

Komunikacija komponenti

1. Uvod

Do sada smo u razvoju Angular aplikacija koristili jednu komponentu. Međutim, za web stranice na kojima se prikazuju složenije strukture podataka je potrebno koristiti više komponenti. Pored toga, potrebno je uspostaviti i veze među komponentama, kako bi podaci mogli da se razmenjuju.

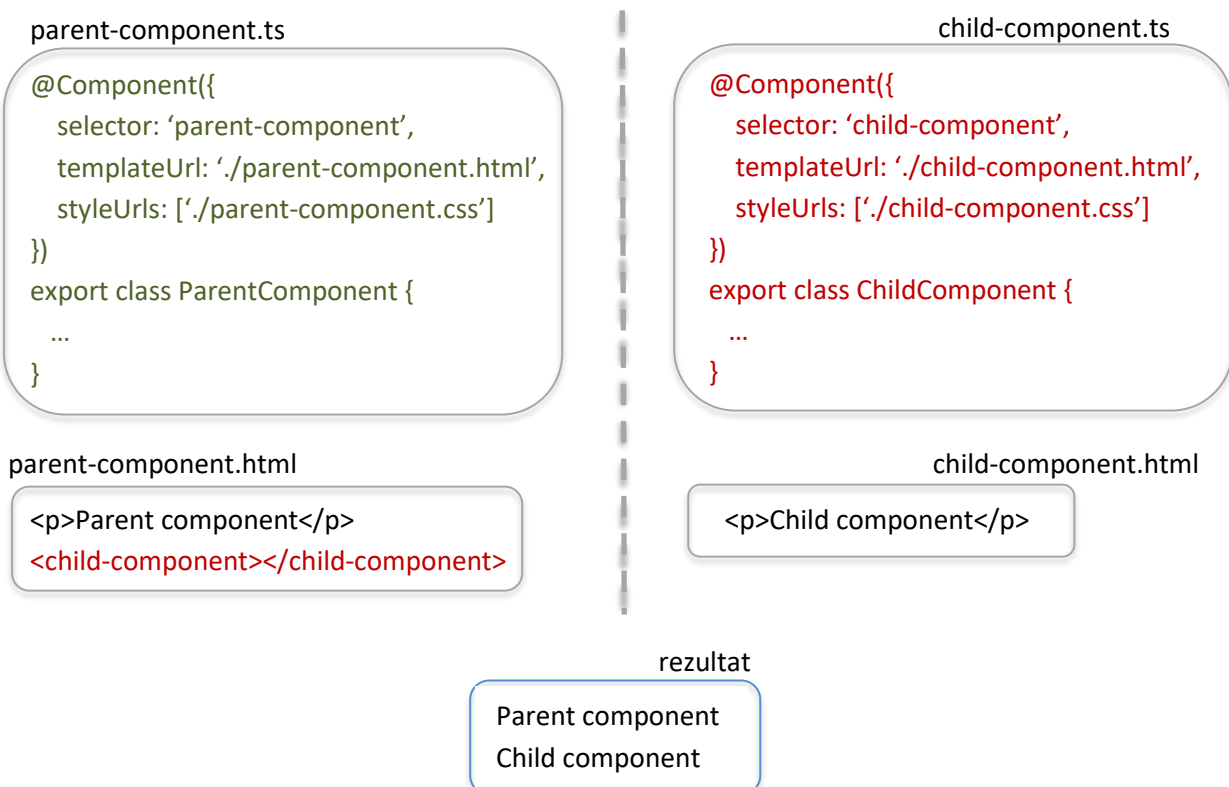
2. Parent-child odnos

Angular komponente dozvoljavaju da se deo njihovog prikaza delegira tj. prepusti drugoj komponenti. Kada jedna komponenta u svom prikazu ima ugrađen prikaz neke druge komponente, onda su te dve komponente u **parent-child** (roditelj-dete) odnosu.

Angular omogućava i prenošenje vrednosti između ovih komponenti radi usklađivanja prikaza.

2.1 Povezivanje komponenti u parent-child odnosu

Child komponenta se ugrađuje u **parent** komponentu tako što se u šablon **parent** komponente ugradi element **child** komponente (tzv. direktiva child komponente).



2.2 Ulazno svojstvo komponente

Nakon ugrađivanja jedne komponente u drugu, potrebno je obezbediti usklađenost između njihovih podataka. U te svrhe, u child komponentu ugrađujemo **ulazno svojstvo**.

Ulazno svojstvo je **promenljiva** ispred koje stavljamo dekorator **Input()**. Za korišćenje ovog dekoratora je potrebno dodati **import deklaraciju**.

```
import { Input } from '@angular/core';
```

```
@Component({  
  selector: 'child-component',  
  templateUrl: './child-component.html',  
  styleUrls: ['./child-component.css']  
})  
export class ChildComponent {  
  @Input()  
  count: number;  
}
```

child-component.ts

child-component.html

```
{{ count }}
```

Kada komponenta ima ulazno svojstvo, ona putem tog svojstva može primati vrednosti. Dodela vrednosti nekom ulaznom svojstvu komponente je moguća u okviru elementa kojim je ta komponenta ugrađena u web stranicu (u direktivi te komponente).

parent-component.html

```
<p>Parent component</p>  
<child-component count="5"></child-component>
```

2.2 Prenošnje vrednosti između komponenti

U šablonu parent komponente možemo primeniti povezivanje svojstava (property binding) kako bi određenu promenljivu parent komponente povezali sa ulaznim svojstvom child komponente.

```
@Component({  
  selector: 'parent-component',  
  templateUrl: './parent-component.html',  
  styleUrls: ['./parent-component.css']  
})  
export class ParentComponent {  
  x: number = 5;  
}
```

parent-component.ts

Sada je promenljiva **count** (iz child komponente) povezana sa promenljivom **x** (iz parent komponente).

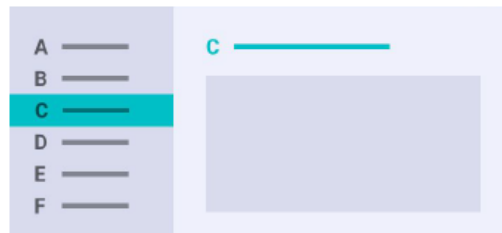
parent-component.html

```
<p>Parent component</p>  
<child-component [count]="x"></child-component>
```

3. Master view – Detail view

Prikaz neke kolekcije podataka obično podrazumeva višestruki prikaz osnovnih podataka o svakom zapisu. Zbog obimnosti podataka i istovremene težnje za preglednošću, prikaz podataka se najčešće organizuje iz dva dela:

- Master view: prikaz svih postojećih zapisa sa osnovnim informacijama
- Detail view: prikaz detalja gde će se navesti svi podaci o jednom zapisu



Master-detail view se može postići pomoću dve komponente u parent-child odnosu, gde je parent komponenta zadužena za master view, a child komponenta za detail view.

3.1 Obrada izbora korisnika

Komponenta koja prikazuje master view treba da omogući korisniku izbor neke od prikazanih stavki. Za ovo je potrebno upotrebiti **obradu događaja** (click) tako što posebnoj metodi prosleđujemo objekat koji predstavlja stavku koju je korisnik izabrao.

Komponenta koja prikazuje detail view inicijalno (pre nego što korisnik izabere neku stavku) ne treba da vrši prikaz. Da li je neka stavka izabrana možemo proveriti pomoću direktive `*ngIf` – ako postoji izabran objekat onda se vrši i prikaz njegovih detalja.

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 lokalne Django backend aplikacije. Koristeći dve komponente omogućiti master-detail prikaz ovih podataka.

Domaći

1. Putem servisa omogućiti preuzimanje podataka o filmovima sa lokalne Django backend aplikacije. Koristeći dve komponente omogućiti master-detail prikaz ovih podataka.

Literatura

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