

# Korišćene tehnologije, alati i biblioteke

\*\* Fajl je predviđen za prikaz na <https://github.com/djurdjevicfilip/Parkiraj-Beograd>

Standardni jezici koji su korišćeni :

- PHP (4.9.2)
- JavaScript
- HTML, CSS
- MySQL

## Tehnologije i biblioteke koje su korišćene

Tehnologija/Bibl.	Opis	Verzija
<a href="#">Laravel</a>	PHP Framework za veb aplikacije, sa ekspresivnom i elegantnom sintaksom	7.x
<a href="#">polyline.js</a>	Biblioteka koja se koristi za dekodiranje polyline objekta dobijenog od Google Maps Directions API. Uz pomoć ove biblioteke možemo da dobijemo niz pozicija koje predstavljaju rutu, a samim tim i da simuliramo kretanje korisnika	
<a href="#">alertify.js</a>	Biblioteka koja se koristi za prikazivanje obaveštenja korisniku	
<a href="#">Pusher API</a>	API koji se koristi za slanje obaveštavanje korisnika. Korisnik se pretplaćuje na kanal, i bilo koja promena senzora (dodavanje, brisanje, senzor postaje zauzet...) na serveru se manifestuje kod klijenata	
<a href="#">Google Maps JavaScript API</a>	Google-ov API za dinamičko prikazivanje mape. On nam služi za postavljanje markera na mapu, kao i za većinu onoga što se vidi na mapi.	3.0
<a href="#">Google Maps Directions API</a>	Google-ov API za prikazivanje rute do mesta. Ovaj API takodje koristimo za uzimanje polyline objekta koji se dalje šalje u polyline.js	3.0
<a href="#">Google Maps Places API</a>	Google-ov API za pretragu i prikazivanje relevantnih mesta. Ovaj API nam omogućava <i>location biasing</i>	3.0
<a href="#">Google Maps Distance Matrix API</a>	Google-ov API za pronalazenje najbliže lokacije. (prethodno korišćen, više detalja prikazano je u nastavku ovog dokumenta, kao i u readme.md fajlu na našem <a href="#">Git Repozitorijumu</a> )	3.0
<a href="#">Apache</a>	Besplatan open-source HTTP server	2.4.41
<a href="#">Ajax</a>	Ajax koristimo za slanje poruka serveru od strane klijenta, o dolasku na određenu lokaciju	

## Ostalo

Stavka	Opis
<a href="#">GitHub</a>	Platforma za kolaboraciju
<a href="#">GitHub Pages</a>	Besplatno host-ovanje stranica, korišćeno za prototip
<a href="#">MS Teams</a>	Platforma korišćena za komunikaciju između članova tima
<a href="#">VS Code</a>	Vrlo mocan Code Editor, sa odličnim izborom ekstenzija
<a href="#">VS Code Extensions</a>	Prettier, Bracket Pair Colorizer, Better Comments
<a href="#">Toad Data Modeler</a>	Alat za modelovanje baze podataka
<a href="#">StarUML</a>	Alat za UML modelovanje

## OSRM API vs. Google Maps API

- **Google Maps Api** korišćen je za prikazivanje mape, rutiranje, clustering...

- **OSRM Api** je korišćen za pronalazak mesta do kog se najbrže dolazi

#### ***Zašto OSRM?***

Google nam uz svoju uslugu **Distance Matrix** omogućava pronalazak najbližeg mesta, ali je broj mesta ograničen, a broj poziva API-a drastično raste sa porastom broja mesta. Na primer, ukoliko bismo imali **200** parking mesta, pronalazak najbližeg mesta bi trajao više od 2 minuta, a broj poziva bi mogao da bude među desetinama hiljada. To je, dakle, samo za jedan korisnički zahtev! OSRM nam sve ovo rešava jer je besplatan i bez ograničenja. Jedina 'mana' je što OSRM trenutno ne podržava **Distance Matrix**, već samo **Duration Matrix**.

## Broadcasting & Observing

---

Korisnička stranica sadrži mapu koju je često potrebno osvežavati i menjati. Trudili smo se da poboljšamo performanse stranice i da je što manje preopteretimo. Iz tog razloga umesto Ajax-a korišćene su Laravel-ove dodatne mogućnosti. **Observing** nam služi za praćenje promena u bazi, a **Broadcasting** i **Pusher API** nam služe za slanje poruka korisničkim stranicama. Ovim dobijamo bolje performanse korisničke stranice, jer se stranica ne osvežava na fiksni vremenski period. Takođe, ovo nam omogućava i trenutno vidljive promene, dok bi kod većeg fiksnog intervala bilo kašnjenja.