

Visoka škola strukovnih studija za informacione i
komunikacione tehnologije
(Visoka ICT škola)

Web programiranje
Angular
Dokumentacija za sajt

Autor:
Jelena Miletic 171/12
Beograd, jun 2019.

Sadržaj

- 1. Uvod
- 1.1 Opis funkcionalnosti
- 2 Stranice
- 3. Kodovi

1.1 Opis funkcionalnosti

Sajt definiše blog na kome korisnici mogu gledati postove po kategorijama. Takođe se nalazi filtracija po kategorijama uradjena pomocu pipes. Rad sa formularom se nalazi na stranici contact, gde je uradjena provera elemenata I obrada gresaka. Binding se nalazi takođe na stranici contact gde se ispisuje prilikom upisivanja. Animacije se nalaze na stranici gde se ispisuje svaki post posebno , prilikom unosa komentara , komentar se ispisuje ispod I boji u drugu boju. Radjeno je sa JSON fajlom.

2.Stranice

Index

```
<!DOCTYPE html>
<html lang="en">

<head>

  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
  <meta name="description" content="">
  <meta name="author" content="">

  <title>Clean Blog - Start Bootstrap Theme</title>
  <base href="/" />

  <!-- Bootstrap core CSS -->
  <link href="assets/vendor/bootstrap/css/bootstrap.min.css" rel="stylesheet">

  <!-- Custom fonts for this template -->
  <link href="assets/vendor/fontawesome-free/css/all.min.css" rel="stylesheet"
type="text/css">
  <link
href='https://fonts.googleapis.com/css?family=Lora:400,700,400italic,700italic'
rel='stylesheet' type='text/css'>
  <link
href='https://fonts.googleapis.com/css?family=Open+Sans:300italic,400italic,600italic,700italic,800italic,400,300,600,700,800' rel='stylesheet' type='text/css'>

  <!-- Custom styles for this template -->
  <link href="assets/css/clean-blog.min.css" rel="stylesheet">

</head>

<body>

  <!-- Navigation -->
  <app-navigacija></app-navigacija>

  <!-- Page Header -->
  <app-header></app-header>

  <!-- Main Content -->
  <div class="container">
    <app-root></app-root>
```

```

</div>

<hr>

<!-- Footer -->
<footer>
  <div class="container">
    <div class="row">
      <div class="col-lg-8 col-md-10 mx-auto">
        <ul class="list-inline text-center">
          <li class="list-inline-item">
            <a href="#">
              <span class="fa-stack fa-lg">
                <i class="fas fa-circle fa-stack-2x"></i>
                <i class="fab fa-twitter fa-stack-1x fa-inverse"></i>
              </span>
            </a>
          </li>
          <li class="list-inline-item">
            <a href="#">
              <span class="fa-stack fa-lg">
                <i class="fas fa-circle fa-stack-2x"></i>
                <i class="fab fa-facebook-f fa-stack-1x fa-inverse"></i>
              </span>
            </a>
          </li>
          <li class="list-inline-item">
            <a href="#">
              <span class="fa-stack fa-lg">
                <i class="fas fa-circle fa-stack-2x"></i>
                <i class="fab fa-github fa-stack-1x fa-inverse"></i>
              </span>
            </a>
          </li>
        </ul>
        <p class="copyright text-muted">Copyright &copy; Your Website 2019</p>
      </div>
    </div>
  </div>
</footer>

<!-- Bootstrap core JavaScript -->
<script src="assets/vendor/jquery/jquery.min.js"></script>
<script src="assets/vendor/bootstrap/js/bootstrap.bundle.min.js"></script>

```

```

    <!-- Custom scripts for this template -->
    <script src="assets/js/clean-blog.min.js"></script>

</body>

</html>

```

HeaderComponent

```

<header class="masthead" style="background-image: url('assets/img/home-bg.jpg')">
  <div class="overlay"></div>
  <div class="container">
    <div class="row">
      <div class="col-lg-8 col-md-10 mx-auto">
        <div class="site-heading">
          <h1>Clean Blog</h1>
          <span class="subheading">A Blog Theme by Start Bootstrap</span>
        </div>
      </div>
    </div>
  </div>
</header>

```

NavigacijaComponent

```

<nav class="navbar navbar-expand-lg navbar-light fixed-top" id="mainNav">
  <div class="container">

    <button class="navbar-toggler navbar-toggler-right" type="button" data-
toggle="collapse" data-target="#navbarResponsive" aria-
controls="navbarResponsive" aria-expanded="false" aria-label="Toggle navigation">
      Menu
      <i class="fas fa-bars"></i>
    </button>
    <div class="collapse navbar-collapse" id="navbarResponsive">
      <ul class="navbar-nav ml-auto">

        <li class="nav-item">
          <a class="nav-link" routerLink="">Home</a>
        </li>

        <li class="nav-item">
          <a class="nav-link" routerLink="/login">About me</a>
        </li>

        <li class="nav-item">
          <a class="nav-link" routerLink="/contact">Contact</a>

```

```

        </li>

    </ul>
</div>
</div>
</nav>

```

Home-Component

Home.html

```

<router-outlet></router-outlet>
<div class="row">

    <div class="col-lg-8 col-md-10 mx-auto">
        <input type="text" placeholder="Searching by category"
        [(ngModel)]="searchTeam" >

        <div *ngFor="let po of post | orderBy:searchTeam" class="post-preview">

            <h2 class="post-title">
                <a [routerLink]="['/post', po.Id]">{{po.Naziv | uppercase}}</a>

            </h2>
            <h5 class="post-subtitle">
                {{po.Tekst|lowercase}}
            </h5>

            <p class="post-meta">Posted by
                {{po.Korisnik}}
                <br>
                Date:
                {{po.Datum | date:'dd/MM/y' }}
                <br>
                Category: {{po.Kategorija}}</p>

        </div>
        <hr>

    </div>

</div>
</div>

```

Home.ts

```
import { Component, OnInit } from '@angular/core';
import { PostService } from '../postovi.service';
import { Router, ActivatedRoute, Params } from '@angular/router';

@Component({
  selector: 'app-home',
  templateUrl: './home.component.html',
  styleUrls: ['./home.component.css']
})
export class HomeComponent implements OnInit {
  post;
  id;
  search:string;
  constructor(private service: PostService, private route: Router,private
putanje:ActivatedRoute) {
    this.id=putanje.snapshot.params['id']
  }

  ngOnInit() {

    this.service.dohvatiPostove().subscribe(

      podaci=>{
        console.log(podaci);
        this.post=podaci;
      },
      error => {
        console.log(error);
      }
    );

  }

}
```

SingleComponent

Single.html

```
<router-outlet></router-outlet>
<div class="row">

  <div class="col-lg-8 col-md-10 mx-auto" *ngFor="let po of post">
    <a href="#">
      
    </a>
    <p>{{po.Naziv}}</p>

    <p>{{po.Tekst}}</p>

    <app-comentar></app-comentar>
  </div>
</div>
```

Single.ts

```
import { Component, OnInit } from '@angular/core';
import { PostService } from '../postovi.service';
import { Router, ActivatedRoute, Params } from '@angular/router';

@Component({
  selector: 'app-single',
  templateUrl: './single.component.html',
  styleUrls: ['./single.component.css']
})
export class SingleComponent implements OnInit {
  post;

  constructor(private service: PostService, private route: ActivatedRoute) { }

  ngOnInit() {

    let id=(this.route.snapshot.params['id']);
    this.service.dohvatiPost(id).subscribe(po=>{
      console.log(po);
      this.post=po;
    },
    error => {
      console.log(error);
    }
  )
}
```

```
);
}

}
```

CommentarComponent

Commentar.html

```
<div class="row">
  <div class="col-lg-8 col-md-10 mx-auto">

    <form name="sendMessage" id="contactForm" novalidate>

      <div class="control-group">
        <div class="form-group floating-label-form-group controls">
          <label>Comment</label>
          <textarea [(ngModel)]="messages" [ngModelOptions]="{standalone:
            true}" rows="5" class="form-control" placeholder="Comment"
id="message" required data-validation-required-message="Please enter a
comment."></textarea>
          <p class="help-block text-danger"></p>
        </div>
      </div>
      <br>
      <div id="success"></div>
      <div class="form-group">
        <button type="submit" class="btn btn-primary" id="sendMessageButton"
(click)="pushItem()" [disabled]="messages===''">Send</button>

      </div>
    </form>

    <br>
    <p *ngFor="let po of post" >{{po.Message}}</p>
    <ul>

      <li @fade *ngFor="let item of items">{{item}}
        <span (click)="RemoveItem(item)"></span>
      </li>

    </ul>
  </div>
</div>
```

Comment.ts

```
import { Component, OnInit } from '@angular/core';
import { trigger, state, animate, transition, style } from '@angular/animations';
import { CommentService } from '../comment.service';

@Component({
  selector: 'app-comentar',
  templateUrl: './comentar.component.html',
  styleUrls: ['./comentar.component.css'],
  animations: [
    trigger('fade', [
      state('void', style({
        display: 'none'
      })),
    ]),
  ],
  state: '*',
  style: {
    background: 'silver'
  },
  transition: 'void=>*',
  animate: 2000,
  transition: '*=>void',
  animate: 2000,
})
export class ComentarComponent implements OnInit {
  post;

  items: any[] = [];
  constructor(private service: CommentService) {}

  ngOnInit() {
    this.service.dohvatiComment().subscribe(
      podaci => {
        console.log(podaci);
        this.post = podaci;
      },
      error => {
        console.log(error);
      }
    );
  }
}
```

```

    }
  );

}
messages='';

pushItem(){
  if(this.messages != ''){
    this.items.push(this.messages);
    this.messages='';
  }
}
RemoveItem(item){
  let index=this.items.indexOf(item);
  this.items.slice(index,1);
}
}

```

ContactComponent

Contact.html

```

<div class="row">
  <div class="col-lg-8 col-md-10 mx-auto">
    <p>Want to get in touch? Fill out the form below to send me a message and I
will get back to you as soon as possible!</p>

    <form (ngSubmit)="onSubmit(f)" #f="ngForm" name="sentMessage"
id="contactForm" novalidate>
      <div class="control-group">
        <div class="form-group floating-label-form-group controls">
          <label>Name</label>
          <input type="text" name="name" [(ngModel)]="name" #nameOne="ngModel"
required minlength="4" #namee="ngModel" class="form-control ng-invalid"

```

```

placeholder="Name" id="name" required data-validation-required-message="Please
enter your name.">
    <p *ngIf="name.invalid && name.touched && name.required">Name is
required</p>

    </div>

</div>

<div class="control-group">
    <div class="form-group floating-label-form-group controls">
        <label>Email Address</label>
        <input type="email" [(ngModel)]="email" name="email"
#email="ngModel" required email #emailOne="ngModel" class="form-control ng-
invalid" placeholder="Email Address" id="email" required data-validation-
required-message="Please enter your email address.">
        <p class="help-block text-danger" *ngIf="email.invalid">Pleas enter
valid email</p>
    </div>
</div>

<div class="control-group">
    <div class="form-group floating-label-form-group controls">
        <label>Message</label>
        <textarea [(ngModel)]="message" [ngModelOptions]="{standalone: true}"
name="message"#message="ngModel" rows="5" minlength="6" required #message="ngModel"
class="form-control ng-invalid" placeholder="Message" id="message" required data-
validation-required-message="Please enter a message."></textarea>
        <p class="help-block text-danger" *ngIf="message.invalid &&
message.minlength" >Email mas have min 4 character </p>
    </div>
</div>
<br>
<div id="success"></div>
<div class="form-group">
    <button name="submit" type="submit" class="btn btn-primary"
(click)="message=''" id="sendMessageButton" [disabled]="f.invalid">Send</button>
</div>
{{message}}
</form>
</div>
</div>

```

Contact.ts

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

```

import { NgForm } from '@angular/forms';

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

  name='';
  email='';
  message='';
  constructor() { }

  ngOnInit() {

  }

  onSubmit(e){

    console.log(e);}

}

```

AboutComponent

About.html

```

<div class="row">

  <div class="col-lg-8 col-md-10 mx-auto" >
    <a href="#">
      
    </a>
  </div>
</div>

```

```

        <p style="text-align: center "><b>Ime i prezime</b>: Jelena
Miletic</p>
        <p style="text-align: center "><b>Datum rođenja</b>:
02.11.1993.</p>
        <p style="text-align: center "><b>Mesto rođenja</b>: Topola</p>
        <p style="text-align: center "><b>Mesto prebivalista</b>:
Beograd</p>
        <p style="text-align: center "><b>Adresa</b>: Brace Jerkovic
71</p>
        <p style="text-align: center "><b>Obrazovanje:</b>
        <ul>
            <li style="text-align: center ">Osnovna skola
"Karadjordje",<br> Topola</li>
            <li style="text-align: center ">Gimnazija "Kralj Petar I",
<br>smer: Opsti , Topola</li>
            <li style="text-align: center ">Visoka ICT,smer:IT,
Beograd</li>
        </ul>

    </div>
</div>

```

PostServices

```

import { Injectable } from '@angular/core';

import { HttpClient } from '@angular/common/http';

@Injectable({
  providedIn: 'root'
})
export class PostService {

  constructor( private service : HttpClient){

  }

  dohvatiPostove(){

    var post =
this.service.get("http://localhost:4200/assets/json/post.json");

```

```

        return post;
    }

    dohvatiPost(id:number){

        var post = this.service.get("http://localhost:4200/assets/json/post.json");
        return post;
    }

}

```

OrderByPipes

```

import { Pipe, PipeTransform } from '@angular/core';

@Pipe({
    name: 'orderBy'
})
export class OrderByPipe implements PipeTransform {

    transform(value: any, args?: any): any {
        if(!args){
            return value;
        }
        else{
            args=args.toUpperCase();
        }
        return value.filter(items=>{
            return items.Kategorija.startsWith(args)==true;
        })
    }

}

```

CommentService

```

import { Injectable } from '@angular/core';

import { HttpClient } from '@angular/common/http';

@Injectable({
    providedIn: 'root'

```



```

}))
export class CommentService {

    constructor( private service : HttpClient){

    }

    add(message){
        var posts =
this.service.post("http://localhost:4200/assets/json/comment.json",message);
        return posts;

    }

    dohvatiComment(){

        var post =
this.service.get("http://localhost:4200/assets/json/comment.json");
        return post;
    }

}

```

Post.json

```

[
  {
    "Id":1,
    "Naziv":"3 Benefits of Minimalism In Interior Design.",
    "Tekst":" Design aficionado will know that minimalism is a growing trend
among modern interior designers. But why is minimalism so popular? For those who
might be unfamiliar with the term, minimalism is a design concept that strips
things down to their bare essentials.",
    "Korisnik":"Jesifer Albert",
    "Datum":"on September 24, 2019",
    "Kategorija":"DESIGN",
    "Slika":"lamp-800.jpg"
  },
  {
    "Id":2,
    "Naziv":"10 Interesting Facts About Caffeine.",

```

```

        "Tekst": " People may not think of caffeine as the most popular mood-
altering drug in the world, even those who use it daily, by drinking coffee, tea,
sodas or energy drinks as part of their routine.",
        "Korisnik": "Mike Jonson",
        "Datum": "on Octobar 03, 2019",
        "Kategorija": "HEALTH",
        "Slika": "tulips-800.jpg"
    },
    {
        "Id": 3,
        "Naziv": " The Power of Music to Reduce Stress",
        "Tekst": " Listening to music can have a tremendously relaxing effect on
our minds and bodies, especially slow, quiet classical music. This type of music
can have a beneficial effect on our physiological functions.",
        "Korisnik": "Jenuy Milbert",
        "Datum": "on May 15, 2019",
        "Kategorija": "DESIGN",
        "Slika": "music-800.jpg"
    },
    {
        "Id": 4,
        "Naziv": "The Pomodoro Technique Really Works",
        "Tekst": " Admittedly, I've never been big on productivity hacks and
tricks. Instead, I keep things pretty simple. I take a look at my planner (yes, a
real paper planner—not my phone), jot down a list of things I want to get done
that day, and then start hustling.",
        "Korisnik": "Melany Stuff",
        "Datum": "on Aprile 10, 2019",
        "Kategorija": "MANAGEMENT",
        "Slika": "watch-800.jpg"
    }
]

```

Comment.json

```

[
    {
        "Id": 1,
        "Message": "I hate this"
    },
    {
        "Id": 2,
        "Message": "I Love this"
    }
]

```

```
}  
  
]
```

Category.json

```
[  
  {  
    "id" : 1,  
    "naziv" : "Travel"  
  },  
  {  
    "id": 2,  
    "naziv" : "LifeStyle"  
  },  
  {  
    "id": 3,  
    "naziv" : "Food"  
  }  
]
```