

Rutiranje i navigacija

Ruter (Angular Router)

- Angular Router je biblioteka za rutiranje iz paketa **@angular/router**
- Omogućava navigaciju između prikaza/komponenti na osnovu URL-a (SPA navigacija)
- Navigacija može da se pokrene klikom (link/dugme), programatski, ili korišćenjem Back/Forward u browseru.
- Omogućava direktan pristup određenom prikazu unosom URL adrese u adresnu liniju
- Prosleđuje parametre rute, query parametre (i po potrebi fragment # indetifikator unutar stranice) ciljnoj komponenti

Ruta(route)

- Ruta određuje koju komponentu Angular treba da prikaže kada korisnik klikne na link ili unese adresu u adresnu liniju
- Ruta se sastoji od putanje (path) i komponente na koju je ta putanja mapirana

```
{ path: 'contact', component: ContactComponent }
```

Klasa Proizvod

ng g class proizvod

```
export class Proizvod {  
  constructor(  
    public proizvodId: number,  
    public naziv: string,  
    public cena: number  
  ) {}  
}
```

Kreiranje servisa

ng g service proizvod-service

```
export class ProizvodService {
  public vratiProizvode(): Proizvod[] {
    return [
      new Proizvod(1, 'Laptop HP Pavilion 15 (Intel i5, 16GB, 512GB SSD)', 74999),
      new Proizvod(2, 'Laptop Lenovo ThinkPad E14 (Ryzen 5, 16GB, 512GB SSD)', 82999),
      new Proizvod(3, 'Laptop Dell XPS 13 (Intel i7, 16GB, 1TB SSD)', 149999),
      new Proizvod(4, 'Laptop ASUS ZenBook 14 (Intel i7, 16GB, 1TB SSD)', 139999),
      new Proizvod(5, 'Laptop Acer Swift 3 (Ryzen 7, 16GB, 1TB SSD)', 89999),
    ];
  }

  public vratiProizvod(id: number): Proizvod | undefined {
    return this.vratiProizvode().find(
      proizvod => proizvod.proizvodId === id
    );
  }
}
```

Komponenta proizvodi - klasa

ng g c proizvodi-componnet

```
export class ProizvodiComponent implements OnInit {
  proizvodi: Proizvod[] | undefined;
  constructor(private pServis: ProizvodService) {
  }

  ngOnInit(): void {
    this.proizvodi = this.pServis.vratiProizvode();
  }
}
```

U TypeScript-u, znak | označava **union tip** (uniju tipova)
Promenljiva proizvodi može biti niz proizvoda ili undefined

Komponenta proizvodi – šablon

```
<div class="container mt-4">
  <h1 class="mb-4">Lista proizvoda</h1>
  <table class="table table-striped table-bordered">
    <thead class="thead-dark">
      <tr>
        <th>Id</th>
        <th>Naziv</th>
        <th class="text-right">Cena (RSD)</th>
      </tr>
    </thead>

    <tbody>
      @for (proizvod of proizvodi; track proizvod.proizvodId) {
        <tr>
          <td>{{ proizvod.proizvodId }}</td>
          <td>{{ proizvod.naziv }}</td>
          <td class="text-right">{{ proizvod.cena | number:'1.0-0' }}</td>
        </tr>
      }
    </tbody>
  </table>
</div>
```

Komponenta ProizvodDetalji

ng g c proizvod-detalji

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

@Component({
  selector: 'app-proizvod-detalji',
  imports: [],
  templateUrl: './proizvod-detalji.html',
  styleUrls: ['./proizvod-detalji.css'],
})
export class ProizvodDetalji {

}
```

Komponenta home

```
ng g c home-component
```

```
šablon komponente:
```

```
<div class="container mt-5">  
  <div class="jumbotron text-center">  
    <h1 class="display-4">Dobro došli!</h1>  
    <hr class="my-4">  
    <p>  
      Aplikacija demonstrira rad sa komponentama, servisima i rutiranjem.  
    </p>  
  </div>  
</div>
```

Komponenta contact

```
ng g c contact-component
```

```
<div class="container mt-5">
  <div class="card">
    <div class="card-body">
      <h1 class="card-title mb-4">Naš kontakt</h1>
      <p class="card-text">
        Akademija poslovnih studija
      </p>
      <hr>
      <p class="mb-1">
        <strong>Adresa:</strong> Beograd, Srbija
      </p>
      <p class="mb-1">
        <strong>Email:</strong> info@bapuss.edu.rs
      </p>
      <p class="mb-0">
        <strong>Telefon:</strong> +381 11 123 456
      </p>
    </div>
  </div>
</div>
```

Komponenta error

ng g c error-component

```
<div class="container mt-5">
  <div class="alert alert-danger" role="alert">
    <h1 class="alert-heading mb-3">Greška</h1>
    <p class="mb-0">
      Tražena stranica ne postoji ili je došlo do greške u zahtevu.
    </p>
  </div>
  <div class="text-muted">
    Proverite adresu u browseru ili se vratite na početnu stranicu.
  </div>
</div>
```

Kreiranje ruta u fajlu app.routes.ts

```
export const routes: Routes = [  
  { path: 'home', component: HomeComponent },  
  { path: 'contact', component: ContactComponent },  
  { path: 'proizvodi', component: ProizvodiComponent },  
  { path: 'proizvod/:id', component: ProizvodDetaljiComponent },  
  { path: 'error', component: ErrorComponent },  
  { path: '', redirectTo: 'home', pathMatch: 'full' },  
  { path: '**', redirectTo: 'error' }  
];
```

Definisanje ruta

- Tip **Routes** je u stvari niz objekata tipa **Route** iz biblioteke `@angular/router`
- Interfejs **Route** predstavlja **konfiguracioni objekat** koji definiše jednu rutu
- Svaka ruta mapira URL u komponentu
- Kada se u adresnoj liniji unese **/home**, prikazuje se **Home** komponenta
- Putanja **proizvod/:id** omogućava definisanje rute sa parametrom, npr. `/proizvod/3`
- Kada je putanja prazna (`"`), radi se o ruti za **korenski URL** (`/`) i tada se (preko redirekcije) prikazuje **Home** komponenta
- Kada je putanja postavljena na `**`, Router selektuje ovu rutu ako uneta putanja ne odgovara nijednoj prethodno definisanoj ruti, i tada se prikazuje **Error** komponenta

Fajl main.ts

```
import { bootstrapApplication } from '@angular/platform-browser';  
import { App } from './app/app';  
import { provideHttpClient } from '@angular/common/http';  
import { provideRouter } from '@angular/router';  
import { routes } from './app/app.routes';
```

```
bootstrapApplication(App, {  
  providers: [  
    provideHttpClient(),  
    provideRouter(routes)  
  ]  
});
```

- Registracija ruta u **standalone Angular aplikacijama** vrši se u fajlu **main.ts**
- Funkcija **provideRouter()** omogućava rutiranje u aplikaciji koja koristi **standalone komponente** (Angular 14 i novije verzije)

Element router-outlet i direktive routerLink i routerLinkActive

- Element **<router-outlet>** određuje mesto u template-u gde Angular Router prikazuje komponentu aktivne rute
- Direktiva **routerLink** povezuje HTML element sa rutom
- Ako se umesto routerLink koristi klasičan href, dolazi do potpunog učitavanja stranice
- Direktiva routerLink omogućava navigaciju bez potpunog osvežavanja stranice, koristeći **AJAX** pristup
- Prilikom navigacije pomoću routerLink menja se samo sadržaj u <router-outlet> elementu, dok ostatak stranice ostaje nepromenjen
- Direktiva **routerLinkActive** automatski dodaje CSS klasu aktivnom linku, što se koristi za vizuelno isticanje trenutne rute

Glavna komponenta

```
import { Component } from '@angular/core';
import { RouterOutlet, RouterLink, RouterLinkActive } from "@angular/router";

@Component({
  selector: 'app-root',
  imports: [RouterOutlet, RouterLink, RouterLinkActive],
  templateUrl: './app.html',
  styleUrls: ['./app.css']
})
export class App {
  title = 'Rutiranje i navigacija';
}
```

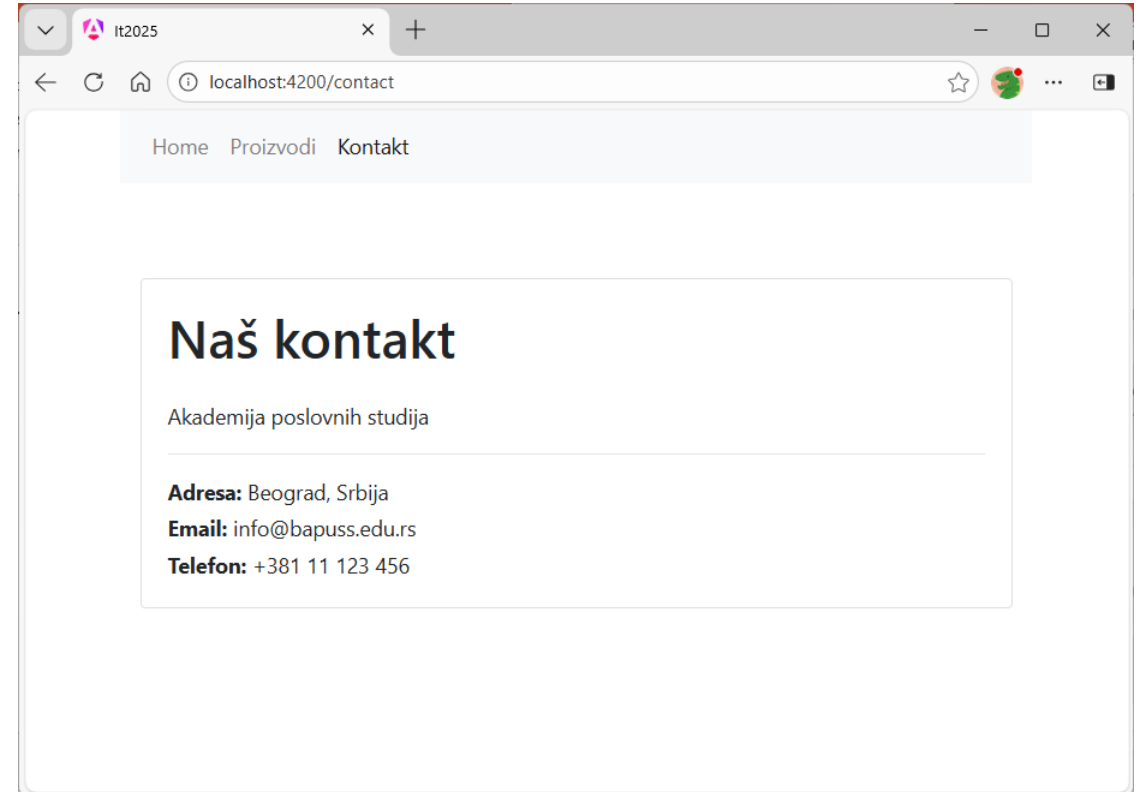
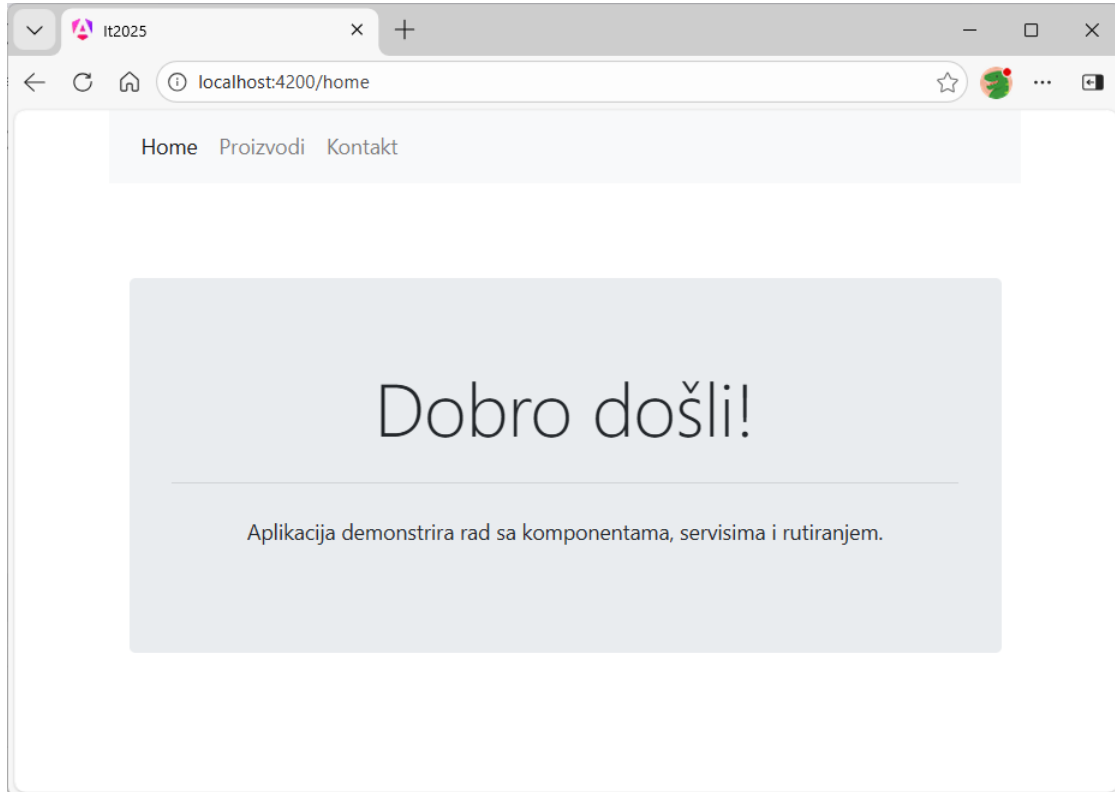
Šablon glavne komponente - navigacija

```
<div class="container">
  <!-- Navigacija -->
  <nav class="navbar navbar-expand navbar-light bg-light mb-4">
    <ul class="navbar-nav">
      <li class="nav-item">
        <a class="nav-link" routerLink="/home" routerLinkActive="active">
          Home
        </a>
      </li>
      <li class="nav-item">
        <a class="nav-link" routerLink="/proizvodi" routerLinkActive="active">
          Proizvodi
        </a>
      </li>
      <li class="nav-item">
        <a class="nav-link" routerLink="/contact" routerLinkActive="active">
          Kontakt
        </a>
      </li>
    </ul>
  </nav>
  ...
</div>
```

Šablon glavne komponente – sadržaj rute

```
<!-- Sadržaj rute -->  
<div class="row">  
  <div class="col-12">  
    <router-outlet></router-outlet>  
  </div>  
</div>
```

Prikaz menija i komponente Home i Contact



Navigacija i aktivna ruta u Angularu

- Aktivna ruta pridružena učitanoj komponenti predstavlja se instancom servisa **ActivatedRoute**
- **Angular Router** je **servis** koji omogućava navigaciju između komponenti klikom na link, dugme ili korišćenjem back/forward dugmadi u browseru
- Metoda **navigate()** servisa Router omogućava programatsku navigaciju do komponente koja je pridružena putanji
- **ActivatedRoute** se koristi za čitanje parametara rute, dok se **Router** koristi za navigaciju

Čitanje parametara rute u Angularu

- Parametri rute se definišu u putanji pomoću prefiksa :
(npr. **proizvod/:id**)
- Vrednosti parametara zavise od URL-a
(npr. /proizvod/3 → id = 3)
- Angular omogućava pristup parametrima **aktivne** rute preko servisa **ActivatedRoute**
- Parametri rute se čitaju pomoću svojstva **paramMap** objekta **ActivatedRoute**
- **route.paramMap** je Observable tipa **ParamMap**
- ParamMap predstavlja parametre rute u obliku ključ–vrednost
- Metoda **paramMap.get(imeParametra)** vraća vrednost parametra kao string
 - Ako parametar ne postoji, metoda get() vraća null
- Korišćenjem **paramMap.subscribe()** komponenta reaguje na svaku promenu parametra rute

Metoda navigate() servisa Router

- Servis Router omogućava programatsku navigaciju unutar Angular aplikacije
- Metoda **navigate()** menja aktivnu rutu bez potpunog osvežavanja stranice
- Kao argument prima niz segmenata putanje

```
this.router.navigate(['/proizvodi']);
```

Navigacija na rutu **/proizvodi**

```
this.router.navigate(['/proizvod', 3]);
```

Navigacija na **/proizvod/3**

Komponenta ProizvodDetalji

```
export class ProizvodDetaljiComponent implements OnInit {
  proizvod?: Proizvod; //opciono svojstvo
  constructor(
    private route: ActivatedRoute,
    private pServis: ProizvodService,
    private router: Router
  ) {}

  ngOnInit(): void {
    this.route.paramMap.subscribe(paramMap => {
      const idParam = paramMap.get('id');
      const id = Number(idParam);
      if (!idParam || Number.isNaN(id)) {
        this.router.navigate(['/proizvodi']);
        return;
      }
      this.proizvod = this.pServis.vratiProizvod(id);
      if (!this.proizvod) {
        this.router.navigate(['/proizvodi']);
      }
    });
  }

  nazad(): void {
    this.router.navigate(['/proizvodi']);
  }
}
```

Šablon komponente ProizvodDetalji

```
@if (proizvod) {  
  <div class="card">  
    <div class="card-header">  
      {{ proizvod.naziv }}  
    </div>  
  
    <div class="card-body">  
      <strong>Id:</strong> {{ proizvod.proizvodId }} <br>  
      <strong>Naziv:</strong> {{ proizvod.naziv }} <br>  
      <strong>Cena:</strong> {{ proizvod.cena }} <br><br>  
  
      <button class="btn btn-primary" (click)="nazad()">Proizvodi</button>  
    </div>  
  </div>  
}
```

Izmena šablona komponente proizvodi

```
<tbody>
  @for (proizvod of proizvodi; track proizvod.proizvodId) {
  <tr>
    <td>{{ proizvod.proizvodId }}</td>
    <td>{{ proizvod.naziv }}</td>
    <td class="text-right">
      {{ proizvod.cena | number:'1.0-0' }}
    </td>
    <td>
      <a [routerLink]="['/proizvod', proizvod.proizvodId]">
        Vidi detalje
      </a>
    </td>
  </tr>
  }
</tbody>
```

Korisnički interfejs

Home Proizvodi Kontakt

Lista proizvoda

Id	Naziv	Cena (RSD)	
1	Laptop HP Pavilion 15 (Intel i5, 16GB, 512GB SSD)	74,999	Vidi detalje
2	Laptop Lenovo ThinkPad E14 (Ryzen 5, 16GB, 512GB SSD)	82,999	Vidi detalje
3	Laptop Dell XPS 13 (Intel i7, 16GB, 1TB SSD)	149,999	Vidi detalje
4	Laptop ASUS ZenBook 14 (Intel i7, 16GB, 1TB SSD)	139,999	Vidi detalje
5	Laptop Acer Swift 3 (Ryzen 7, 16GB, 1TB SSD)	89,999	Vidi detalje

Home Proizvodi Kontakt

Laptop HP Pavilion 15 (Intel i5, 16GB, 512GB SSD)

Id: 1
Naziv: Laptop HP Pavilion 15 (Intel i5, 16GB, 512GB SSD)
Cena: 74999

[Proizvodi](#)

Pitanje 1

Mesto u šablonu gde Angular Router prikazuje komponentu aktivne rute definiše se elementom:

- a. `<router-outlet>`
- b. `<router-views>`
- c. `<router-show>`

Odgovor: a

Pitanje 2

Povezivanje html elementa sa rutom vrši se korišćenjem direktive:

- a. routerView
- b. routerLink
- c. router-outlet

Odgovor: b

Pitanje 3

Aktivna ruta pridružena učitanoj komponenti predstavlja se objektom klase:

- a. CurrentRoute
- b. ActivatedRoute
- c. ActiveRouter

Odgovor: b

Pitanje 4

Ruta se definiše kao javascript objekat koji ima svojstva:

- a. path i component
- b. routeName i url
- c. route i componentName

Odgovor: a

Pitanje 5

Ako se definiše ruta sa svojstvom path postavljenim na vrednost praznog stringa radi se o :

- a. ruti kojom se prikazuje komponenta sa porukom o grešci
- b. default ruti
- c. ruti koja odgovara svakoj url adresi

Odgovor: b

Pitanje 6

Registracija ruta kod Angular standalone aplikacija vrši se:

- a. U `app.module.ts` fajlu koristeći `RouterModule.forRoot()`
- b. U `main.ts` fajlu koristeći `provideRouter()` metode
- c. U `app.component.ts` fajlu koristeći `RouterModule.forChild()`

Odgovor: b

Pitanje 7

Gde se u Angular standalone aplikaciji definiše niz ruta (Routes)?

- a. U fajlu main.ts
- b. U fajlu app.routes.ts
- c. U fajlu app.module.ts

Odgovor: b