwala znaleźć fachowca najlepiej dopasowanego do preferencji klienta. *Oferteo* natomiast posiada wiele sekcji z poradami ułatwiającymi użytkownikom realizację usług oraz poruszanie po serwisie. Z kolei *homerun* dzięki uproszczonemu interfejsowi oraz formularzom, pozwala szybciej znaleźć to, czego oczekuje klient.

Oprócz przytoczonych funkcjonalności wyróżniających opisane serwisy, tym co je łączy jest możliwość wystawiania ofert do realizacji dla fachowców, przeglądanie ich profili oraz kontakt z nimi poprzez specjalny komunikator. Każde z opisanych narzędzi posiada również wielość kategorii, podkategorii oraz szczegółów, które klient musi określić, a które mogą również odpychać poziomem swojej złożoności.

Pod tym kątem, prezentowana w pracy aplikacja różni się od przytoczonych serwisów. Jest to narzędzie zapewniające podstawowe funkcjonalności, które posiadają najpopularniejsze strony w tej kategorii, przy jednoczesnym zawężeniu szczegółowości i rozbudowania, dzięki czemu narzędzie to staje się łatwiejsze w obsłudze i bardziej przystępne chociażby dla osób starszych.

# Projekt aplikacji

## Wymagania funkcjonalne i niefunkcjonalne

### Wymagania funkcjonalne

Wszystkie przytoczone powyżej serwisy posiadają szereg funkcjonalności, umożliwiających użytkownikom rezerwację usług specjalistów oraz ułatwiających im ten proces. Wiele z nich występuje we wszystkich serwisach, są też takie, które wyróżniają każdy z nich.

Opisywany w pracy serwis powinien posiadać więc funkcjonalności pozwalające na wyszukiwanie, rezerwowanie usług i zarządzanie rezerwacjami, zarówno przez klientów jak i usługodawców. Wszystkie z nich powinny być zaimplementowane w taki sposób, aby były jak najbardziej intuicyjne i łatwe w użyciu dla korzystających z serwisu użytkowników. Funkcjonalności będą podzielone według trzech ich rodzajów: zwykłych użytkowników, czyli niezalogowanych osób odwiedzających serwis, oraz klientów i specjalistów, czyli użytkowników zalogowanych. Funkcjonalności, które powinny zostać zaimplementowane dla wspomnianych grup:

* **rejestracja nowego konta**: użytkownicy serwisu powinni mieć możliwość utworzenia konta po wprowadzeniu odpowiednich danych. W przypadku konta klienta, użytkownik powinien podać imię, nazwisko, adres email, hasło do konta, oraz listę adresów (nazwa miasta, nazwa ulicy, numer budynku, opcjonalnie również numer mieszkania). Natomiast usługodawca powinien wprowadzić imię oraz nazwisko, adres email, hasło, nazwę miasta, numer telefonu, opis swojej osoby, nazwę specjalizacji, oraz wybrać z przypisanej do niej listy usługi, podając przy tym zakres cenowy ich realizacji. W oby przypadkach użytkownik powinien podać wszystkie informacje oznaczone gwiazdką („\*”). W przypadku nie wypełnienia któregoś z wymaganych pól, na ekranie zwrócona zostanie informacja o błędzie. Po naciśnięciu przycisku „Zarejestruj” w przypadku powodzenia operacji, na ekranie powinno pojawić się okienko z informacją oraz możliwością przejścia do strony logowania przez naciśnięcie przycisku „Zaloguj”,
* **logowanie do konta**: serwis powinien dać możliwość zarejestrowanym użytkownikom zalogowania się do swojego konta po podaniu adresu email oraz hasła. W przypadku błędu, zostanie on wyświetlony na ekranie, natomiast w przypadku powodzenia operacji, użytkownik zostanie przekierowany na odpowiednią stronę główną – klienta lub specjalisty,
* **wyszukiwanie specjalistów**: klient powinien móc wyszukać specjalistów według specjalizacji, miasta oraz usługi. Po naciśnięciu na stronie głównej „wyszukaj usługę”, klient z listy specjalizacji powinien wybrać jedną z nich. Następnie zostanie on przekierowany na stronę z wyświetloną listą specjalistów przypisanych do wybranej specjalizacji,
* **filtrowanie wyników wyszukiwania**: klient powinien mieć opcję filtrowania listy specjalistów w wyszukiwaniuwedług miejscowości, specjalizacji oraz usługi. Po wybraniu odpowiednich kryteriów i kliknięciu w przycisk „wyszukaj”, na ekranie wyświetlona zostanie lista specjalistów spełniających wprowadzone filtry,
* **przeglądanie listy specjalistów**: po wybraniu kryteriów wyszukiwania, użytkownik powinien mieć możliwość przeglądania listy specjalistów, wraz z podstawowymi informacjami o nich, takich jak imię, nazwisko, specjalizacja oraz miejscowość działalności,
* **przeglądanie kont specjalistów**: serwis powinien umożliwić klientowi odwiedzenie profilu każdego ze specjalistów z wyświetlonej w wyszukiwaniu listy. Po kliknięciu w przycisk „odwiedź profil” przy danym usługodawcy, klient zostanie przekierowany na stronę, na której wyświetlone zostaną szczegółowe informacje o specjaliście, takie jak imię, nazwisko, adres email, numer telefonu, miasto działalności, opis, data rejestracji w serwisie, specjalizacja, lista wykonywanych usług wraz z cenami, oceny innych klientów oraz kalendarz zarezerwowanych usług,
* **rejestracja wizyty**: klient powinien mieć dostęp do funkcji rezerwowania usług u danego specjalisty. Po wybraniu w kalendarzu konkretnej daty, na ekranie wyświetli się formularz, w którym klient powinien wybrać usługę, adres jej realizacji, oraz wprowadzić jej opis. Po kliknięciu przycisku „zarezerwuj”, klient zostanie poinformowany o powodzeniu operacji, lub o ewentualnych błędach w formularzu,
* **przeglądanie listy zarezerwowanych wizyt**: klient oraz specjalista powinien móc przeglądać listę złożonych rezerwacji na usługi. Po kliknięciu w rozsuwanym menu „moje rezerwacje”, zalogowany użytkownik zostanie przekierowany na stronę z kalendarzem wyświetlającym wizyty użytkownika – zaakceptowane, odrzucone, oraz oczekujące na akcję klienta lub specjalisty,
* **modyfikacja wizyt**: system powinien umożliwić klientom oraz specjalistom modyfikację wizyt. W przypadku specjalisty, powinien on móc zaakceptować, odrzucić wizytę lub zmienić jej szczegóły (datę wykonania, szacowaną cenę), natomiast klient powinien być w stanie, po zmianach wprowadzonych przez specjalistę, do zaakceptowania ich, odrzucenia bądź wprowadzenia własnych zmian (jak zmiana daty realizacji),
* **wystawianie ocen**: po wykonaniu usługi, klient powinien mieć opcję wystawienia oceny specjaliście. Po kliknięciu w daną usługę, klient wypełni odpowiedni formularz poprzez wybranie wartości liczbowej, wpisanie opinii w odpowiednim polu oraz kliknięcie przycisku „wystaw opinie”,
* **przeglądanie listy powiadomień**: klient oraz specjalista powinien mieć możliwość przeglądania listy powiadomień utworzonych po wykonaniu działania na rezerwacji,
* **wylogowanie**: zalogowany użytkownik musi być w stanie wylogować się z serwisu. Po otworzeniu rozsuwanego menu i kliknięciu przycisku „wyloguj”, użytkownik zostanie wylogowany i przeniesiony do strony logowania.

Przedstawione wyżej funkcjonalności będą niezbędne do właściwego funkcjonowania serwisu, a odpowiednia ich implementacja znacząco przyczyni się do stworzenia serwisu na miarę potrzeb obecnego rynku.

### Wymagania niefunkcjonalne

Oprócz wymagań funkcjonalnych, ważnym fundamentem zapewniającym wysokiej jakości doświadczenie użytkownika są również wymagania niefunkcjonalne. Definiują one atrybuty jakościowe systemu, jego właściwości, oraz pomagają zapewnić skuteczność jego funkcji, dalsze utrzymanie i rozwój. Określają one również oczekiwania co do jakości działania systemu oraz pożądanego jego zachowania.

Chcąc więc spełnić potrzeby użytkowników serwisu, powinien on posiadać poniższe wymagania:

* **użyteczność**
  + system powinien być łatwy w obsłudze oraz intuicyjny, zwłaszcza dla osób starszych, jego kolorystyka kontrastowa a design przyjazny dla użytkowników,
  + system powinien działać na wszystkich nowoczesnych przeglądarkach,
  + w formularzach, pola obowiązkowe powinny być oznaczone w inny sposób niż pola nieobowiązkowe,
* **bezpieczeństwo**
  + zalogowani użytkownicy systemu, w przypadku bezczynności, po upływie 15 minut powinni zostać automatycznie wylogowani,
  + system powinien również zabezpieczyć odpowiednie strony (np. strony z rezerwacjami) i akcje (np. rezerwacja usługi) przed nieautoryzowanym dostępem innych użytkowników,
* **dostępność / niezawodność** – system powinien być dostępny 24 godziny na dobę, 7 dni w tygodniu przez cały rok,
* **wydajność** - czas odpowiedzi na zapytanie wysłane do API nie powinien przekraczać 5 sekund,
* **konserwacja** – system powinien być łatwy w utrzymaniu i aktualizacji
* **pojemność**
  + system powinien umożliwić rejestrację dużej ilości użytkowników (kilkanaście tysięcy),
  + ilość danych przechowywanych w bazie nie będzie większa niż 32 TB.

Przedstawione i opisane wyżej wymagania niefunkcjonalne, powinny zapewnić zgodność działania systemu z oczekiwaniami użytkowników oraz sprostać wymaganiom panującym na rynku.

## Przypadki użycia

Opisane powyżej wymagania funkcjonalne stanowią ogólny zarys akcji, jakie mogą wykonywać użytkownicy w systemie. Natomiast do opisu sekwencji interakcji, które muszą zajść pomiędzy użytkownikiem a serwisem w celu poprawnego wykonania akcji, służą przypadki użycia. Stanowią one również ważny element w procesie projektowania interfejsu użytkownika

Przedstawia interakcje w ramach danych funkcjonalności systemu, z opisem kolejnych działań podejmowanych przez użytkownika, oraz powiązań między nimi,

# Użyte technologie informatyczne

Na etapie planowania pracy nad implementacją opisywanego, usługowego serwisu ogłoszeniowego, zdecydowano się na wybranie trzech technologii, które miały posłużyć do zarządzania bazą danych aplikacji, wykonywania operacji na tych danych, oraz wyświetlania ich użytkownikom. Głównym kryterium wyboru była ich dotychczasowa znajomość, oraz chęć poszerzenia wiedzy i umiejętności w ich zakresie przez autora. Ważnym czynnikiem była również kwestia dostępności tych rozwiązań na rynku – każde z nich jest powszechnie używane przez szerokie grono programistów na całym świecie, co powinno ułatwić rozwiązywanie problemów, w momencie ich pojawienia. Duża społeczność wpływa również pozytywnie na dalszy rozwój i aktualizacje technologii, co otwiera możliwość dalszego poszerzania serwisu o kolejne funkcjonalności w przyszłości.

Wspomniane trzy technologie, wykorzystane przy implementacji serwisu, zostały podzielone i odpowiadają za backend[[1]](#footnote-1) oraz frontend[[2]](#footnote-2). Ten pierwszy, służący do zarządzania danymi oraz odpowiadający za logikę aplikacji, zaimplementowany został przy użyciu języka *Go*, oraz z wykorzystaniem bazy danych *PostgreSQL*. Z kolei komunikacja z backendem i wyświetlanie danych użytkownikowi odbywa się przy użyciu technologii *React* z wykorzystaniem języka *TypeScript*.

## PostgresDB

Obiektowo relacyjna baza danych stanowi połączenie cech relacyjnej bazy danych oraz obiektowej bazy danych. Dane zorganizowane są w postaci tabel, składających się z wierszy, posiadających unikatowy identyfikator zwany kluczem, oraz kolumn, zawierających nazwy oraz rodzaj przechowywanych danych. Elementy tabel z kolei są połączone ze sobą relacjami za pomocą kluczy obcych. Dodatkowo, funkcje znane z baz relacyjnych są tutaj poszerzone o elementy charakterystyczne dla obiektowości: klasy, obiekty, oraz mechanizmy pozwalające na pracę z nimi, jak chociażby dziedziczenie. Stąd bazy tego typu pozwalają na operowanie na danych w taki sam sposób jak na obiektach.

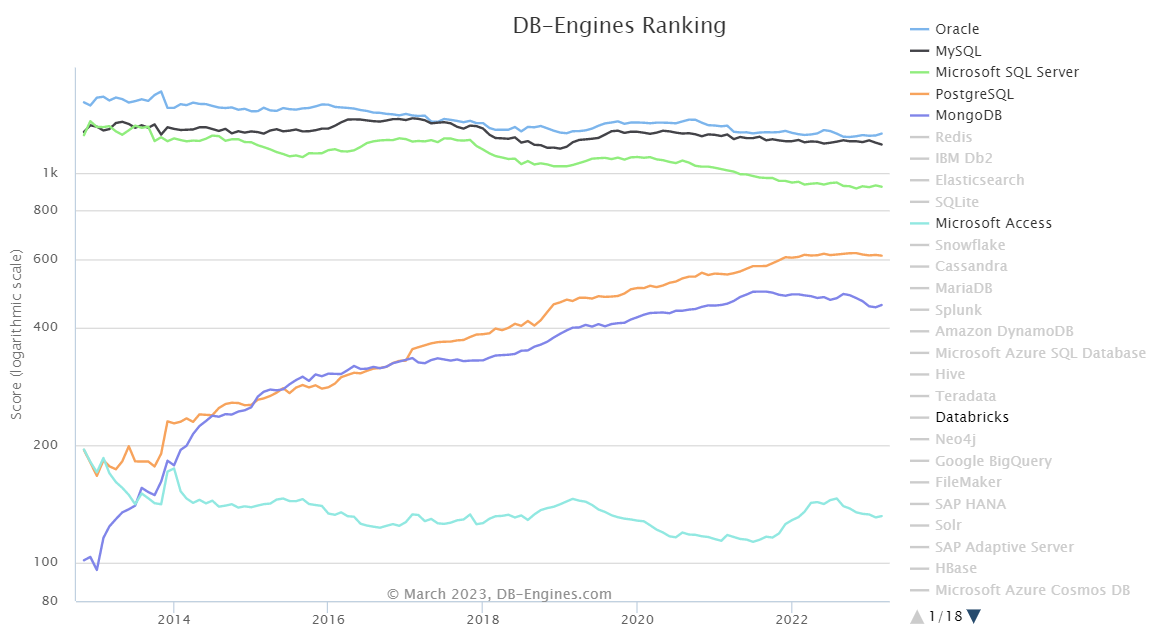
Jednym z najpopularniejszych systemów do zarządzania takimi bazami (*ORDBMS* - Object-Relational Database Management System) jest *PostgreSQL*. Powstały w 1986 r. na Uniwersytecie Kalifornijskim i działający na licencji open-source[[3]](#footnote-3) [8], określany jest jako niezawodny oraz bezpieczny, czym zdobył zaufanie wśród deweloperów. Jest on zgodny ze zbiorem właściwości przetwarzania transakcji w bazach danych takimi jak niepodzielność, spójność, izolacja i trwałość (*ACID*).

Dodatkowo system ten jest zgodny ze wszystkimi najpopularniejszymi systemami operacyjnymi, takimi jak *Windows*, *Linux* czy *MacOS*, dzięki czemu każdy użytkownik może z niego korzystać, bez względu na swoje preferencje.

Kolejną z jego zalet jest skalowalność. Doskonale sprawdza się on w projektach Big Data, gdyż system ten potrafi radzić sobie z największymi bazami danych, posiadających chociażby tabele o rozmiarze 32 TB. Dodatkowo *PostgreSQL* posiada indeksowanie *GiST* (Generalized Search Tree), zapewniające algorytmy do sortowania i wyszukiwania danych [9], co przy dużej ich ilości działa na korzyść użytkowników.

Warto również dodać, że *PostgreSQL* obsługuje format *JSON*[[4]](#footnote-4)do obsługi zapytań i wymiany danych między serwerem a aplikacją WWW. Jest on prosty w użytku, a jego składnia jest czytelna nawet dla niedoświadczonych użytkowników, co stanowi również ułatwienie przy wykonywaniu operacji na danych [10].

*PostgreSQL* zyskuje w ostatnich latach popularność wśród deweloperów. Jego zalety przyciągają nowych użytkowników, o czym świadczą badania *StackOverflow* z 2022 roku, z których wynika, że system ten w tymże roku zanotował wzrost popularności względem roku 2020 o 6% [11]. Potwierdza to również poniższy wykres (rys. 4):



Rys. 4 – Popularność systemów bazodanowych [12]

Wspomniana popularność przekłada się na zaangażowanie użytkowników w rozwój i dalsze usprawnianie opisywanego systemu. W połączeniu z jego niezawodnością, cechy te stanowiły główny powód wyboru *PostgreSQL* przez autora przy implementacji bazy danych serwisu.

## Golang

Za zarządzanie danymi oraz logikę aplikacji, odpowiada język *Go* (znany również jako *Golang*). Stworzony w 2009 roku przez trójkę programistów z firmy *Google* łączy w sobie cechy charakterystyczne zarówno dla języków dynamicznych, takich jak *JavaScript* czy *Python*, oraz języków kompilowanych, jak *C++* czy *Pascal*.

Przy jego tworzeniu, autorzy postawili sobie za cel przede wszystkim utworzenie języka prostego. Projektanci chcieli, aby nie wymagał on złożonego kodowania i był łatwy do nauczenia dla początkujących użytkowników [13]. O jego prostocie świadczy chociażby fakt, iż posiada on jedynie 25 słów kluczowych [14], co przydaje się zwłaszcza mając doświadczenie z poprzednich języków – dzięki temu proces implementacji jest szybszy i bardziej intuicyjny.

Pracę przy kodzie w *Go* ułatwiają również zaimplementowane w nim różne dobre praktyki programistyczne.

Ważną cechą języka jest również wsparcie dla współbieżności. Z pomocą gorutyn[[5]](#footnote-5) oraz kanałów, kilka czynności może być wykonywane jednocześnie z wykorzystaniem wielu wątków przy jednoczesnym minimalnym obciążeniu, co w przypadku aplikacji serwerowych jest szczególnie ważne.

Przyspieszenie pracy w *Go* zapewnia również fakt, iż jest on językiem kompilowanym. Kompilacja odbywa się tutaj o wiele szybciej niż w przypadku języków interpretowanych, a już na jej etapie, dzięki statycznemu typowaniu, programista może wykryć błędy swojego programu. Dzięki kompilacji, również proces debugowania i testowania staje się szybszy.

Dodatkowo, opisywany język posiada w sobie rozbudowaną funkcję garbage collectora, który odpowiada za zarządzanie pamięcią i jej czyszczenie, tak aby była ona jak najlepiej zoptymalizowana. Zapobiega on również wyciekom pamięci, które często mogą być trudne do manualnego skontrolowania przez programistę, przy tym odciążając go i ułatwiając jego pracę.

*Go* ze względu na swoje zalety zyskuje coraz większą popularność, a kolejne duże firmy wykorzystują go do swoich celów. Oprócz *Google*, swoje rozwiązania przy użyciu *Go* implementują takie firmy jak *Dropbox*, *Microsoft* czy *Slack*. Jego wciąż rosnące zasięgi są szczególnie widoczne w raportach *TIOBE Index*. Według nich, na początku 2024 roku, *Go* znalazł się wśród dziesięciu najpopularniejszych języków, a w połowie 2024 roku – zajął miejsce siódme, co potwierdza poniższa tabela (rys. 5):

Obraz zawierający zrzut ekranu, tekst, numer, oprogramowanie

Opis wygenerowany automatycznie

Rys. 5 – Tabela popularności języków programowania w lipcu 2024 r. [15]

Wraz z rosnącą popularnością, rośnie również cały ekosystem języka – powstają nowe biblioteki czy narzędzia, które ułatwiają tworzenie API[[6]](#footnote-6), zarządzanie bazą danych, czy tworzenie aplikacji terminalowych. Duża społeczność użytkowników, coraz więcej forów poruszających tematy związane z *Go*, pozwalają również na łatwe rozwiązywanie pojawiających się problemów. Pomocna przy tym jest również rozbudowana dokumentacja, która zawiera szczegółowe informacje na temat działania poszczególnych pakietów.

Wspomniana popularność, prostota i inne, wymienione wcześniej zalety, stanowiły główny powód wybrania *Go* jako technologii do zarządzania logiką i danymi w opisywanej aplikacji.

## React

*JavaScript* jest dynamicznym, jednowątkowym językiem programowania wysokiego poziomu, służącym do tworzenia interaktywnych stron internetowych. Oprócz wyświetlania statycznych informacji, umożliwia on obsługę zmian treści w zależności od sytuacji, a także dodawanie na stronach takich elementów jak animacje, mapy, filmy czy grafiki 2D i 3D. Wraz z technologiami *HTML*[[7]](#footnote-7) i *CSS*[[8]](#footnote-8), stanowi on podstawę przy tworzeniu stron internetowych.

Został stworzony przez Brendana Eicha, jako element pracy w firmie *Nerscape*, w 1995 roku, a jego celem w tym procesie było uczynienie go językiem prostym i łatwym [16]. Tak też się stało – *JavaScript* jest przystępny i zrozumiały nawet dla początkujących użytkowników, również ze względu na dużą społeczność oraz liczbę zasobów edukacyjnych, pomocnych w procesie nauki. Dodatkowo, do pisania kodu wystarczy jedynie notatnik i przeglądarka.

Do głównych cech, które wyróżniają ten język na tle innych należy fakt, iż jest on językiem interpretowanym. Jego kod wykonywany jest przez interpreter w czasie rzeczywistym, bez potrzeby wcześniejszej jego kompilacji. Dodatkowo wykonuje się on po stronie klienta przez przeglądarkę internetową, co umożliwia modyfikację zawartości strony bez konieczności komunikacji z serwerem po wykonaniu akcji przez użytkownika [17].

Jednak z rozwojem języka zaczął stawać się on coraz bardziej złożony, przez co przez co programiści zaczęli szukać alternatyw. Rozwiązaniem na tą sytuacje miało być powstanie w 2012 roku *TypeScriptu*, czyli języka programowania, który stanowi nadzbiór *JavaScriptu*. Jego składnia jest identyczna, jednak dodaje on funkcjonalności ułatwiające tworzenie aplikacji. Główną z nich jest silniejszy system typów, który pozwala na definiowanie wprost rodzajów zmiennych wykorzystywanych przez programistę, co w dużej mierze przyspiesza pracę oraz ułatwia utrzymanie kodu. Oprócz tego *TypeScript* dostarcza elementy charakterystyczne dla programowania obiektowego – interfejsy, modyfikatory dostępu, abstrakcje, co czyni kod bardziej przejrzystym i stanowi duże ułatwienie dla programistów znających inne języki obiektowe [18]. Dodatkowo TypeScript pozwala na tworzenie wielu niezależnych względem siebie części aplikacji za pomocą modułów. Takie podejście zapewnia lepszą organizację i zarządzanie kodem.

Wspomniane wyżej cechy, oraz fakt iż *TypeScript* jest językiem działającym na licencji open-source, przyczyniły się do szybkiego wzrostu jego popularności wśród programistów oraz dużych firm, które zaczęły z niego korzystać, takich jak *Netflix*, *LinkedIn* czy *Facebook*. Również autor pracy, biorąc pod uwagę powyższe rzeczy, zdecydował się na wybranie tego języka przy implementacji *frontendu*.

*TypeScript* obsługiwany jest przez *React* – jeden z najpopularniejszych frameworków[[9]](#footnote-9) *JavaScriptowych*, którego używanie przekłada się między innymi na większą niezawodność aplikacji. Powstał w 2011 roku z inicjatywy jednego z pracowników *Facebooka*, a głównym celem który mu przyświecał, było uproszczenie procesu budowania interfejsu. Wprowadzone przy tym rozwiązania i innowacje, które pomagają przy tworzeniu złożonych interfejsów użytkowników do różnego rodzaju aplikacji powodują, że *React* zdobywa coraz większe uznanie wśród programistów i podbija branżę programistyczną.

Jedną z nich jest fakt, iż *React* oparty jest w dużej mierze na komponentach. Raz utworzone, mogą być używane w wielu miejscach i na różnych poziomach niezależnie od siebie, co zapobiega duplikowaniu kodu, wpływa pozytywnie na zarządzanie nim i zwiększa wydajność programisty.

Ważną cechą *Reacta*, która wyróżnia go wśród innych rozwiązań tego typu, jest wysoka szybkość implementacji. Dzieje się to między innymi za sprawą dynamicznego budowania interfejsów przez *React*, w których aktualizacje strony odbywają się w czasie rzeczywistym. Stało się to możliwe dzięki używaniu przez framework własnej kopii *Virtual DOM*[[10]](#footnote-10), do której trafiają nawet najmniejsze zmiany, a dopiero później pozwalają na aktualizowanie oryginalnej struktury *DOM* [19].

Oprócz tego w *React* ważny element stanowi przepływ danych, który odbywa się tutaj z góry na dół. Elementy nadrzędne przekazują dane do elementów potomnych, a te nie mogą wpływać na komponenty rodzicielskie. Dzięki temu tylko wyznaczone przez programistę komponenty zostają zaktualizowane, co zwiększa stabilność całego kodu [20].

*React* dzięki swoim zaletom stworzył dużą społeczność programistów, którzy śledzą jego rozwój, a także sami wpływają na niego poprzez tworzenie nowych funkcjonalności i rozszerzeń. Wszystkie z nich można w łatwy sposób pobrać za pomocą managera pakietów *NPM*, a dzięki istniejącym forom i dostępnej dokumentacji można w łatwy sposób zaimplementować je i dostosować do własnego kodu.

Rosnącą popularność *Reacta* zauważają również duże firmy, które coraz częściej decydują się na wykorzystanie go do swoich celów, jak np. *Instagram* czy *Pinterest*. Powszechną popularność potwierdzają również dostarczane co roku raporty i wykresy, jak np. poniższy z 2023 roku (rys. 6):

Obraz zawierający tekst, zrzut ekranu, numer, Równolegle

Opis wygenerowany automatycznie

Rys. 6 – Wykres popularności frameworków webowych w 2023 roku, [21]

Fakt przytoczonych powyżej zalet, wynikającej z nich popularności i szerokiej gamy funkcji i rozwiązań zdecydował, że *React* został wybrany przy implementacji opisywanego serwisu usługowego.

# Implementacja

## Wstęp

## Użyte środowiska i narzędzia

Opisuje narzędzia wykorzystane do implementacji aplikacji,

## Schemat bazy danych

Przedstawia wygląd utworzonej bazy danych, encje, oraz relacje między nimi,

## Architektura kodu po stronie backendu

Opisuje implementacje wybranych elementów, jak np. dostęp do tabel bazodanowych, obsługa zapytań http,

## Architektura kodu po stronie frontendu

Opisuje implementacje wybranych elementów, jak np. komunikacja z backendem, modele danych,

## Implementacja wybranych funkcjonalności

Przedstawia implementacje niektórych funkcjonalności aplikacji, z opisem przepływu danych, wykorzystanych elementach technologii itp.

# Instrukcja obsługi

Zawiera zrzuty ekranu pokazujące kolejne kroki, jakie powinien podjąć użytkownik, aby wykonać konkretną funkcjonalność w serwisie,

## Strony ogólnodostępne

## Instrukcja obsługi dla zalogowanych klientów

## Instrukcja obsługi dla zalogowanych specjalistów

# Podsumowanie

Zawiera informacje o możliwym dalszym rozwoju aplikacji, listę funkcjonalności, które zostały zrealizowane oraz te niezrealizowane.

# Bibliografia

|  |  |
| --- | --- |
| [1] | e-pasje, „e-pasje,” 19 06 2024. [Online]. Available: https://e-pasje.pl/wady-i-zalety-internetu/. |
| [2] | Maksym248, „Silver Sharky,” 05 07 2023. [Online]. Available: https://silversharky.pl/pl/blog/Jak-Technologia-wplywa-na-nasze-zycie/74. |
| [3] | Sprzedajemy.pl, „Gdzie szukać usług, zleceń, fachowców, wykonawców? Internet, gazety, znajomi, gdzie jeszcze?,” [Online]. Available: https://sprzedajemy.pl/poradnik/gdzie-szukac-uslug-zlecen-fachowcow-wykonawcow-internet-gazety-znajomi-gdzie-jeszcze-85b084-art3015. |
| [4] | „Fixly,” [Online]. Available: https://fixly.pl/. |
| [5] | A. Wojtaszek, 28 08 2017. [Online]. Available: https://fixly.pl/niezbednik/fixly-o-nas/. |
| [6] | „Oferteo,” [Online]. Available: https://www.oferteo.pl/. |
| [7] | „homerun,” [Online]. Available: https://homerun.pl/. |
| [8] | „Fingoweb,” [Online]. Available: https://www.fingoweb.com/pl/slownik-it/postgresql/. |
| [9] | „vavatech,” [Online]. Available: https://vavatech.pl/technologie/bazy-danych/postgresql. |
| [10] | „OVHcloud,” [Online]. Available: https://www.ovhcloud.com/pl/lp/postgresql-definition/. |
| [11] | „linuxpolska,” [Online]. Available: https://linuxpolska.com/pl/baza-wiedzy/blog/postgresql-najczesciej-wybierana-baza-danych-przez-deweloperow-i-przyszlosc-dbms/. |
| [12] | „linuxpolska,” [Online]. Available: https://linuxpolska.com/pl/baza-wiedzy/blog/postgresql-najczesciej-wybierana-baza-danych-przez-deweloperow-i-przyszlosc-dbms/. |
| [13] | T. Kozon, „Boring Owl,” 26 9 2023. [Online]. Available: https://boringowl.io/blog/go-jezyk-programowania-ksztaltujacy-przyszlosci. |
| [14] | „mindboxgroup,” [Online]. Available: https://mindboxgroup.com/pl/do-jakich-zastosowan-najlepszy-bedzie-jezyk-go/. |
| [15] | „TIOBE,” [Online]. Available: https://www.tiobe.com/tiobe-index/. |
| [16] | D. Bigosiński, „semcore,” 5 9 2023. [Online]. Available: https://semcore.pl/co-to-jest-java-script-na-stronie-internetowej/. |
| [17] | K. Dopierała, „widoczni,” [Online]. Available: https://widoczni.com/blog/najwiekszy-slownik-marketingu-internetowego-w-polsce/javascript/. |
| [18] | P. Ćwik, „Clockwork Java,” 4 7 2019. [Online]. Available: https://clockworkjava.pl/2019/07/co-to-typescript/. |
| [19] | A. Siczek, „global4net,” 27 8 2020. [Online]. Available: https://global4net.com/blog/ecommerce/czym-jest-react-i-jakie-ma-zalety/. |
| [20] | „smartbees,” 4 2 2020. [Online]. Available: https://smartbees.pl/blog/react-js. |
| [21] | „statista,” [Online]. Available: https://www.statista.com/statistics/1124699/worldwide-developer-survey-most-used-frameworks-web/. |

1. Backend to element infrastruktury niewidoczny dla użytkowników, odpowiadający za logikę i przetwarzanie danych aplikacji, https://cyberfolks.pl/slownik/backend/ [↑](#footnote-ref-1)
2. Frontend odpowiada za wyrenderowanie warstwy wizualnej użytkownikowi, wyświetlenie danych z backendu oraz przekazanie ich z powrotem w formie zapytań, https://webmakers.expert/blog/co-to-jest-front-end [↑](#footnote-ref-2)
3. Open source to oprogramowanie, którego kod źródłowy udostępniany jest bezpłatnie do użytku, rozpowszechniania i modyfikowania przez użytkowników, https://harbingers.io/definicje/open-source [↑](#footnote-ref-3)
4. JSON to otwarty format zapisu i wymiany struktur danych, składający się z par klucz – wartość, https://grupa-improve.pl/json-co-to-jest/ [↑](#footnote-ref-4)
5. Gorutyna to wydajny wątek wykonujący funkcje, https://sii.pl/blog/wspolbieznosc-w-go/ [↑](#footnote-ref-5)
6. API (Application Programming Interface) to zbiór reguł i protokołów umożliwiających komunikację między oprogramowaniem różnego typu, https://coderslab.pl/pl/blog/co-to-jest-api-wszystko-o-interfejsie-programowania-aplikacji [↑](#footnote-ref-6)
7. HTML (Hyper Text Markup Language) to standardowy język znaczników, wykorzystywany przy tworzeniu i wyświetlaniu stron internetowych w przeglądarce, https://adboosters.pl/tworzenie-stron-internetowych/co-to-jest-html-i-do-czego-sluzy-poznaj-najpopularniejsze-znaczniki-i-atrybuty/ [↑](#footnote-ref-7)
8. CSS (Cascading Style Sheets) to język używany do opisu układu elementów i sposobu ich wyświetlania na stronie, https://udigroup.pl/blog/co-to-jest-jezyk-css-kaskadowe-arkusze-stylow/ [↑](#footnote-ref-8)
9. Framework to szkielet budowy aplikacji, dostarczający biblioteki oraz komponenty potrzebne do działania aplikacji, https://mayko.pl/slownik-e-marketingu/framework/ [↑](#footnote-ref-9)
10. DOM (Document Object Model) reprezentuje w pamięci komputera strukturę dokumentu HTML w postaci modelu obiektowego, https://miroslawzelent.pl/kurs-javascript/document-object-model-dom-hierarchia/ [↑](#footnote-ref-10)