

KRZYSZTOF DRAGAN

Numer albumu: 083524

**Aplikacja internetowa
do wyszukiwania połączeń lotniczych**

Praca dyplomowa inżynierska
na kierunku Informatyka

Opiekun pracy dyplomowej:

dr inż. Arkadiusz CHROBOT

Katedra Systemów Informatycznych

POLITECHNIKA ŚWIĘTOKRZYSKA
WYDZIAŁ ELEKTROTECHNIKI, AUTOMATYKI I INFORMATYKI

Zatwierdzam:

PROF. DR
ds. Kształcenia i Sprawy Studenckich
na Studiach Stacjonarnych
Wydziału Elektrotechniki, Automatyki i Informatyki

Rok akademicki: 2018/19

Temat nr. 70/1579/2018/19

Dnia: 21 marca 2018

dr inż. Andrzej Stoblecki

ZADANIE NA PRACĘ DYPLOMOWĄ

Studiów pierwszego stopnia na kierunku
INFORMATYKA

Wydano dla studenta: **Krzysztof Dragan**

I. Temat pracy:

Aplikacja internetowa do wyszukiwania połączeń lotniczych

II. Plan pracy:

1. Wstęp
2. Opis rozwiązywanego zagadnienia
3. Przegląd istniejących rozwiązań
4. Projekt aplikacji
5. Implementacja
6. Testy
7. Uwagi i wnioski

III. Cel pracy:

Celem pracy jest stworzenie aplikacji internetowej, która będzie zbierała informacje o dostępnych połączeniach lotniczych i umożliwiała użytkownikowi zaplanowanie na ich podstawie podróży. Źródłem informacji o dostępnych połączeniach lotniczych, które aplikacja będzie pobierała automatycznie, powinny być strony internetowe agregujące dane o lotach lub serwisy przewoźników. Opracowane oprogramowanie powinno pozwalać na określenie takich parametrów wyszukiwania lotów, jak: czas podróży, termin i miejsce jej rozpoczęcia i zakończenia, liczba przesiadek, cena przelotów i gabaryty oraz waga zabieranego bagażu. Aplikacja powinna się składać z części serwerowej oraz klienckiej i być zaimplementowana w języku Java oraz przy użyciu technologii związanych z tym językiem.

IV. Uwagi dotyczące pracy:

V. Termin oddania pracy: **30 stycznia 2019**

VI. Konsultant:

Kierownik Zakładu
Zakładu Informatyki
Katedry Systemów Informatycznych
Wydziału Elektrotechniki, Automatyki i Informatyki
dr hab. inż. Roman Stokłowski, prof. PŚK
(pieczęć i podpis)

Opiekun pracy dyplomowej

Andrzej Chudzik
(podpis)

.....
(imię i nazwisko)

Temat pracy dyplomowej celem jej wykonania otrzymałem:

Kielce, dnia 21.03.2018 r. Dragan Krzysztof
czytelny podpis studenta



Kielce, dnia 12.10.2018

Krzysztof Dragan 083524
Imię i nazwisko studenta nr albumu

ul. Denkowska 83, Ostrowiec Świętokrzyski
Adres zamieszkania

Informatyka, Systemy Informacyjne, IV rok, stacjonarne
Kierunek, specjalność, rok studiów, rodzaj studiów (stacjonarne, niestacjonarne)

dr inż. Arkadiusz Chrobot
Opiekun pracy dyplomowej inżynierskiej/magisterskiej*

OŚWIADCZENIE

Przedkładając w roku akademickim 2018/19 opiekunowi pracy dyplomowej inżynierskiej/magisterskiej*, powołanemu przez Dziekana Wydziału Elektrotechniki Automatyki i Informatyki Politechniki Świętokrzyskiej, pracę dyplomową inżynierską/magisterską* pod tytułem:

Aplikacja internetowa do wyszukiwania połączeń lotniczych

oświadczam, że:

- 1) przedstawiona praca dyplomowa inżynierska/magisterska* została opracowana przeze mnie samodzielnie, stosownie do wskazówek merytorycznych opiekuna pracy,
- 2) przy wykonywaniu pracy dyplomowej inżynierskiej/magisterskiej* wykorzystano materiały źródłowe, w granicach dozwolonego użytku wymieniając autora, tytuł pozycji i miejsce jej publikacji,
- 3) praca dyplomowa inżynierska/magisterska* nie zawiera żadnych danych, informacji i materiałów, których publikacja nie jest prawnie dozwolona,
- 4) przedstawiona praca nie była wcześniej przedmiotem procedur związanych z uzyskaniem stopnia zawodowego/naukowego w wyższej uczelni,
- 5) niniejsza wersja pracy jest identyczna z załączoną treścią elektroniczną (na CD i w systemie Archiwum Prac Dyplomowych).

Przyjmuję do wiadomości, iż w przypadku ujawnienia naruszenia przepisów ustawy o prawie autorskim i prawach pokrewnych, praca dyplomowa inżynierska/magisterska* może być unieważniona przez Uczelnię, nawet po przeprowadzeniu obrony pracy.

Zostałem uprzedzony:

- 1) o odpowiedzialności karnej wynikającej z art. 115 ust. 1 i 2 ustawy z dnia 4 lutego 1994 r. o prawie autorskim i prawach pokrewnych (t. j. Dz. U. z 2006 r. Nr 90, poz. 631 z późn. zm.): „Kto przywłaszcza sobie autorstwo albo wprowadza w błąd co do autorstwa całości lub części cudzego utworu albo artystycznego wykonania, podlega grzywnie, karze ograniczenia wolności albo pozbawienia wolności do lat 3. Tej samej karze podlega, kto rozpowszechnia bez podania nazwiska lub pseudonimu twórcy cudzy utwór w wersji oryginalnej albo w postaci opracowania, artystyczne wykonanie albo publicznie zniekształca taki utwór, artystyczne wykonanie, fonogram, wideogram lub nadanie.”,
- 2) odpowiedzialności dyscyplinarnej na podstawie art. 211 ust. 1 ustawy z dnia 27 lipca 2005 r. Prawo o szkolnictwie wyższym (t. j. Dz. U. z 2012 r. poz. 572, z późn. zm.) „Za naruszenie przepisów obowiązujących w uczelni oraz za czyny uchybiające godności studenta student ponosi odpowiedzialność dyscyplinarną przed komisją dyscyplinarną albo przed sądem koleżeńskim samorządu studenckiego, zwanym dalej sądem koleżeńskim”

Prawdziwość powyższego oświadczenia potwierdzam własnoręcznym podpisem.

Dragan Krzysztof
czytelny podpis studenta

*) niepotrzebne skreślić

Aplikacja internetowa do wyszukiwania połączeń lotniczych

Streszczenie

Celem niniejszej pracy było opracowanie aplikacji internetowej która pozwoliłaby na wyszukiwanie połączeń lotniczych korzystając z danych zawartych na stronach internetowych przewoźników bądź z innych centr danych. Aplikacja została podzielona na część kliencką oraz serwerową. Klient został napisany przy użyciu technologii Angular 6, natomiast część serwerowa w technologii Java 10. W pracy znajduje się opis architektury stworzonej aplikacji, modułu wyszukiwania połączeń lotniczych a także zagadnień teoretycznych związanych z projektowaniem interfejsu użytkownika dla przeglądarki internetowej.

Słowa kluczowe: Java, Angular 6, REST, programowanie obiektowe, protokół HTTP, programowanie funkcyjne

A web application to search for flight connections

Summary

The purpose of thesis was to build a web application, which will be able to search flight connections using data included on air websites or others data sources. Application was divided into two parts: client and server. Client was implemented using technology of Angular 6, whereas server in Java 10 technology. Description of architecture built application, module of air connections searching and theoretical issues related to building user interface for web application are included in this thesis.

Keywords: - Java, Angular 6, REST, Object Oriented Programming, HTTP Protocol, Functional Programming

Spis treści

1	Wstęp	10
2	Opis rozwiązywanego zagadnienia	12
2.1	Źródła danych o połączeniach lotniczych	12
2.2	Parsowanie różnych rodzajów danych	13
2.3	Wydajność wyszukiwania	14
3	Przegląd istniejących rozwiązań	15
3.1	Wyszukiwarka lotów Skyscanner	15
3.2	Wyszukiwarka lotów Google Flights	18
4	Projekt aplikacji	19

1 Wstęp

Celem pracy było stworzenie aplikacji internetowej umożliwiającej wyszukiwanie połączeń lotniczych. Do zrealizowania tego celu potrzebne było dogłębne zbadanie problemów związanych z tym zagadnieniem. Pierwszym krokiem była analiza dostępnych źródeł danych o połączeniach lotniczych. Nie było to zadanie łatwe, tylko niektórzy przewoźnicy udostępniają swoje dane lotnicze na zewnątrz. Oprócz stron przewoźników istnieją też specjalne centra chmurowe gromadzące informacje o połączeniach lotniczych, jednakże w przeważającej części firmy reprezentujące te centra pobierały opłaty za możliwość korzystania z ich serwisów. Do zrealizowania celu wymienionego w pierwszym zadaniu posłużyła usługa chmurowa firmy FlightLookup która udostępniała zarejestrowanemu użytkownikowi jej dane w limicie 500 żądań protokołu HTTP na miesiąc. Oprócz tego do zrealizowania pracy zgodnie z założonymi wymaganiami potrzebne były źródła udostępniające informacje o liniach lotniczych, wymiarach i wagi bagażu podróznego oraz cen przelotów. Kolejnym problem było poradzenie sobie z różnymi formatami danych które pozyskiwała część serwerowa. Deserializacja i serializacja danych odbywała się pomiędzy formatami: XML, JSON, HTML, CSV oraz obiektem języka Java. Dokładniejszy opis tego problemu znajduje się w rozdziale Implementacja. Ważną kwestią była też wydajność aplikacji, czas odpowiedzi powinien być dla użytkownika jak najmniejszy. Kod źródłowy modułu zbierającego dane był odpowiednio przygotowany do wykonania tego zadania z jak najlepszym efektem. Zadbano o wyeliminowanie fragmentów kodu które mogłyby opóźnić odpowiedź aplikacji. Ponadto w pracy został zaimplementowany mechanizm cachowania danych aby w przypadku gdy na przykład konkretny lot był wyszukiwany wiele razy, informacja o nim została zwrócona o wiele szybciej niż standardowo.

Aplikacja została podzielona na dwie części: część serwerową oraz kliencką. Pierwsza część została stworzona w języku programowania Java w wersji 10. Oprócz standardowego pakietu oprogramowania i narzędzi dostarczonego w JDK (Java Development Kit) zostały dodane biblioteki takie jak: Spring, Apache commons-io, Apache-csv czy też OkHttp. Biblioteka Spring jest odpowiedzialna w pracy za funkcjonowanie serwera udostępniającego dane oraz obsługę żądań sieciowych. Pozostałe biblioteki pomogły w procesie deserializacji i serializacji różnych formatów danych przychodzących do części serwerowej.

Oprócz samego rozwiązania zbierającego dane o połączeniach lotniczych potrzebny był internetowy interfejs użytkownika który umożliwiał by prezentowanie danych w przyjemny dla użytkownika sposób. Do tego zadania, a więc zadania stworzenia części klienckiej wybrano technologię Angular 6. Angular jest platformą opartą na języku TypeScript która razem w połączeniu ze stronami HTML oraz arkuszami stylów CSS pozwoliła na stworzenie responsywnego interfejsu, co na obecnym rynku internetowym jest dużą zaletą. Zadaniem części klienckiej było odebranie danych od części serwerowej a następnie pokazanie ich w przeglądarce internetowej. Zewnętrznymi bibliotekami w tej części były Angular Material oraz Bootstrap. Przyspieszyły one budowanie komponentów interfejsu. Kolejnym ważnym krokiem w pisaniu tej pracy było zaimplementowanie testów jednostkowych oraz integracyjnych które pozwoliłyby na automatyczne sprawdzenie aplikacji pod względem wymaganych funkcjonalności zawartych w założeniach. Dopiero poprawne zakończenie wszystkich testów pozwoliło uznanie części praktycznej pracy za skończoną.

2 Opis rozwiązywanego zagadnienia

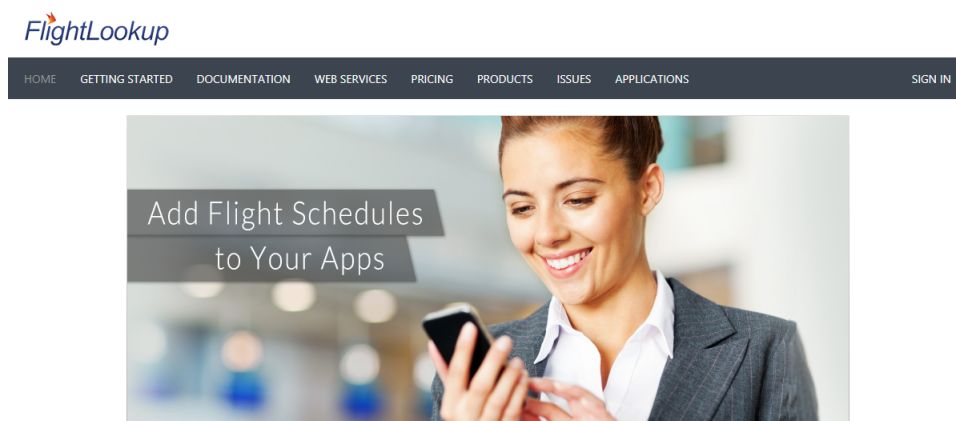
Głównym zagadnieniem podjętym w pracy było znalezienie sposobu na pozyskanie realnych danych o połączeniach lotniczych które można by było zaprezentować w kompleksowym interfejsie oraz we względnie optymalnym czasie dla użytkownika powstałej aplikacji. Zagadnienie to można podzielić na 3 części:

- Znalezienie źródeł danych o połączeniach lotniczych
- Parsowanie różnych rodzajów danych
- Zapewnienie dobrej wydajności podczas wyszukiwania połączeń lotniczych

Części te zostaną opisane w podrozdziałach bieżącego rozdziału.

2.1 Źródła danych o połączeniach lotniczych

Największą trudnością podczas pisania pracy było znalezienie odpowiednich zasobów danych który nie byłyby płatne oraz które zapewniałyby rzetelne i sprawdzone dane lotnicze. Poszukiwania zaczęto od złożenia podań do centr danych o dostęp do ich zasobów. Większość z nich wymagała opłaty za swoje usługi które sięgały nawet 10000\$. Niektóre z nich oferowały jednak darmowy dostęp do ich zasobów jednak był to dostęp limitowany. Na potrzeby pracy wybrano serwis FlightLookup jako głównego dostawcę danych, informacje przez niego dostarczone stanowiły lwią część odpowiedzi serwera. Darmowy dostęp jest limitowany 500 zapytaniami w trakcie miesiąca.



Rysunek 2.1: Portal serwisu FlightLookup

Dodatkowymi źródłami danych były:

- Skyscanner - serwis udostępniające średnie ceny przelotów w określonym przedziale czasowym oraz informacje dotyczące dozwolonego bagażu
- Aviation Edge - usługa chmurowa udostępniające dane o liniach lotniczych
- ourairport.com - strona internetowa umożliwiające pobranie danych o większości lotnisk na świecie

2.2 Parsowanie różnych rodzajów danych

Dane dostarczone przez zewnętrzne serwisy prezentowały swoją treść w różnych formatach. Aby zebrać pełną odpowiedź serwera należało w pierwszym kroku sparsować pojedyncze elementy a następnie zbudować z nich obiekt języka Java. Parsowanie zaczynało się od odebrania danych z serwisu FlightLookup w postaci XML (eXtensible Markup Language). Do zrealizowania tej czynności posłużono się zewnętrzną biblioteką jackson-dataformat-xml która efektywnie przełożyła obiekty XML na odpowiedni obiekty Javy.

```
1 <FlightDetails TotalFlightTime="PT3H35M"
2     TotalMiles="931"
3     TotalTripTime="PT4H25M"
4     FLSDepartureDateTime="2018-11-15T06:40:00"
5     FLSDepartureTimeOffset="+0100"
6     FLSDepartureCode="WAW"
7     FLSDepartureName="Warsaw"
8     FLSArrivalDateTime="2018-11-15T10:05:00"
9     FLSArrivalTimeOffset="+0000"
10    FLSArrivalCode="LHR"
11    FLSArrivalName="London Heathrow"
12    FLSFlightType="Connect"
13    FLSFlightLegs="2"
14    FLSFlightDays="...4..."
15    FLSDayIndicator=""
16 >
```

Listing 2.1: Przykładowe dane w XML

Informacje o liniach lotniczych dostarczane były w postaci JSON (JavaScript Object Notation). Do sparsowania ich posłużono się biblioteką Gson organizacji Google. Dzięki niej konwersja obiektu JSON do obiektu Java odbywała się w jednej linii kodu. W ten sam sposób odbywało się parsowanie danych zawierających ceny przelotów.

Ostatnim formatem danych który wymagał sparsowania był HTML (Hypertext Markup Language). Informacje przez niego opisane dotyczyły wymiarów oraz wagi bagażów dozwolonych na określonej linii lotniczej. Do wykonania tego zadania posłużyła zewnętrzna biblioteka Jsoup. Dedykowana dla języka Java, z gotowym pakietem oprogramowania bardzo dobrze poradziła sobie z parsowaniem treści HTML.

2.3 Wydajność wyszukiwania

Wyszukiwanie tak złożonych jak informacje o połączeniach lotniczych a następnie parsowanie ich niesie za sobą pewne konsekwencje. Są to konsekwencje czasownik, użytkownik powinien otrzymać interesującą go treść w czasie jak najkrótszym. W celu optymalizacji wydajności aplikacji wprowadzono mechanizmy skracające czas odpowiedzi części serwerowej. Dla zbierania danych dotyczących lotnisk oraz bagażów wprowadzono rozwiązania polegające na pobieraniu pełnych zasobów tych danych do bazy danych lub do pliku znajdującego się na serwerze. Pozwoliło to na pominięcie opóźnienia sieciowego związanego z potencjalną koniecznością pobierania tych danych ze stron lub zewnętrznych baz danych.

Kolejnym rozwiązaniem było wprowadzenia stylu programowania funkcyjnego w kluczowych elementach części serwerowej które odpowiadały za wyszukiwanie połączeń lotniczych. Programowanie funkcyjne wprowadzone w Javie 8 pozwala skrócić operacje po stronie wirtualnej maszyny Javy a więc też zaoszczędzić cenne milisekundy w trakcie wyszukiwania lotów. Ostatnim mechanizmem był moduł cachowania danych. Jego przeznaczeniem jest skracanie czasu odpowiedzi dla określonych zapytań do części serwerowej które są wielokrotnie i wywoływane przez wielu użytkowników. Wykonując operacje wyszukiwania lotów, moduł wyszukiwania sprawdza czy w cache'u znajdują się poszukiwane operacje. Jeśli tak zwraca je użytkownikowi, jeśli nie wyszukuje loty standardowym sposobem a wynik zapisuje do cache'u. Cache przechowuje w sobie do 100 zapytań.

3 Przegląd istniejących rozwiązań

Analiza istniejących rozwiązań aplikacji wyszukujących połączeń lotniczych pozwoliła nadać pracy bardziej precyzyjne wymagania oraz zaprojektować jej ogólny przebieg. W internecie można znaleźć wiele aplikacji o podobnych lub takich samych funkcjonalnościach jak tworzona praca. W tym rozdziale zostaną opisane najbardziej znane wyszukiwarki lotów dostępnych na rynku.

3.1 Wyszukiwarka lotów Skyscanner

Pierwszym przykładem została aplikacja internetowa Skyscanner. Jest to wyszukiwarka lotów, która umożliwia użytkownikom szukanie lotów według ceny i lokalizacji. Oprócz funkcji wyszukiwania lotów, Skyscanner oferuje opcje wyszukiwania hotelów blisko lotnisk oraz wypożyczenia auta w pobliżu lotniska docelowego. Aplikacja ta została stworzona oraz wdrożona w 2002 roku. Od tego czasu firma Ctrip która jest właścicielem tego produktu zatrudnia ponad 200 pracowników. Warto wspomnieć o jej innej usłudze która udostępnia dane o połączeniach lotniczym zewnętrznym firmom i deweloperom. Jej dane były brane pod uwagę w czasie szukania źródeł danych lecz Skyscanner wymaga dużych opłat za swoje usługi, w warunkach akademickich niemożliwe było z ich skorzystanie. Aplikacja ta jest dostępna w ponad 30 językach oraz używana przez 60 milionów użytkowników miesięcznie. Aplikacja wiele razy nagradzana była za swoją funkcjonalność i użyteczność użytkownikom. Skyscanner znajduje się pod adresem: <https://www.skyscanner.net/>



Rysunek 3.1: Logo aplikacji Skyscanner

Na głównej stronie wyszukiwarki Skyscanner na pierwszy rzut oka można zauważyć panel wyszukiwania lotów.

The screenshot shows the Skyscanner website's search interface. At the top, there's a navigation bar with the Skyscanner logo, a 'Help' link, a currency selector set to 'GBP', and a 'Log in' button. Below this is a category bar with 'Flights', 'Hotels', and 'Car Hire' buttons. The main search area features a background image of a person relaxing by a lake. It includes radio buttons for 'Return' (selected), 'One way', and 'Multi-city'. The search form has fields for 'From' (Krakow (KRK)), 'To' (Country, city or airport), 'Depart' (21/12/2018), 'Return' (28/12/2018), and 'Cabin Class & Travellers' (1 adult, Economy). There are checkboxes for 'Add nearby airports', 'Direct flights only', and a 'Map' link. A green 'Search flights' button is on the right.

Rysunek 3.2: Panel wyszukiwania aplikacji Skyscanner

W powyższym panelu można wybrać lotnisko wylotu, lotnisko docelowe, datę wylotu i powrotu w przypadku lotu z opcją powrotu, a także ilość osób. Oprócz tych funkcji można zaznaczyć opcję lotu tylko w jedną stronę oraz lotu bezpośredniego. Ciekawą funkcjonalnością jest możliwość zaplanowania podróży między wieloma miejscami. Pozwala on wybrać maksymalnie 6 lotów i zaplanować na ich podstawie przebieg podróży. Interfejs jest intuicyjny i zrozumiały dla użytkownika.

This screenshot shows the multi-city search feature in the Skyscanner interface. The 'Multi-city' radio button is selected. The form displays a sequence of flight legs: the first leg is from 'Krakow (KRK)' to 'Berlin Schoenefeld (SXF)' on '21/12/2018'. Below this, there are two more empty rows, each with 'From', 'To', and 'Depart' fields. A '+ Add another flight' button is located between the first and second empty rows. At the bottom left, the 'Cabin Class & Travellers' dropdown is set to '1 adult, Economy'. A green 'Search flights' button is at the bottom right.

Rysunek 3.3: Komponent podróży między wieloma miejscami

Po wybraniu parametrów wyszukiwania takich jak miejsce wylotu, miejsce docelowe czy data wylotu kolejną funkcjonalnością aplikacji będzie wyświetlenie wyników wyszukiwania.

Interfejs ukazujący rezultaty wyszukiwania został wyposażony w wiele opcji filtracji danych. Wyświetlone informacje uwzględniają daty rozpoczęcia oraz zakończenia przelotów, realne ceny biletów, nazwy linii lotniczych wraz z logami graficznymi.

The screenshot displays the Skyscanner website interface for a flight search from Krakow (KRK) to Berlin (SXF). The search parameters are set for 1 adult in Economy class, with the departure date set to Friday, 21st December. The results are sorted by 'Best' and show 15 results. On the left, there are filters for 'Stops' (Direct, 1 stop, 2+ stops), 'Departure times' (Outbound 00:00 - 23:59), 'Journey duration' (1.5 hours - 36.5 hours), and 'Airlines'. The main results area shows three flight options:

Best	Cheapest	Fastest
£9 1h 15	£9 1h 15	£9 1h 15

The first flight option is a direct flight by Ryanair, departing at 08:40 from KRK and arriving at 09:55 at SXF, priced at £9. The second flight option is a 1-stop flight by LOT + Wizz Air, departing at 07:55 from KRK, stopping at BUD, and arriving at 12:05 at SXF, priced at £81. The third flight option is a 1-stop flight by easyJet, departing at 08:35 from KRK, stopping at BSL, and arriving at 14:40 at SXF, priced at £138. Each flight option includes a 'Select' button and a note about 'Self-transfer' for the multi-stop flights.

Rysunek 3.4: Komponent podróży między wieloma miejscami

Powyższy przykład prezentuje wyniki dla przykładu połączenia w jedną stronę pomiędzy lotniskiem w Krakowie a lotniskiem Schönefeld w Berlinie. Interfejs pozwala skonfigurować wyniki pod względem liczby przesiadek w przelocie, długością lotów czy też godziną odlotu/przylotu. Istnieje też opcja włączenia powiadomień z aplikacji Skyscanner która powiadamia użytkownika mailowo o okazjach cenowych na wybranej trasie przelotu.

3.2 Wyszukiwarka lotów Google Flights

Drugim przykładem jest wyszukiwarka lotów stworzona przez światowego giganta Google. Jej nazwa to Google Flights. Korzystając z aplikacji takich jak Google Cloud, Google Maps czy opisywane Google Flights można zauważyć wspólny dla tych aplikacji styl wyglądu. Nosi on nazwę Material Design, gwarantuje on responsywność wyszukiwarki Google'a a także przyjazne użytkownikowi środowisko. Google Flights oferuje takie same funkcjonalności jak poprzednio opisywana wyszukiwarka a więc wyszukiwanie lotów według ceny i lokalizacji. Niemniej jednak aplikacja można poszczycić się niekonkurencyjną szybkością odpowiedzi gwarantowaną przez technologię specjalnego indexowania wymyśloną pierwotnie przy silniku Google Search. Wyszukiwarka wykonując operacje wyszukiwania połączeń lotniczych kalkuluje cenę przelotu każdego dnia w okresie następnych 12 miesięcy. Taka operacja pozwala później przedstawić użytkownikowi ceny lotów zwizualizowane w postaci grafu lub tabelki. Dzięki temu osoba wyszukująca loty może wybrać dla siebie najbardziej korzystną cenę. Aplikacja ta jest dostępna w ponad 30 językach, a jej użycie przekracza 100 milionów użytkowników miesięcznie. Została stworzona w roku 2010, a wdrożona w 2011. Warto zwrócić uwagę na jej integrację z pozostałymi produktami firmy Google. Google Flights potrafi pokazywać dostępne trasy pomiędzy lotniskami wykorzystując do tego celu Google Maps.

Aplikacja znajduje się pod adresem sieciowym: <https://www.google.com/flights>



Rysunek 3.5: Logo wyszukiwarki Google Flights

Główna strona Google flights prezentuje moduł wyszukiwania lotów wraz z najbardziej popularnymi kierunkami.



Rysunek 3.6: Logo wyszukiwarki Google Flights

4 Projekt aplikacji