

Geolokalizacja strony www

Położenie: (nie dotyczy).

© Copyright by 3bird Projects 2020, <http://edukacja.3bird.pl>

Opis

Do ustalenia pozycji przeglądarki internetowej, trzeba posiadać potężną bazę danych na temat lokalizacji punktów dostępowych na całym świecie. Nie dziwi więc, że taki mechanizm posiadają tylko duże firmy... a zwykli programiści mogą co najwyżej zaimplementować ich rozwiązania.

Dawniej, firma *Google* uzyskiwała informacje na temat położenia poszczególnych punktów Wi-Fi za pomocą swoich pojazdów "StreetView". Od pewnego czasu jednak firma gromadzi takie dane za pomocą smartfonów samych użytkowników (SSID + MAC).

Generalnie, pozycja może być ustalana poprzez: GPS, Wi-Fi, IP, siłę sygnału GSM. Każda z tych metod ma swoją precyzję (dokładność). Na precyzję wpływa nawet rodzaj przeglądarki w smartfonie: mogą się one odwoływać do różnych dostawców lokalizacji (polecam Firefoksa).

HTML5 Geolocationa API

Organizacja W3C opracowała specyfikację, którą powinni wykorzystywać webmasterzy przy tworzeniu stron internetowych: <https://www.w3.org/TR/geolocation-API/>

Powyższą specyfikację implementują w swoich mechanizmach twórcy przeglądarek internetowych. Większość z nich wprowadza odwołania do baz danych *Google* (np. *Firefox*, choć nie we wszystkich przypadkach).

Należy także pamiętać, że strona internetowa, na której odbywa się lokalizacja, musi być ładowana za pomocą protokołu **https**.

Google Maps API

Aby wykorzystywać lokalizację *Google* i mapy *Google* na swoich stronach (czyli *de facto* w swoich aplikacjach mobilnych), należy:

- założyć projekt na stronie <https://console.developers.google.com/>;
- założyć konto billingowe (płatnicze / rozliczeniowe), gdyż korzystanie z tego API jest płatne; firma *Google* w promocji daje \$300, więc przez rok możesz używać za darmo; wysokość opłat uzależniona jest od ilości wyświetleń twojej strony;
- wygenerować unikalny klucz (API key) i ograniczyć jego użycie do konkretnych sytuacji / aplikacji; poniżej, przykład użycia klucza (DOM + JS):

```
var mapa = new Image();
```

```
mapa.src = "https://maps.googleapis.com/maps/api/js?center=" + szerokoscGeograficzna + "," + dlugoscGeograficzna + "&size=500x400&key=AlzaSytCjftkzINI2whcN";
```

lub

```
<script async defer src="https://maps.googleapis.com/maps/api/js?key=AlzaSytCjftkzINI2whcN&callback=initMap"></script>
```

Uwaga: Korzystanie z rozwiązań *Google*, nie jest konieczne. Zamiast tego można wykorzystać rozwiązania zaimplementowane w HTML5 oraz mapy *OpenStreetMap (OSM)*.

Przykłady kodu (HTML5+JS)

Sprawdzenie, czy przeglądarka obsługuje funkcję geolokalizacji:

```
if(navigator.geolocation) {  
    jakiś komunikat...  
}  
else {  
    jakiś komunikat...  
}
```

Funkcja wypisująca pozycję (czas podany jest w milisekundach):

navigator.geolocation.getCurrentPosition(mojaLokalizacja, obsługaBłędów, opcje);

Uwaga: Jako opcję dobrze jest dodać `{timeout: 15000}`.

Funkcja wypisująca pozycję w sposób ciągły:

navigator.geolocation.watchPosition(mojaLokalizacja, obsługaBłędów, opcje);

Uwaga: Jako opcję należy dodać `{maximumAge:20000}`, co spowoduje odświeżanie pozycji co 20 sekund.

Aby zakończyć śledzenie bieżącej pozycji:

```
var watchId = navigator.geolocation.watchPosition(mojaLokalizacja, obsługaBłędów);  
navigator.geolocation.clearWatch(watchId);
```

Przykład kompletnego kodu:

```
var opcje = {  
    enableHighAccuracy: true, (używa możliwie najbardziej dokładnej metody)  
    timeout: 15000, (czas na pobranie pozycji to 15 sekund)  
    maximumAge: 0 (co jaki czas ma odświeżać pozycję; 0 oznacza „za każdym razem”, zaś „Infinity” - że nigdy ponownie)  
};
```

```
function wypiszDane(pozycja) {  
    dane = "Pozycja: " + pozycja.coords.latitude + "," + pozycja.coords.longitude ". Dokładność:  
    " + pozycja.coords.accuracy + " metrów."  
    miejsceNaWynik = document.getElementById("jakiśID");  
    miejsceNaWynik.appendChild(dane);  
}
```

```
function obsługaBledow(blad) {  
    switch(blad.code) {  
        case blad.PERMISSION_DENIED:  
            alert("Użytkownik ma zabroniony dostęp do lokalizacji.");  
            break;  
        case blad.POSITION_UNAVAILABLE:  
            alert("Dane o pozycji są niedostępne.");  
            break;  
        case blad.TIMEOUT:  
            alert("Zbyt długi czas oczekiwania na dane.");  
            break;
```

```

    case blad.UNKNOWN_ERROR:
        alert("Wystąpił nieznany błąd.");
        break;
    }
}

```

navigator.geolocation.getCurrentPosition(wypiszDane, obslugaBledow, opcje);

Serwisy

Poniższe serwisy umożliwiają dostarczenie danych na temat lokalizacji bazując na bazie IP (są w większości przypadków płatne):

Serwis ipInfoDB.com

Korzystanie z tego API wymaga rejestracji w serwisie.

<http://api.ipinfodb.com/v3/ip-city/?key=<zarejestrowane-Haslo>&ip=74.125.45.100>

Serwis ip-api

Zwraca dane na temat miasta i ISP w formie tekstowego json-a.

<http://ip-api.com/json>

Serwis ip2location

<http://www.ip2location.com/free/widgets>

Serwis ipinfo.io

```
<?php
```

```
$mojRegion = file_get_contents("http://ipinfo.io/{$_SERVER['REMOTE_ADDR']}/region");
```

```
if (trim($mojRegion) != "Slaskie") {
```

```
    print("Nie jesteś na Śląsku");
```

```
    exit();
```

```
}
```

```
else {
```

```
    print("Jesteś na Śląsku!");
```

```
}
```

```
?>
```

Problemy

W niektórych przypadkach (przeglądarka *Firefox* w systemie *Linux*), ma problem z uzyskaniem współrzędnych pomimo prawidłowej konfiguracji i dobrze napisanego kodu. Problemem są niewłaściwe dane providera dostarczającego danych.

Rozwiązanie:

1. W przeglądarce wpisać adres „*about:config*”.
2. Odnaleźć pozycję „*geo.provider.network.url*”.
3. Wpisać wartość:

https://location.services.mozilla.com/v1/geolocate?key=%MOZILLA_API_KEY%

(stara wartość to: https://www.googleapis.com/geolocation/v1/geolocate?key=%GOOGLE_LOCATION_SERVICE_API_KEY%)

Ostatnia aktualizacja: 8 maja 2020.