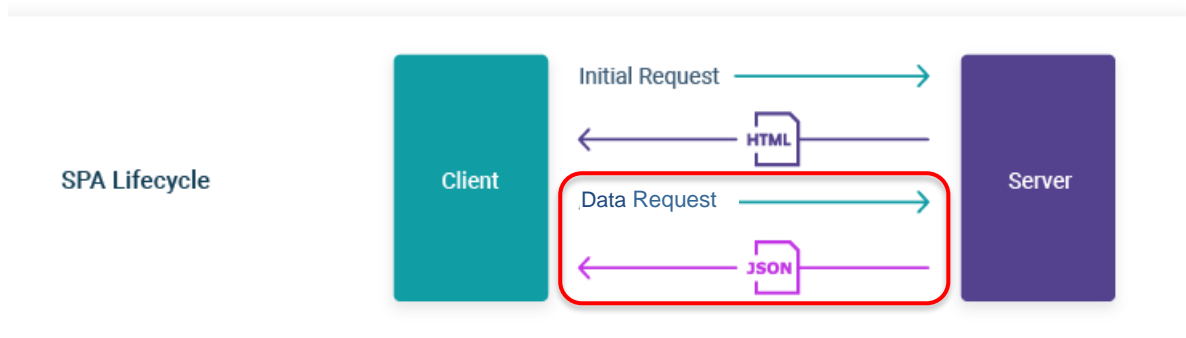


Http komunikacija

1. Uvod

Ako se podsetimo slike sa prvih vežbi, možemo videti da klijent komunicira sa serverom, a za komunikaciju između klijenta i servera se koristi uobičajeni komunikacioni protokol – **HTTP protokol**.



Kada front-end segment aplikacije koristi podatke koji se nalaze na back-end segmentu, onda se umesto HTML stranica šalju konkretni podaci. Format u kom back-end obično pruža podatke je **JSON** format.

2. JSON format

JSON (JavaScript Object Notation) je jezik za predstavljanje strukturiranih podataka korišćenjem notacije koja je čitljiva za čoveka (za JavaScript je to prirodan format podataka).

Back-end aplikacije najčešće pruža podatke u JSON formatu. Podaci o nekom objektu se u JSON formatu predstavljaju unutar vitičastih zagrada { }, u obliku parova **ime:vrednost**. Nizovi se predstavljaju pomoću uglastih zagrada [].

```
[
  {
    'ime': 'Pera',
    'prezime': 'Peric',
    'address': 'Trg Dositeja Obradovica 55',
    'city': 'Novi Sad'
  },
  {
    'ime': 'Mika',
    'prezime': 'Mikic',
    'address': 'Maksima Gorkog 5',
    'city': 'Novi Sad'
  }
]
```

Ovo je JSON **niz** koji sadrži **2 objekta**.

Svaki objekat je opisan nekim **podacima** (ime, prezime, adresa, mesto).

3. Servis i HTTP protokol

Kako bi Angular aplikacija (klijent) mogla da komunicira sa serverom (recimo Django aplikacija) preko HTTP protokola, Angular nudi već implementiranu podršku za ovaj način komunikacije.

Kako bi Angular aplikacija mogla da generiše zahtev (HTTP request) i pošalje ga na određenu adresu (URL adresu) i zatim preuzme odgovor (HTTP response), potrebno je koristiti modul `HttpClientModule` u kom se nalazi `HttpClient`. Ova klasa sadrži sve što je potrebno za komunikaciju putem HTTP protokola.

3.1 `get()` i `subscribe()`

Za slanje HTTP zahteva, nad referencom `HttpClient`-a, pozivamo metodu `get('http://...')` pri čemu kao parametar navodimo URL na koji šaljemo zahtev.

! Preuzimanje podataka putem HTTP protokola se odvija **asinhrono** tj. naredba za preuzimanje neće 'dočekati' da podaci stignu, već program nastavlja se izvršavanjem ostalih naredbi.

Da bi aplikacija obradila tražene podatke po njihovom prispeću, implementiramo blok naredbi koji će izvršiti prihvrat ovih podataka. To se postiže tako što funkciju za obradu odgovora prijavimo tj. pretplatimo (eng. `subscribe`) na odgovor.

U pozivu metode `subscribe` kao parametar navodimo naredbe kojima će dobijeni HTTP odgovor biti transformisan u oblik pogodan za našu aplikaciju. Za ovaj vid transformacije podataka su korisne tzv. *arrow* funkcije.

3.2 Arrow funkcije

Arrow funkcije predstavljaju skraćeni način zapisivanja funkcija. Sintaksa arrow funkcija:

```
(arg1, arg2, arg3,...) => izraz
```

Kada je moguće, uz argumente je korisno navesti tipove:

```
(x: number, y: number): number => x + y
```

Telo funkcije se može implementirati i blokom naredbi:

```
(x: number, y: number): number => {  
    return x + y;  
}
```

3.2 `map()`

Pristigli podaci ne moraju biti modelovani na identičan način kao podaci u modelu naše Angular aplikacije. Zbog toga je potrebno izvršiti mapiranje dobijenih podataka na naš model.

Angular poseduje ugrađenu podršku za preslikavanje cele liste podataka iz jednog modela u drugi tako što implementiramo preslikavanje jednog objekta u drugi, a Angular će to pravilo primeniti na celu listu. Za ovo preslikavanje se koristi funkcija `map`.

Funkciju `map` ćemo pozvati nad pristiglim podacima tako što ćemo joj kao **parametar** dati **arrow funkciju** kojom se podaci jednog pristiglog objekta koriste za instanciranje jednog objekta iz našeg modela.

Zadaci

Alat koji će se koristiti za pravljenje Angular aplikacija na vežbama je <https://stackblitz.com/>

1. Putem servisa omogućiti preuzimanje podataka o pesmama sa postojeće Django aplikacije, pa pomoću komponente i šablona prikazati ove podatke.

Domaći

1. Putem servisa omogućiti preuzimanje podataka o filmovima sa postojeće Django aplikacije, pa pomoću komponente i šablona prikazati ove podatke.

Literatura

1. <https://angular.io/>
2. <https://material.angular.io/>
3. Materijali sa predavanja