
Bootstrap

Autori:
Milan Segedinac
Goran Savić

1. Bootstrap

Bootstrap je besplatan front-end radni okvir koji omogućuje brzo i jednostavno uređenje izgleda veb aplikacija. Uključuje HTML i CSS templejte za tipografiju, forme, dugmiće, tabele, navigaciju, modalne dijaloge, prikaze slika i još mnogo toga. Pored toga uključuje i opcione JavaScript dodatke. Važno je napomenuti da Bootstrap značajno pojednostavljuje pravljenje stranica responsivnog dizajna.

Korišćenje Bootstrap radnog okvira u Angular aplikaciji zahteva instaliranje Bootstrap paketa i dodavanje Bootstrapa u aplikaciju. Bootstrap paket se instalira pomoću npm:

```
$ npm install --save bootstrap
```

Nakon uspešne instalacije paketa, potrebno je dodati ga u Angular aplikaciju. Za to je potrebno u angular-cli.json dodati stil "../node_modules/bootstrap/dist/css/bootstrap.min.css". Element styles nakon toga treba da izgleda kao

```
"styles": [
  "../node_modules/bootstrap/dist/css/bootstrap.min.css",
  "styles.css"
]
```

Bootstrap grid sistem

Kostur bootstrap stranice je grid sistem koji se sastoji od 12 kolona u koju se raspoređuju elementi HTML stranice. Struktura grida data je tabelom ispod.

span 3	span 4	span 4	span 1
span 6		span 6	
span 12			

Grid sistem nam omogućuje da HTML elemente organizujemo u redove i za svaki element u redu definišemo koliko će kolona zauzeti. Red se definiše klasom row, a kolone se definišu klasama col-*-*. Širina kolone zavisiće od uređaja na kom se stranica prikazuje, tako da Bootstrap definiše četiri grupe col klasa:

Klasa	Uređaj
col-xs-*	Telefoni, širina manja od 768px
col-sm-*	Tableti, širina veća ili jednaka 768px
col-md-*	Desktop, širina veća ili jednaka 992px
col-lg-*	Veliki desktop, širina veća ili jednaka 1200px

Primer definisanja strukture stranice dat je listingom ispod

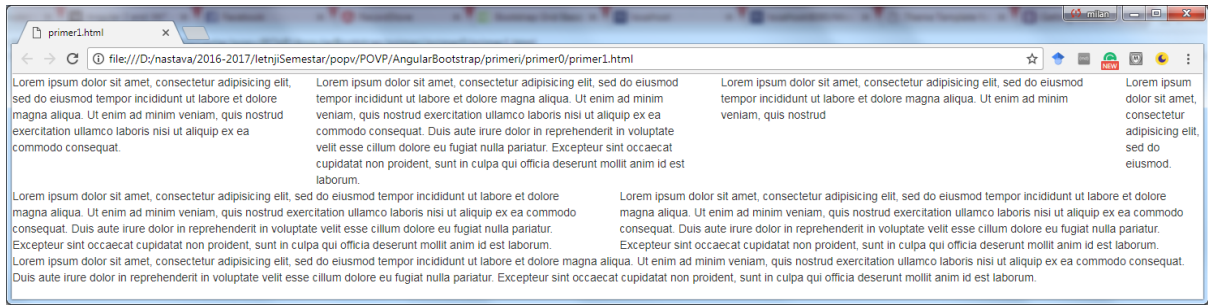
Angular Komponente

```
<html>

<head>
  <title></title>
  <link rel="stylesheet" type="text/css" href="bootstrap/dist/css/bootstrap.css">
</head>
<body>
  <div class="row">
    <div class="col-md-3">Lorem ipsum dolor sit amet, consectetur
adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna
aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi
ut aliquip ex ea commodo consequat.</div>
    <div class="col-md-4">Lorem ipsum dolor sit amet, consectetur
adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna
aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi
ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in
voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint
occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim
id est laborum.</div>
    <div class="col-md-4">Lorem ipsum dolor sit amet, consectetur
adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna
aliqua. Ut enim ad minim veniam, quis nostrud </div>
    <div class="col-md-1">Lorem ipsum dolor sit amet, consectetur
adipisicing elit, sed do eiusmod.</div>
  </div>
  <div class="row">
    <div class="col-md-6"> Lorem ipsum dolor sit amet, consectetur
adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna
aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi
ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in
voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint
occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim
id est laborum. </div>
    <div class="col-md-6">Lorem ipsum dolor sit amet, consectetur
adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna
aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi
ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in
voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint
occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim
id est laborum.</div>
  </div>
  <div class="row">
    <div class="col-md-12"> Lorem ipsum dolor sit amet, consectetur
adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna
aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi
ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in
voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint
occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim
id est laborum. </div>
  </div>
</body>
</html>
```

Navedenim HTML kodom dobijena je stranica sa slike ispod.

Angular Komponente



Pored mogućnosti organizacije strukture stranice, Bootstrap donosi i brojne komponente i u nastavku će biti dat pregled najvažnijih.

Tipografija i tekst u Bootstrapu

Osnovni font u Bootstrap stranicama je veličine 14px i primenjuje se na <body> element sa svim <p> podelementima, naravno, osim ako se postavi drugačiji stil.

Bootstrap postavlja izgled HTML zaglavlja, kao što je dato listingom i slikom ispod

```
<h1>h1. Bootstrap heading</h1>
<h2>h2. Bootstrap heading</h2>
<h3>h3. Bootstrap heading</h3>
<h4>h4. Bootstrap heading</h4>
<h5>h5. Bootstrap heading</h5>
<h6>h6. Bootstrap heading</h6>
```

A visual representation of the Bootstrap heading hierarchy. It shows six lines of text, each representing a heading level from h1 to h6. The text is: h1. Bootstrap heading, h2. Bootstrap heading, h3. Bootstrap heading, h4. Bootstrap heading, h5. Bootstrap heading, and h6. Bootstrap heading. Each heading is indented further to the right than the one above it, illustrating the visual hierarchy and spacing provided by Bootstrap.

Pored toga, Bootstrap omogućuje predstavljanje različitih vrsta teksta, kao što je prikazano listingom i slikom ispod.

Angular Komponente

```
<p>Može se koristiti mark tag za <mark>markiranje</mark> teksta.</p>

<p><del>Element del se prikazuje kao obrisani tekst.</del></p>
<p><s>Elementom s prikazuje se tekst koji više nije ažuran.</s></p>
<p><ins>Tekst koji je umetnut u dokument predstavlja se ins elementom.</ins></p>
<p><u>Podvučen tekst se predstavlja u elementom.</u></p>
<p><strong>Za predstavljanje naglašenog teksta koristi se strong element.</strong></p>
<p><em>Italik tekst se predstavlja em elementom.</em></p>
```

Može se koristiti mark tag za markiranje teksta.

~~Element del se prikazuje kao obrisani tekst.~~

~~Elementom s prikazuje se tekst koji više nije ažuran.~~

Tekst koji je umetnut u dokument predstavlja se ins elementom.

Podvučen tekst se predstavlja u elementom.

Za predstavljanje naglašenog teksta koristi se strong element.

Italik tekst se predstavlja em elementom.

Bojom i pozadinom se koduju različite važnosti teksta.

```
<p class="text-muted">Manje bitan tekst.</p>

<p class="text-primary">Veoma bitan tekst.</p>
<p class="text-success">Tekst koji naznačava uspešnost radnje.</p>
<p class="text-info">Tekst koji predstavlja dodatne informacije.</p>
<p class="text-warning">Tekst koji upozorava.</p>
<p class="text-danger">Tekst koji ukazuje na opasnost.</p>

<p class="bg-primary">Veoma bitan tekst.</p>
<p class="bg-success">Tekst koji naznačava uspešnost radnje.</p>
<p class="bg-info">Tekst koji predstavlja dodatne informacije.</p>
<p class="bg-warning">Tekst koji upozorava.</p>
<p class="bg-danger">Tekst koji ukazuje na opasnost.</p>
```

Angular Komponente

Manje bitan tekst.

Veoma bitan tekst.

Tekst koji naznačava uspešnost radnje.

Tekst koji predstavlja dodatne informacije.

Tekst koji upozorava.

Tekst koji ukazuje na opasnost.

Veoma bitan tekst.

Tekst koji naznačava uspešnost radnje.

Tekst koji predstavlja dodatne informacije.

Tekst koji upozorava.

Tekst koji ukazuje na opasnost.

Pored navedenih svojstava teksta, Bootstrap omogućuje i brojna druga podešavanja, pa za detalje čitaoca upućujemo na <https://v4-alpha.getbootstrap.com/content/typography/>.

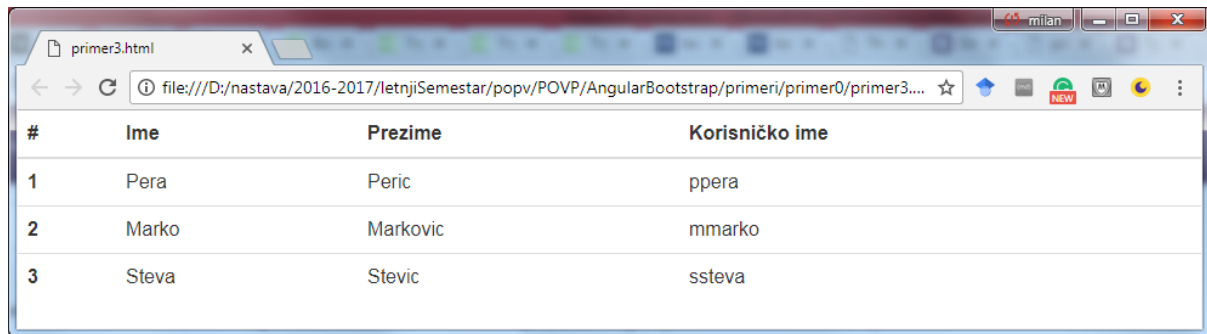
Tabele

Bootstrap omogućuje jednostavno uređivanje izgleda tabela. Osnovna klasa koju je pri tome potrebno postaviti na tabelu je `.table`. Osnovna Bootstrap tabela data je listingom i prikazana slikom ispod.

```
<table class="table">

  <thead>
    <tr>
      <th>#</th><th>Ime</th><th>Prezime</th><th>Korisničko ime</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <th>1</th><td>Pera</td><td>Peric</td><td>pperica</td>
    </tr>
    <tr>
      <th>2</th><td>Marko</td><td>Markovic</td><td>mmarko</td>
    </tr>
    <tr>
      <th>3</th><td>Steva</td><td>Stevic</td><td>ssteva</td>
    </tr>
  </tbody>
</table>
```

Angular Komponente



#	Ime	Prezime	Korisničko ime
1	Pera	Peric	ppera
2	Marko	Markovic	mmarko
3	Steva	Stevic	ssteva

Postavljanjem klase `.table-striped` postiže se efekat zebra-striped tabele. Ukoliko želim da tabela ima okvir, postavljamo `.table-bordered` klasu. Ukoliko želimo da bude indikovano iznad kog reda u tabeli se nalazi kurzor miša koristimo `.table-hover`. Ove klase se, naravno, mogu kombinovati i primer tabele za koju su postavljene sve ove klase dat je listingom i slikom ispod.

```
...  
<table class="table table-bordered table-striped table-hover">  
...
```



#	Ime	Prezime	Korisničko ime
1	Pera	Peric	ppera
2	Marko	Markovic	mmarko
3	Steva	Stevic	ssteva

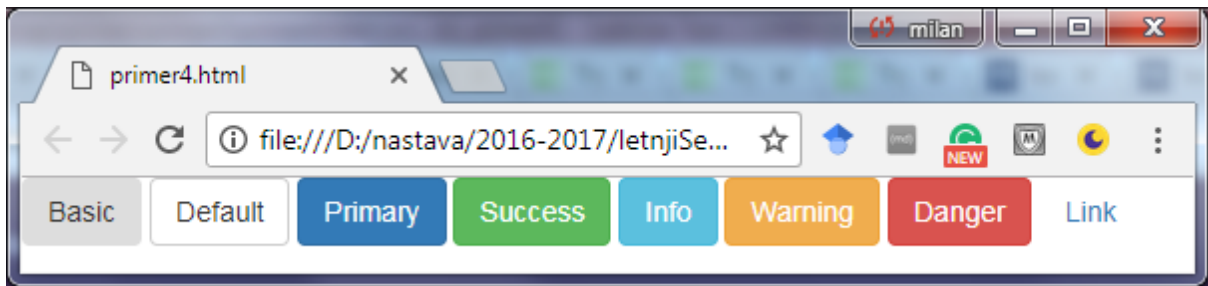
Pored ovoga, Bootstrap omogućuje i dodatno uređenje izgleda tabele, pa zainteresovanog čitaoca upućujemo na <http://getbootstrap.com/docs/4.0/content/tables/>.

Dugmići

Osnovni stil dugmića u Bootstrapu je definisan klasom `.btn`. Pored toga, stilovi dugmića organizovani su po važnosti na sličan način kao i stilovi teksta, tako da imamo, na primer, `.btn-primary` koji je plav i `.btn-success` koji je zelen. Pregled dugmića dat je listingom i slikom ispod.

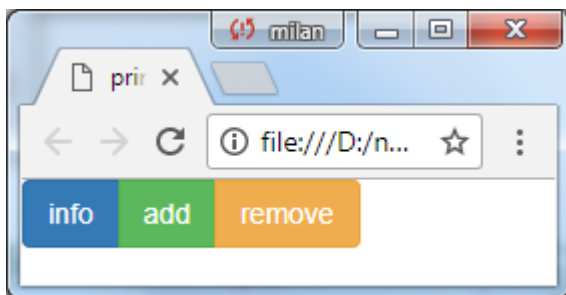
```
<button type="button" class="btn">Basic</button>  
  
<button type="button" class="btn btn-default">Default</button>  
<button type="button" class="btn btn-primary">Primary</button>  
<button type="button" class="btn btn-success">Success</button>  
<button type="button" class="btn btn-info">Info</button>  
<button type="button" class="btn btn-warning">Warning</button>  
<button type="button" class="btn btn-danger">Danger</button>  
<button type="button" class="btn btn-link">Link</button>
```

Angular Komponente



Dugmiće je moguće vizuelno organizovati u grupe i za to se koristi klasa `.btn-group`, kao što je prikazano listingom i slikom ispod.

```
<div class="btn-group">
  <button type="button" class="btn btn-primary">info</button>
  <button type="button" class="btn btn-success">add</button>
  <button type="button" class="btn btn-warning">remove</button>
</div>
```

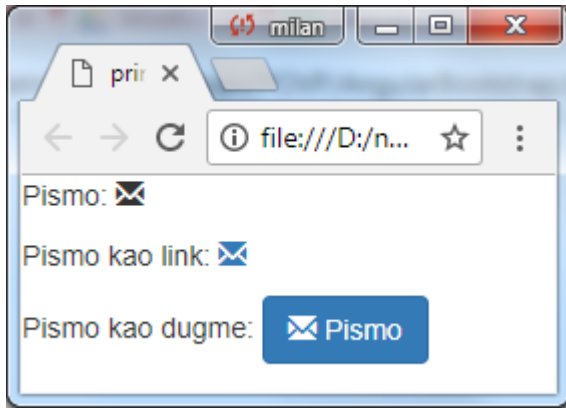


Glifikoni

Bootstrap dolazi sa velikim skupom predefinisanih ikonica (glifikona) koji se mogu jednostavno koristiti u HTML stranicama. Primeri korišćenja glifikona dat je slikom i listingom ispod.

```
<p>Pismo: <span class="glyphicon glyphicon-envelope"></span></p>
<p>Pismo kao link:
  <a href="#"><span class="glyphicon glyphicon-envelope"></span></a>
</p>
<p>Pismo kao dugme:
  <button type="button" class="btn btn-primary">
    <span class="glyphicon glyphicon-envelope"></span> Pismo
  </button>
</p>
```


Angular Komponente

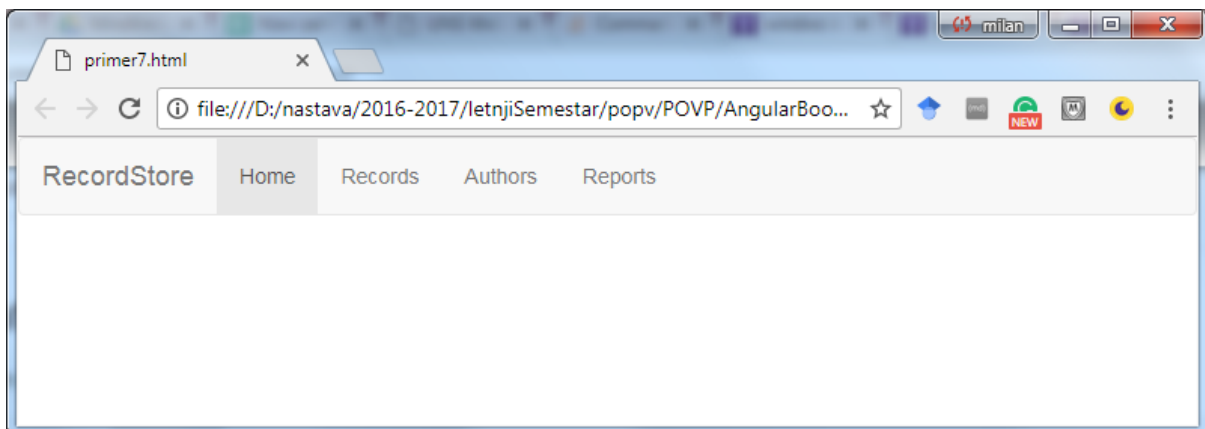


Navigacioni meni

Navigacioni meni (navbar) je horizontalni meni na samom vrhu stranice u kome se nalaze linkovi za navigaciju u aplikaciji. Element je tipa `nav` i ima postavljenu klasu `.navbar`. Pored toga ima i dodatne klase koje definišu stil: osnovna je `.navbar-default`, ali se može postaviti i `.navbar-inverse` kojim se dobija taman navigacioni meni sa svetlim tekstom. Primer je dat listingom i slikom ispod.

```
<nav class="navbar navbar-default">

  <div class="container-fluid">
    <div class="navbar-header">
      <a class="navbar-brand" href="#">RecordStore</a>
    </div>
    <ul class="nav navbar-nav">
      <li class="active"><a href="#">Home</a></li>
      <li><a href="#">Records</a></li>
      <li><a href="#">Authors</a></li>
      <li><a href="#">Reports</a></li>
    </ul>
  </div>
</nav>
```



Angular Komponente

U vep aplikacijama, *rutiranjem* se postiže da se aplikacija podeli na sekcije koje će biti prikazivane u zavisnosti od trenutnog URL-a u pregledaču. Ukoliko URL identifikuje stranicu na serveru, rutiranje uključuje slanje get zahteva za zadati URL i učitavanje i prikaz dobavljene stranice. Međutim, u jednostraničnim aplikacijama, HTML stranica se učitava samo jednom, a rutiranje podrazumeva logičku izmenu prikaza u aplikaciji.

Rutiranje je veoma važan aspekt razvoja aplikacija jer razdvaja sekcije aplikacije, pojednostavljuje održavanje stanje aplikacije i omogućuje da se sekcije aplikacije zaštite od neautorizovanog pristupa odgovarajućim pravilima rutiranja.

Rutiranje u Angular aplikacijama

U Angular aplikacijama rutiranje se realizuje zadavanjem *putanja* (eng. *paths*) za Angular komponente. Pošto će se rutiranje dešavati na klijentskoj strani, potrebno je konfigurisati Spring backend aplikaciju da, ukoliko se zahteva resurs koji ne postoji, bude izvršena redirekcija na index.html strnicu. Kod kojim se to postiže dat je listingom ispod.

```
@Bean

ErrorViewResolver supportPathBasedLocationStrategyWithoutHashes()
{
    return new ErrorViewResolver() {

        @Override
        public ModelAndView resolveErrorView(HttpServletRequest request,
            HttpStatus status, Map<String, Object> model) {
            return status == HttpStatus.NOT_FOUND
                ? new ModelAndView("index.html", Collections.<String,
                    Object>emptyMap(), HttpStatus.OK) : null;
        }
    };
}
```

Deklarisanje ruta

U Angular aplikaciji korenska putanja ruta zadaje se postavljanjem base elementa index.html strance. U našoj aplikaciji postavljeno je `<base href="/">` što znači da će korenska ruta biti `"/`.

U Angularu rutiranje je opcioni servis i nije deo Angular Core. Stoga je neophodno importovati odgovarajuće objekte (`Routes` i `RouterModule`) iz `'@angular/router'` u `app.module.ts`.

`Routes` objekat služi za predstavljanje pravila rutiranja: govori pozivom koje putanje će biti prikazana koja komponenta. Stoga je `Routes` objekat lista objekata koji sadrže atribut `route` i `component`. Objekat tipa `Routes` je prikazan listingom ispod. Vidimo da je putanja `'main'` ima postavljenu komponentu `MainComponent`, odnosno da će na URL `http://localhost:8080/main` biti prikazana `MainComponent` komponenta.

Prilikom zadavanja rute moguće je zadati i parametre putanja. Ruta za putanju `'record/:id'` omogućuje prosleđivanje parametra `id` komponenti

Angular Komponente

`RecordDetailsComponent`. Ukoliko bi korisnik uneo url `http://localhost:8080/record/5` otvorila bi se komponenta `RecordDetailsComponent` sa prosleđenim parametrom 5.

Prilikom zadavanja ruta moguće je realizovati i redirekciju. U tom slučaju se za zadatu putanju postavlja atribut `redirectTo`. U primeru u listingu ispod vidimo da će se, ukoliko korisnik unese samo `http://localhost:8080/` izvršiti redirekcija na putanju `http://localhost:8080/main`, odnosno da će se otvoriti `MainComponent`.

Obratite pažnju da poslednja ruta ima putanju `'**'`. To je vrednost koja će odgovarati bilo kojoj putanji. Postavljanjem ove putanje dobili smo da će svaki url koji počinje sa `http://localhost:8080/` koji nije poklopljen ni sa jednom unetom rutom u aplikaciji otvarati komponentu `PageNotFoundComponent`. Ovde vidimo i da se poklapanje ruta vrši tako što se prvo pokušava poklapanje prve rute, pa druge,... da bi se na kraju pokušalo poklapanje poslednje rute.

<https://angular-2-training-book.rangle.io/handout/routing/routeparams.html>

```
import { RouterModule, Routes } from '@angular/router';

...
const appRoutes: Routes = [
  { path: 'record/:id', component: RecordDetailsComponent },
  { path: 'main', component: MainComponent },
  { path: '', redirectTo: 'main', pathMatch: 'full' },
  { path: '**', component: PageNotFoundComponent }
];
...
@NgModule({
  ...
  imports: [
    ...
    RouterModule.forRoot(
      appRoutes,
      { enableTracing: true } // <-- debugging purposes only
    )
  ],
  ...
})
```

Link na rutu

Postavljanje linka u Angular aplikaciji moguće je realizovati postavljanjem običnog `<a>` elementa sa `href` atributom. Međutim, Angular uvodi direktivu `[routerLink]` koja omogućuje interno rutiranje u aplikaciji i koja pojednostavljuje rad sa rutama (npr. postavljanje parametra putanje). Primer korišćenja `routerLink` direktive dat je listingom ispod. Vidimo da `routerLink` direktiva prima niz čiji je prvi element string `'/record'`, a drugi element `record.id`. Ukoliko je vrednost `record.id` na primer 5, ovaj link bi upućivao na `http://localhost:8080/record/5`.

```
<a [routerLink]="['/record',record.id]">{{record.title}}</a>
```

Pristup parametrima rute

Da bi rutiranje bilo kompletno realizovano, potrebno je omogućiti i pristup parametrima rute. U Angularu se za to koristi poseban servis, `ActivatedRoute`, koji se importuje iz modula `'@angular/router'`. Primer pristupa parametrima rute prikazan je listingom ispod.

U konstruktoru komponente injektujemo `ActivatedRoute` u privatni atribut `route`. Preko ovog atributa možemo da pristupimo parametrima putanje, odnosno atributu `params`. Ovaj atribut je tipa `Observable`, što znači da možemo da možemo da napišemo kod koji će se pozvati svaki put kada se parametri rute promene, što je urađeno kodom `this.route.params.subscribe`. Reakcija na promenu parametara putanje je poziv `GET` metode za preuzimanje vrednosti, odnosno `this.http.get(`/api/records/${this.id}`)`.

```
Import { ActivatedRoute } from '@angular/router';

...

@Component({
  selector: 'app-record-details',
  templateUrl: './record-details.component.html',
  styleUrls: ['./record-details.component.css']
})
export class RecordDetailsComponent implements OnInit {

  id: number;

  private sub: any;

  record: Record;

  isDataAvailable: boolean;

  constructor(private route: ActivatedRoute, private http: Http) {
  }

  ngOnInit() {
    this.sub = this.route.params.subscribe(params => {
      this.isDataAvailable = false;
      //unarni + konvertuje string u number
      this.id = +params['id'];
      this.http.get(`/api/records/${this.id}`).subscribe(
        (res: Response) => {
          this.record=res.json();
          this.isDataAvailable = true;
        }
      );
    });
  }
}
```