

PROJEKT ZESPOŁOWY

Wykonanie:

German Andrejew
Przemysław Kowalczyk
Michał Krajnik
Anna Matyjewicz

STRATUS - webowa aplikacja pogodowa

1. Wprowadzenie do aplikacji

Aplikacja stworzona przez nasz zespół służy do sprawdzania lokalnej prognozy pogody. Umożliwia sprawdzenie informacji o pogodzie a także dokładną prognozę pogody na najbliższe 24h oraz kilka dni do przodu.

Aplikacja jest skierowana do osób, które codziennie rano przed wyjściem z domu sprawdzają aktualną pogodę i przygotowują się na różne atmosferyczne niespodzianki. A także do osób, które po prostu nie mają czasu dowiedzieć się o pogodzie w sposób tradycyjny, czyli za pomocą telewizyjnych prognoz pogodowych a także innych źródeł informacji.

Aplikacja znajdzie swoje zastosowanie także wśród ludzi, którzy dużo podróżują, dzięki funkcji zmiany globalizacji możliwe jest sprawdzenie pogody w miejscu, do którego właśnie udaje się dana osoba.

Dzięki tej aplikacji każdy użytkownik będzie w stanie odpowiednio się przygotować na warunki atmosferyczne!

2. Technologie użyte do stworzenia aplikacji

System - przeglądarka

Nasza aplikacja będzie działać na każdej nowoczesnej przeglądarce. Ponad to będzie ona responsywna co oznacza, że będzie można z niej korzystać także z urządzeń mobilnych.

HTML, CSS, JavaScript

Do zbudowania aplikacji użyjemy takich technologii jak:

HTML5 - język znaczników, który służy do tworzenia stron internetowych oraz do prezentowania ich zawartości użytkownikom. Używana przez nas wersja jest najnowsza w porównaniu do poprzedniej (HTML4) dodaje nowe elementy, usprawnia tworzenie serwisów i aplikacji internetowych a dodatkowo jest zaprojektowany tak aby starsze przeglądarki mogły zignorować nowe konstrukcje i nie powodować niepożądanych błędów.

CSS - czyli kaskadowe arkusze stylów, stosujemy je do opisywania wyświetlanych dokumentów napisanych w HTML. CSS opisuje w jaki sposób dokumenty mają wyglądać po wyświetleniu.

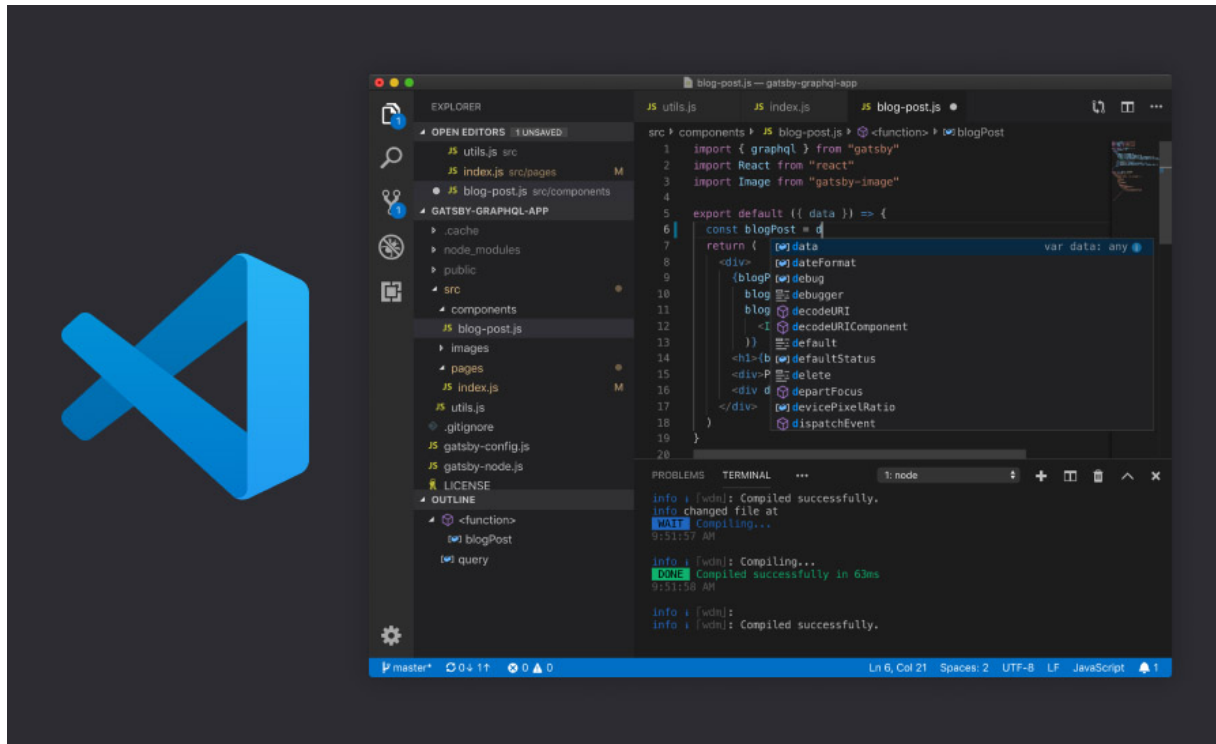
JavaScript - jest językiem programowania często stosowanym na stronach internetowych. Umożliwia reagowanie na różne zdarzenia, np. odświeżenie części strony po naciśnięciu przez użytkownika odpowiedniego przycisku na stronie.

Na naszej stronie użyjemy JavaScript do wysłania zapytania do API, przetworzenie otrzymanej informacji, a następnie wysłania zawartości do html aby została przedstawiona użytkownikowi

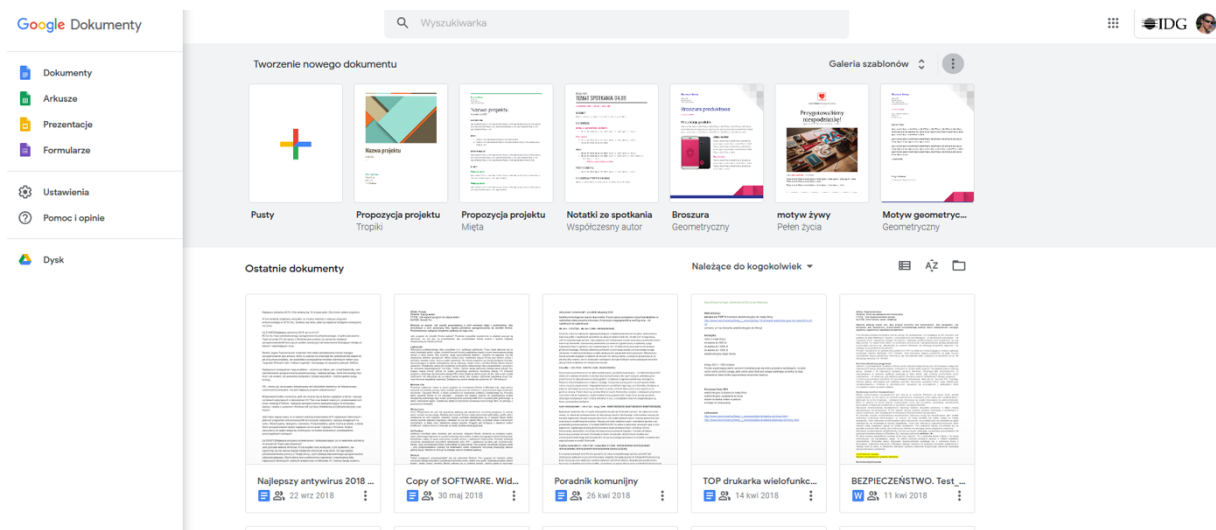
```
fetch(
  "https://api.openweathermap.org/data/2.5/weather?q=" +
    city +
    "&units=metric&appid=" +
    this.apiKey
)
.then((response) => {
  if (!response.ok) {
    alert("Nie znaleziono miasta/api key jest usuniety");
    throw new Error("Nie znaleziono.");
  }
  return response.json();
})
.then((data) => this.displayWeather(data));
},
displayWeather: function (data) {
  const { name } = data;
  const { temp, humidity } = data.main;
  const { speed } = data.wind;
  document.querySelector(".city").innerText = "Pogoda w mieście " + name;
  document.querySelector(".temp").innerText = temp + "°C";
  document.querySelector(".humidity").innerText =
    "Wilgotność: " + humidity + "%";
  document.querySelector(".wind").innerText =
    "Prędkość wiatru: " + speed + " km/h";
```

3. Narzędzia z których będziemy korzystać do stworzenia aplikacji

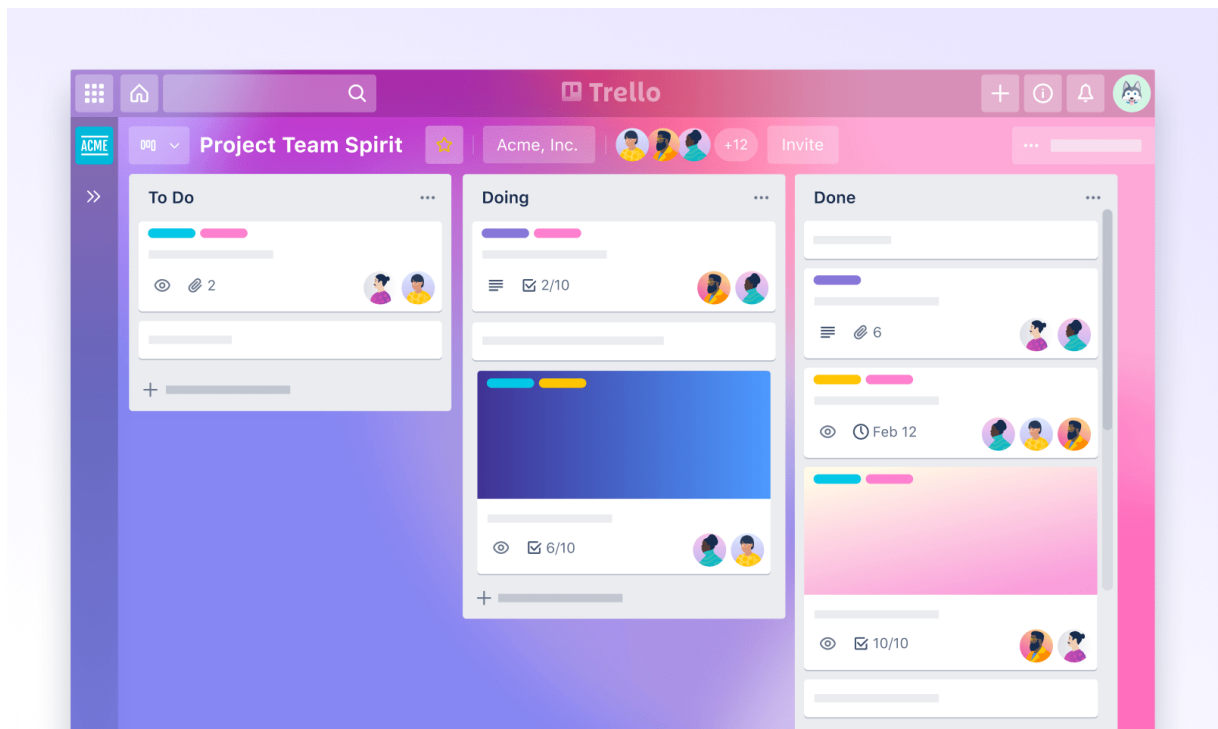
Visual Studio Code - do napisania tej aplikacji webowej użyjemy Visual Studio Code jest to darmowy edytor kodu, zoptymalizowany pod kątem budowania i usuwania błędów w nowoczesnych aplikacjach internetowych i chmurowych. Został stworzony przez firmę Microsoft.



Dysk Google - jest to narzędzie pozwalające dzielić się między innymi członkami zespołu materiałami niezbędnymi do stworzenia aplikacji a także dokumentami, gdzie każdy z nas na bieżąco będzie zapisywał swoje uwagi oraz opisywał stworzony przez niego fragment aplikacji



Trello - jest to darmowy program do zarządzania projektem a także dzieleniem się i informowaniem na bieżąco postępem tasków (zadań) przydzielonych do każdego z nas. Większość dzisiejszych firm korzysta z podobnego oprogramowania co znacznie zwiększa efektywność pracy.



4. Hosting + domena aplikacji webowej

Aby nasza aplikacja stała się dostępna dla innych użytkowników niezbędne jest wybranie przez nas domeny także usług firmy hostingowej

Postanowiliśmy skorzystać z usług firmy Hosting House, gdzie po krótkim reasearchu znaleźliśmy satysfakcjonującą nas domenę.

Nasz wybór to: **stratus.net.pl**

Gratulacje, stratus.net.pl jest dostępna!

Zarejestruj ją lub [Wypróbuj inną domenę](#)

[Kliknij aby kontynuować >>](#)

Hosting naszej aplikacji webowej na rok wyniesie nas **36.90 zł** co naszym zdaniem jest zadowalającym wynikiem.

Opis	Cena
Hosting www - Pakiet Srebrny (stratus.net.pl) [Usur]	30.00zł
Rejestracja domeny - stratus.net.pl - 1 Rok/Lat [Usur]	15.00zł
Suma:	45.00zł
Rabat:	15.00zł
VAT (23.00%):	6.90zł
Do zapłaty:	36.90zł

5. API - OpenWeatherMap

W aplikacji wykorzystano OpenWeatherMap API. Prognoza pogody i niezbędne informacje są pobierane z OpenWeatherMap a następnie wyświetlane w naszej aplikacji.

Aby skorzystać z tego API potrzebujemy API key a co za tym idzie darmowe konto na stronie openweathermap.org, przy wybraniu darmowego pakietu mamy do dyspozycji 60 zapytań na minutę oraz aż 1,000,000 zapytań na miesiąc lecz wraz ze wzrostem zapotrzebowania mamy możliwość wybrać jeden z płatnych pakietów

Free	Startup	Developer	Professional	Enterprise
Get API key	30 GBP / month	140 GBP / month	370 GBP / month	from 1500 GBP / month
60 calls/minute 1,000,000 calls/month	600 calls/minute 10,000,000 calls/month	3,000 calls/minute 100,000,000 calls/month	30,000 calls/minute 1,000,000,000 calls/month	200,000 calls/minute 5,000,000,000 calls/month
Current Weather Minute Forecast 1 hour* Hourly Forecast 2 days* Daily Forecast 7 days* National Weather Alerts* Historical weather 5 days* Climatic Forecast 30 days Bulk Download	Current Weather Minute Forecast 1 hour** Hourly Forecast 2 days** Daily Forecast 16 days National Weather Alerts** Historical weather 5 days** Climatic Forecast 30 days Bulk Download	Current Weather Minute Forecast 1 hour Hourly Forecast 4 days Daily Forecast 16 days National Weather Alerts Historical weather 5 days Climatic Forecast 30 days Bulk Download	Current Weather Minute Forecast 1 hour Hourly Forecast 4 days Daily Forecast 16 days National Weather Alerts Historical weather 5 days Climatic Forecast 30 days Bulk Download	Current Weather Minute forecast 1 hour Hourly Forecast 4 days Daily Forecast 16 days National Weather Alerts Historical weather 5 days Climatic Forecast 30 days Bulk Download
Basic weather maps Historical maps Global Precipitation Map - Historical data	Basic weather maps Historical maps Global Precipitation Map - Historical data	Advanced weather maps Historical maps Global Precipitation Map - Historical data	Advanced weather maps Historical maps Global Precipitation Map - Historical data	Advanced weather maps Historical maps Global Precipitation Map - Historical data

W celu kontaktu z api wykorzystamy metodę fetch jest to metoda która przyjmuje jeden obowiązkowy argument w naszym przypadku jest to zapytanie o miasto wybrane przez użytkownika wysłane do openweathermap wraz z naszym API KEY, otrzymujemy dzięki temu Promise który konsumujemy metodą then(). Dzięki temu możemy zbindować nasze dane do poszczególnych elementów w naszej funkcji.