

PRAKTIKUM / ANGULAR

DOKUMENTACIJA – MARKO MUTAVDŽIĆ 125/16

1. UVOD

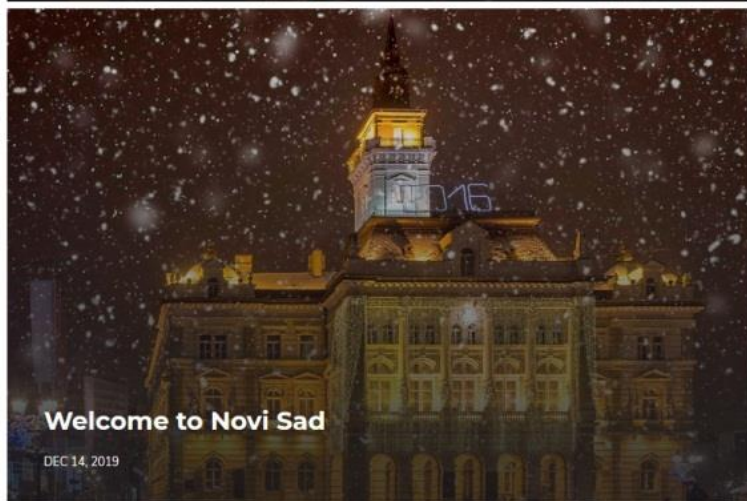
Tema ovog sajta je blog za putnike i sve ljubitelje putovanja, sajt je zamišljen kao blog gde korisnici mogu svakodnevno pregledati najnovije i najatraktivnije destinacije Evrope, takodje mogu komentarisati svaki od postova. Svaki post dodeljen je jednoj kategoriji tj. državi.

1.1. Korišćeni programski jezici

Prilikom izrade sajta korišćeni su: HTML4, HTML5, CSS2, CSS3, JavaScript, JSON, TypeScript, Angular.

1.2. Template

- Navigacioni deo
 - o Meni i linkovi društvenih mreža, kao i bočni meni
- Sadržaj
 - o Mesto na kom se učitava sadržaj u zavisnosti od tražene strane
- Footer
 - o Osnovne informacije i meni



2. KOD

Index.html

```
<!DOCTYPE
html>

<html lang="en">
<head>
  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <!-- The above 3 meta tags *must* come first in the head; any other head content
must come *after* these tags -->
  <title>Travel diary</title>

  <base href="/" /> <!-- Indeks stranica je ruta sajta i od nje se krece -->
  <!-- Google font -->
  <link
href="https://fonts.googleapis.com/css?family=Montserrat:400,700%7CMuli:400,700"
rel="stylesheet">
  <!-- Bootstrap -->
  <link type="text/css" rel="stylesheet" href="assets/css/bootstrap.min.css" />
  <!-- Font Awesome Icon -->
  <link rel="stylesheet" href="assets/css/font-awesome.min.css">
  <!-- Custom stlylesheet -->
  <link type="text/css" rel="stylesheet" href="assets/css/style.css" />
</head>
<body>
  <!-- HEADER -->
  <app-nav></app-nav>
  <!-- /HEADER -->
  <!-- SADRZAJ -->
  <div id="sadrzaj">
    <app-root></app-root>
  </div>
  <!-- /SADRZAJ -->
  <!-- FOOTER -->
  <app-footer></app-footer>
  <!-- /FOOTER -->
  <!-- jQuery Plugins -->
  <script src="assets/js/jquery.min.js"></script>
  <script src="assets/js/bootstrap.min.js"></script>
  <script src="assets/js/jquery.stellar.min.js"></script>
  <script src="assets/js/main.js"></script>
</body>
```

app.module.ts

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

  import { BrowserModule } from '@angular/platform-browser';
  import { HttpClientModule } from '@angular/common/http';
  import { AppRoutingModule } from './app-routing.module';
  import { ReactiveFormsModule } from '@angular/forms';
  import { BrowserAnimationsModule } from '@angular/platform-browser/animations';
  import { AppComponent } from './app.component';
  import { NavComponent } from './nav/nav.component';
  import { FooterComponent } from './footer/footer.component';
  import { ContactComponent } from './contact/contact.component';
  import { AboutUsComponent } from './about-us/about-us.component';
  import { HomeComponent } from './home/home.component';
  import { AuthorComponent } from './author/author.component';
  import { AsideComponent } from './aside/aside.component';
  import { DestinationsComponent } from './destinations/destinations.component';
  import { DestinationComponent } from './destination/destination.component';
  import { NotFoundComponent } from './not-found/not-found.component';
  import { NavService } from './services/nav.service';
  import { DestinationsService } from './services/destinations.service';
  import { ShortenPipe } from './pipes/shorten.pipe';
  import { SortDescPipe } from './pipes/sortDesc.pipe';
  import { SortAscPipe } from './pipes/sort-asc.pipe';
  @NgModule({
    declarations: [
      AppComponent,
      NavComponent,
      FooterComponent,
      ContactComponent,
      AboutUsComponent,
      HomeComponent,
      AuthorComponent,
      AsideComponent,
      DestinationsComponent,
      ShortenPipe,
      SortDescPipe,
      SortAscPipe,
      DestinationComponent,
      NotFoundComponent
    ],
    imports: [
      BrowserModule,
      BrowserAnimationsModule,
      HttpClientModule,
      AppRoutingModule,
      ReactiveFormsModule
    ],
    providers: [NavService, DestinationsService],
```

```
        bootstrap: [AppComponent, NavComponent, FooterComponent]
    })
    export class AppModule { }
```

app.component.ts

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

@Component({
  selector: 'app-root',
  templateUrl: './app.component.html',
  styleUrls: ['./app.component.css']
})
export class AppComponent {
  title = 'travel';
}
```

app.component.html

```
<router-outlet></router-outlet>
```

app-routing.module.ts

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

import { Routes, RouterModule } from '@angular/router';
import { ContactComponent } from './contact/contact.component';
import { AboutUsComponent } from './about-us/about-us.component';
import { HomeComponent } from './home/home.component';
import { AuthorComponent } from './author/author.component';
import { DestinationsComponent } from './destinations/destinations.component';
import { DestinationComponent } from './destination/destination.component';
import { NotFoundComponent } from './not-found/not-found.component';
const routes: Routes = [
  {
    path: '',
    component: HomeComponent
  },
  {
    path: 'contact',
    component: ContactComponent
  },
  {
    path: 'about-us',
    component: AboutUsComponent
  },

```

```

    {
      path: 'author',
      component: AuthorComponent
    },
    {
      path: 'destinations',
      component: DestinationsComponent
    },
    {
      path: 'destinations/category/:category',
      component: DestinationsComponent
    },
    {
      path: 'destination/:id',
      component: DestinationComponent
    },
    {
      path: 'not-found',
      component: NotFoundComponent
    },
    {
      path: '**',
      redirectTo: '/not-found'
    }
  ];
  @NgModule({
    imports: [RouterModule.forRoot(routes)],
    exports: [RouterModule]
  })
  export class AppRoutingModule { }

```

about-us.component.html

```

<!--
SECTION
-->

<div class="section">
  <!-- container -->
  <div class="container">
    <!-- row -->
    <div class="row">
      <div class="col-md-5">
        

        <div class="section-row">
          <div class="section-title">
            <h2 class="title">Our story</h2>
          </div>
          <p>We were inspired to explore the world by
following other people's blogs online. Travel blogs are a great source for cool ideas and tips

```

if you're planning a trip! Want to travel and learn more about the most beautiful cities in the world? You're in the right place.</p>

```
</div>
</div>
<div class="col-md-7">
  <div class="section-row">
    <div class="section-title">
      <h2 class="title">Our Vision</h2>
    </div>
    <p>You are probably tired of being stuck in a
hamster cart and want to regain some freedom, ease and joy.
Each of us has our own interpretation of
what the ultimate form of travel looks like.

I am sure we can agree that travel allows us
to avoid daily formalities, headaches and routines and gives us a sense of ownership of life.

It offers an exotic spark of adventure,
wonder and discovery that seems to flood our lives day by day.

Here you can find how to spend your vacation
on the best way... We will tell you where to go, what to eat, what to visit, where to sleep,
where not to go, what to avoid, how to get across Europe on the fastest and cheapest way.</p>
</div>
</div>
<!-- /row -->
</div>
<!-- /container -->
</div>
<!-- /SECTION -->
```

about-us.component.ts

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

import { slide } from '../animations';
@Component({
  selector: 'app-about-us',
  templateUrl: './about-us.component.html',
  styleUrls: ['./about-us.component.css'],
  animations: [
    slide
  ]
})
export class AboutUsComponent implements OnInit {
  constructor() { }
  ngOnInit() {
```



```
}
```

aside.component.html

```
<div
class="col-
md-4">

  <!-- post widget -->
  <div class="aside-widget">

    <div class="section-title">
      <h2 class="title">Recent Posts</h2> <!--popular-->
    </div>

    <!-- post -->
    <div class="post post-widget" *ngFor="let dest of destinations | slice:0:8 |
sortDesc: 'date'">
      <a class="post-img" routerLink="/destination/{{dest.id}}"
[routerLink]="['/destination',dest.id]"></a>
      <div class="post-body">
        <div class="post-category">
          <b>#{{dest.category}}</b>
        </div>
        <h3 class="post-title"><a routerLink="/destination/{{dest.id}}">{{dest.title
| shorten:40}}</a></h3>
      </div>
    </div>
  <!-- /post -->
</div>
<!-- /post widget -->
</div>
```

aside.component.ts

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

import { DestinationsService } from '../services/destinations.service';
@Component({
  selector: 'app-aside',
  templateUrl: './aside.component.html',
  styleUrls: ['./aside.component.css']
})
export class AsideComponent implements OnInit {
  destinations: Object;
  constructor(private service : DestinationsService) {
  }
  ngOnInit() {
    this.service.getDestinations().subscribe( destinations => { this.destinations =
```

```
destinations; }));
    }
}
```

author.component.html

```
<!--
PAGE
HEADER
-->

<div class="page-header">
  <div class="container">
    <div class="row">
      <div class="col-md-offset-1 col-md-10 text-center">
        <div class="author">
          

          <h1>Marko Mutavdžić</h1>
          <ul class="author-social">
            <li><a href="https://github.com/mutavdzitch"><i class="fa fa-
github"></i></a></li>
            <li><a href="https://www.linkedin.com/in/mutavdzic/"><i class="fa
fa-linkedin"></i></a></li>
            <li><a href="#"><i class="fa fa-google-
plus"></i></a></li>
            <li><a href="#">Documentation: <i class="fa
fa-file"></i></a></li>
          </ul>
        </div>
      </div>
    </div>
  </div>
</div>

<!-- /PAGE HEADER -->

<!-- SECTION -->
<div class="section">
  <!-- container -->
  <div class="container">
    <!-- row -->
    <div class="row">
      <div class="col-md-5">
        <div class="section-row">
          <blockquote class="blockquote">
            <p>Your work is going to fill a large part of
your life, and the only way to be truly satisfied is to do what you believe is great work. And
the only way to do great work is to love what you do. If you haven't found it yet, keep looking.
Don't settle. As with all matters of the heart, you'll know when you find it.</p>
            <footer class="blockquote-footer">Steve
Jobs</footer>
          </blockquote>
```

```

        </div>
    </div>
    <div class="col-md-7">
        <div class="section-row">
            <div class="section-title">
                <h2 class="title">About author</h2>
            </div>
            <p>The author of the "Travel diary" blog, which is
part of the prerequisites for the subject Web programming, is a graduate of the Higher ICT
School in Belgrade, course Internet technology.

                He graduated from the Radoje Ljubičić
Technical High School in Uzice, course Architectural Engineering. In addition to web
programming, he is involved in design, acting and directing. He enrolled in college in the
2016/2017 school year.</p>
            </div>
        </div>
    </div>
    <!-- /row -->
</div>
<!-- /container -->
</div>
<!-- /SECTION -->

```

author.component.ts

```

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

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

```

contact.component.html

```

<!--
SECTION
-->

<div class="section">
    <!-- container -->
    <div class="container">
        <!-- row -->
        <div class="row">

```

```

<div class="col-md-8">
  <div class="section-row">
    <div class="section-title">
      <h2 class="title">Mail us</h2>
    </div>
    <form [formGroup]="contactForm" class="post-reply" (ngSubmit)="onSubmit()">
      <div class="row">
        <div class="col-md-12">
          <div class="form-group" [class.has-
error]="!contactForm.get('subject').valid && contactForm.get('subject').touched" [class.has-
success]="contactForm.get('subject').valid" >
            <input class="input" name="subject" id="subject"
formControlName="subject" placeholder="Subject">
            <span *ngIf="!contactForm.get('subject').valid &&
contactForm.get('subject').touched" class="error-message">
              <span
*ngIf="contactForm.get('subject').errors['maxlength']">The subject is too long ( 3 to 30
characters)</span>
              <span
*ngIf="contactForm.get('subject').errors['minlength']">The subject is too short ( 3 to 30
characters)</span>
              <span
*ngIf="contactForm.get('subject').errors['required']">The subject field is required</span>
            </span>
          </div>
        </div>
        <div class="col-md-12">
          <div class="form-group" [class.has-
error]="!contactForm.get('message').valid && contactForm.get('message').touched" [class.has-
success]="contactForm.get('message').valid" >
            <textarea class="input" name="message" id="message"
formControlName="message" placeholder="Message"></textarea>
            <span *ngIf="!contactForm.get('message').valid &&
contactForm.get('message').touched" class="error-message">
              <span
*ngIf="contactForm.get('message').errors['maxlength']">The message is too long ( 10 to 255
characters)</span>
              <span
*ngIf="contactForm.get('message').errors['minlength']">The message is too short ( 10 to 255
characters)</span>
              <span
*ngIf="contactForm.get('message').errors['required']">The message field is required</span>
            </span>
          </div>
        </div>
        <div class="col-md-6">
          <div class="form-group" [class.has-
error]="!contactForm.get('name').valid && contactForm.get('name').touched" [class.has-
success]="contactForm.get('name').valid" >
            <input class="input" name="name" id="name"
formControlName="name" placeholder="Name">

```

```

        <span *ngIf="!contactForm.get('name').valid &&
contactForm.get('name').touched" class="error-message">
            <span
*ngIf="contactForm.get('name').errors['maxlength']">The name is too long ( 3 to 30
characters)</span>
            <span
*ngIf="contactForm.get('name').errors['minlength']">The name is too short ( 3 to 30
characters)</span>
            <span
*ngIf="contactForm.get('name').errors['required']">The name field is required</span>
        </span>
    </div>
</div>
<div class="col-md-6">
    <div class="form-group" [class.has-
error]="!contactForm.get('email').valid && contactForm.get('email').touched" [class.has-
success]="contactForm.get('email').valid" >
        <input class="input" type="email" name="email"
id="email" formControlName="email" placeholder="Email">
        <span *ngIf="!contactForm.get('email').valid &&
contactForm.get('email').touched" class="error-message">
            <span
*ngIf="contactForm.get('email').errors['email']">The email is not in good format (
example@example.com )</span>
            <span
*ngIf="contactForm.get('email').errors['required']">The email field is required</span>
        </span>
    </div>
</div>
<div class="col-md-12">
    <button type="submit" class="primary-button">Send</button>
</div>
</div>
</form>
</div>
</div>
<div class="col-md-4">
    <!-- contact informations -->
    <div class="section-row">
        <div class="section-title">
            <h2 class="title">Contact Information</h2>
        </div>
        <ul class="contact">
            <li><i class="fa fa-phone"></i>+381 11 234567</li>
            <li><i class="fa fa-envelope"></i> <a
href="#">traveldiary@email.com</a></li>
            <li><i class="fa fa-map-marker"></i>Nemanjina 45, Belgrade</li>
        </ul>
    </div>
    <!-- /contact informations -->
</div>

```

```

        </div>
        <!-- /row -->
    </div>
    <!-- /container -->
</div>
<!-- /SECTION -->

```

contact.component.ts

```

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

import { FormGroup, FormControl, Validators } from '@angular/forms';
import { Router, ActivatedRoute, ParamMap } from '@angular/router';

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

  constructor() { }

  ngOnInit() {
    this.contactForm = new FormGroup({
      'message' : new FormControl(null, [Validators.required,
Validators.minLength(10), Validators.maxLength(255)]),
      'subject' : new FormControl(null, [Validators.required, Validators.minLength(3),
Validators.maxLength(30)]),
      'name' : new FormControl(null, [Validators.required, Validators.minLength(3),
Validators.maxLength(30)]),
      'email' : new FormControl(null, [Validators.required, Validators.email])
    });
  }

  onSubmit() {
    if (this.contactForm.valid) {
      console.log(
        "Your message was successfully sent.\n\n" +
        "Subject:" + this.contactForm.value.subject + "\n" +
        "Message:" + this.contactForm.value.message + "\n" +
        "Name:" + this.contactForm.value.name + "\n" +
        "Email:" + this.contactForm.value.email
      );
    }
  }
}

```

destination.component.html

```
<!--  
SECTION  
-->  
  
<div class="section" *ngIf="loaded">  
  <!-- container -->  
  <div class="container">  
    <!-- row -->  
    <div class="row">  
      <div class="col-md-12">  
        <!-- post content -->  
        <div class="section-row">  
          <figure>  
              
          </figure>  
          <h1>{{ destination?.title }}</h1>  
          <ul class="post-meta">  
            <li>{{destination?.date | date: 'mediumDate'}}</li>  
            <li><div class="post-tags">  
              <ul>  
                <li>  
                  <a  
routerLink="/destinations/category/{{destination?.category}}">{{ destination?.category }}</a>  
                </li>  
              </ul>  
            </div>  
          </li>  
        </ul>  
        <p>{{destination?.description}}</p>  
      </div>  
      <!-- /post content -->  
      <!-- post tags -->  
      <div class="section-row">  
        <div class="post-tags">  
          <ul>  
            <li>Country: </li>  
            <li>  
              <a  
routerLink="/destinations/category/{{destination?.category}}">{{ destination?.category }}</a>  
            </li>  
          </ul>  
        </div>  
        <div class="footer-widget">  
          <div class="tags-widget">  
            <ul>  
              <li *ngFor="let category of categories">  
                <a class="categories-link"  
routerLink="/destinations/category/{{category.path}}">{{category.name}}</a>  
              </li>  
            </ul>  
          </div>  
        </div>  
      </div>  
    </div>  
  </div>  
</div>
```

```

        </ul>
      </div>
    </div>
  </div>
  <!-- /post tags -->
  <!-- post comments -->
  <div class="section-row" *ngIf="commentsBool">
    <div class="section-title">
      <h3 class="title">Comments</h3>
    </div>
    <div class="post-comments">
      <!-- comment -->
      <div class="media" *ngFor="let comment of commentsOfThisId | sortDesc:
'created'">

        <div class="media-left">
          

        </div>
        <div class="media-body">
          <div class="media-heading">
            <h4>{{comment.name}}</h4>
            <span class="time">{{comment.created | date: 'medium'
}}</span>

          </div>
          <p>{{comment.message}}</p>
        </div>
      </div>
    </div>
    <!-- /comment -->
  </div>
</div>
<div class="section-row" *ngIf="!commentsBool">
  <div class="section-title">
    <h3 class="title">No comments</h3>
  </div>
</div>
<!-- /post comments -->
<!-- post reply -->
<div class="section-row">
  <div class="section-title">
    <h3 class="title">Leave a comment</h3>
  </div>
  <form [formGroup]="addCommentForm" class="post-reply"
(ngSubmit)="onSubmit()">
    <div class="row">
      <div class="col-md-12">
        <div class="form-group" [class.has-
error]="!addCommentForm.get('message').valid && addCommentForm.get('message').touched"
[class.has-success]="addCommentForm.get('message').valid" >
          <textarea class="input" name="message" id="message"
formControlName="message" placeholder="Message"></textarea>
          <span *ngIf="!addCommentForm.get('message').valid &&

```



```

addCommentForm.get('message').touched" class="error-message">
    <span
*ngIf="addCommentForm.get('message').errors['maxlength']">The message is too long ( 10 to 255
characters)</span>
    <span
*ngIf="addCommentForm.get('message').errors['minlength']">The message is too short ( 10 to 255
characters)</span>
    <span
*ngIf="addCommentForm.get('message').errors['required']">The message field is required</span>
    </span>
</div>
</div>
<div class="col-md-6">
    <div class="form-group" [class.has-
error]="!addCommentForm.get('name').valid && addCommentForm.get('name').touched" [class.has-
success]="addCommentForm.get('name').valid" >
        <input class="input" name="name" id="name"
formControlName="name" placeholder="Name">
        <span *ngIf="!addCommentForm.get('name').valid &&
addCommentForm.get('name').touched" class="error-message">
            <span
*ngIf="addCommentForm.get('name').errors['maxlength']">The name is too long ( 3 to 30
characters)</span>
            <span
*ngIf="addCommentForm.get('name').errors['minlength']">The name is too short ( 3 to 30
characters)</span>
            <span
*ngIf="addCommentForm.get('name').errors['required']">The name field is required</span>
            </span>
        </div>
    </div>
<div class="col-md-6">
    <div class="form-group" [class.has-
error]="!addCommentForm.get('email').valid && addCommentForm.get('email').touched" [class.has-
success]="addCommentForm.get('email').valid" >
        <input class="input" type="email" name="email"
id="email" formControlName="email" placeholder="Email">
        <span *ngIf="!addCommentForm.get('email').valid &&
addCommentForm.get('email').touched" class="error-message">
            <span
*ngIf="addCommentForm.get('email').errors['email']">The email is not in good format (
example@example.com )</span>
            <span
*ngIf="addCommentForm.get('email').errors['required']">The email field is required</span>
            </span>
        </div>
    </div>
<div class="col-md-12">
    <button type="submit" class="primary-button">Submit</button>
</div>
</div>

```

```

        </form>
    </div>
    <!-- /post reply -->
</div>

</div>
<!-- /row -->
</div>
<!-- /container -->
</div>
<!-- /SECTION -->
<app-not-found *ngIf="!loaded"></app-not-found>

```

destination.component.ts

```

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

import { DestinationsService } from '../services/destinations.service';
import { FormGroup, FormControl, Validators, FormArray } from '@angular/forms';
import { Router, ActivatedRoute, ParamMap } from '@angular/router';
import { NavService } from '../services/nav.service';
@Component({
  selector: 'app-destination',
  templateUrl: './destination.component.html',
  styleUrls: ['./destination.component.css']
})
export class DestinationComponent implements OnInit {
  // Id of this post
  public id;
  //Get Destination
  destinations;
  public destination: any;
  loaded = false;
  param;
  //Get Comments
  comments: Object;
  public commentsOfThisId = [];
  commentsBool = false;
  //Add comment Form
  addCommentForm: FormGroup;
  public lastCommentId;
  //Get Categories
  categories: Object;
  constructor(private route: ActivatedRoute, private service : DestinationsService,
  private router : Router, private navservice : NavService) {
    this.route.params.subscribe(params => {this.param = params['id'];});
    //console.log(this.param);
  }

```

```

ngOnInit() {
  //Get Id
  this.route.paramMap.subscribe((params: ParamMap) => {
    let id = parseInt(params.get('id'));
    this.id = id;
    this.router.navigate(['/destination/' + this.id]);
  });
  //Get Destination
  this.service.getDestinations().subscribe(
    data => {
      this.destinations = data;
      //console.log(data); //svi
      var count = Object.keys(this.destinations).length;
      //console.log(count);
      for(var i = 0; i < count; i++){
        if(this.param == this.destinations[i].id){
          this.destination = this.destinations[i];
          this.loaded = true;
        }
      }
    },
    error=>{
      console.log(error);
    }
  );
  //Get Categories
  this.navservice.getCategories().subscribe( categories => { this.categories =
categories; });
  //Get Comments
  this.service.getComments().subscribe(
    data => {
      this.comments = data;
      //console.log(data); //svi
      var count = Object.keys(this.comments).length;
      //console.log(count);

      this.lastCommentId = this.comments[count - 1].id;
      //console.log(this.lastCommentId + " zadnji id"); //zadnji id
      for(var i = 0; i < count; i++){
        if(this.param == this.comments[i].postId){
          this.commentsOfThisId.push(this.comments[i]);
          this.commentsBool = true;
        }
      }
    },
    error=>{
      console.log(error);
    }
  );
  //Add Comment

```

```

    let date = new Date();
    let currentDate = date.toISOString().slice(0, 10);
    let idComment = parseInt(this.lastCommentId) + 1;
    this.addCommentForm = new FormGroup({
      'message' : new FormControl(null, [Validators.required,
Validators.minLength(10), Validators.maxLength(255)]),
      'name' : new FormControl(null, [Validators.required, Validators.minLength(3),
Validators.maxLength(30)]),
      'email' : new FormControl(null, [Validators.required, Validators.email]),
      'created' : new FormControl(currentDate),
      'postId' : new FormControl(this.id),
      'id' : new FormControl(idComment)
    });
  }

  onSubmit() {
    if (this.addCommentForm.valid) {
      //this.service.saveComment(this.addCommentForm.value);
      //this.router.navigate(['/destination/' + this.id], {state: { added: true,
subscribed: false }});
      //location.reload();
      console.log(
        "Message: " + this.addCommentForm.value.message + "\n" +
        "Name: " + this.addCommentForm.value.name + "\n" +
        "Email : " + this.addCommentForm.value.email + "\n" +
        "Created : " + this.addCommentForm.value.created + "\n" +
        "Post: " + this.addCommentForm.value.postId + "\n" +
        "Id: " + this.addCommentForm.value.id
      );
    }
  }
}

```

destinations.component.html

```

<!--
SECTION
main --
>

<div class="section">
  <!-- container -->
  <div class="container">
    <!-- row -->
    <div class="row">
      <div class="col-md-8">
        <!-- row -->
        <div class="row">

          <!-- post -->

```

```

<div *ngIf="!filterBool">
  <div class="col-md-12">
    <div class="section-title">
      <h2 class="title">Countries</h2>
    </div>
  </div>
  <div class="cleaner"></div>
  <div class="footer-widget" @slide>
    <div class="tags-widget">
      <ul>
        <li *ngFor="let category of categories">
          <a class="categories-link"
routerLink="/destinations/category/{{category.path}}">{{category.name}}</a>
        </li>
      </ul>
    </div>
  </div>
  <div class="col-md-12">
    <div class="section-title">
      <h2 class="title">Destinations</h2>
    </div>
  </div>
  <div class="cleaner"></div>
  <div class="col-md-6" *ngFor="let dest of destinations | sortDesc: 'date'">
    <div class="post">
      <a class="post-img" routerLink="/destination/{{dest.id}}"></a>
      <div class="post-body">
        <div class="post-category">
          <a
routerLink="/destinations/category/{{dest.category}}">{{dest.category}}</a>
        </div>
        <h3 class="post-title"><a
routerLink="/destination/{{dest.id}}">{{dest.title | shorten:40}}</a></h3>
        <ul class="post-meta">
          <li>{{dest.date | date: 'mediumDate'}}</li>
        </ul>
      </div>
    </div>
  </div>
</div>
<div *ngIf="filterBool">
  <div class="col-md-12">
    <div class="section-title">
      <h2 class="title">Destinations in {{category}}</h2>
    </div>
  </div>
  <div class="cleaner"></div>
  <div class="col-md-6" *ngFor="let dest of filteredDestinations | sortDesc:
'date'">
    <div class="post">

```

```

        <a class="post-img" routerLink="/destination/{{dest.id}}"></a>
        <div class="post-body">
            <div class="post-category">
                <a
routerLink="/destinations/category/{{dest.category}}">{{dest.category}}</a>
            </div>
            <h3 class="post-title"><a
routerLink="/destination/{{dest.id}}">{{dest.title | shorten:40}}</a></h3>
            <ul class="post-meta">
                <li>{{dest.date | date: 'mediumDate'}}</li>
            </ul>
        </div>
    </div>
</div>
<!-- /post -->
</div>
<!-- /row -->
</div>

<!--Side-->
<app-aside></app-aside>
<!--Side-->
</div>
<!-- /row -->
</div>
<!-- /container -->
</div>
<!-- /SECTION main -->

```

destinations.component.ts

```

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

import { DestinationsService } from '../services/destinations.service';
import { Router, ActivatedRoute, ParamMap } from '@angular/router';
import { NavService } from '../services/nav.service';
import { slide } from '../animations';
@Component({
    selector: 'app-destinations',
    templateUrl: './destinations.component.html',
    styleUrls: ['./destinations.component.css'],
    animations: [
        slide
    ]
})
export class DestinationsComponent implements OnInit {

```

```

public category;
public destinations;
public filteredDestinations = [];
filterBool = false;
param;
//Get Categories
categories: Object;
constructor(private route: ActivatedRoute, private service : DestinationsService,
private router : Router, private navservice : NavService) {
    this.route.params.subscribe(params => {this.param = params['category'];});
    //console.log(this.param);
}
ngOnInit() {
    //this.service.getDestinations().subscribe( destinations => { this.destinations =
destinations; });
    this.route.paramMap.subscribe((params: ParamMap) => {
        let category = params.get('category');
        this.category = category;
    });
    this.service.getDestinations().subscribe(
        data => {
            this.destinations = data;
            //console.log(data);
            var count = Object.keys(this.destinations).length;
            //console.log(count);
            for(var i = 0; i < count; i++){
                if(this.param === this.destinations[i].category){
                    this.filteredDestinations.push(this.destinations[i]);
                    this.filterBool = true;
                }
            }
        },
        error=>{
            console.log(error);
        }
    );
    //Get Categories
    this.navservice.getCategories().subscribe( categories => { this.categories =
categories; });
}
}

```

footer.component.html

```

<footer
id="footer">

    <!-- container -->
    <div class="container">
        <!-- row -->

```

```

<div class="row">
  <div class="col-md-6">
    <div class="footer-widget">
      <div class="footer-logo">
        <a routerLink="/" class="logo"></a>
      </div>
      <ul class="contact-social">
        <li><a href="https://www.facebook.com/"
class="social-facebook"><i class="fa fa-facebook"></i></a></li>
        <li><a href="https://www.twitter.com/"
class="social-twitter"><i class="fa fa-twitter"></i></a></li>
        <li><a href="https://www.google.com/"
class="social-google-plus"><i class="fa fa-google-plus"></i></a></li>
        <li><a
href="https://www.instagram.com/" class="social-instagram"><i class="fa fa-
instagram"></i></a></li>
      </ul>
    </div>
  </div>
  <div class="col-md-6">
    <div class="footer-widget">
      <h3 class="footer-title">Destinations</h3>
      <div class="tags-widget">
        <ul>
          <li *ngFor="let category of
categories">
            <a
routerLink="destinations/category/{{category.path}}">{{category.name}}</a>
          </li>
        </ul>
      </div>
    </div>
  </div>
</div>
<!-- /row -->
<!-- row -->
<div class="footer-bottom row">
  <div class="col-md-6 col-md-push-6">
    <ul class="footer-nav">
      <li *ngFor="let page of pages">
        <a
routerLink="{{page.path}}">{{page.title}}</a>
      </li>
    </ul>
  </div>
  <div class="col-md-6 col-md-pull-6">
    <div class="footer-copyright">
      Copyright &copy; All rights reserved | <i class="fa fa-code" aria-hidden="true"></i>
      Adapted by: Marko Mutavdzic 125/16
    </div>
  </div>
</div>

```



```

        </div>
    </div>
    <!-- /row -->
</div>
<!-- /container -->
</footer>

```

footer.component.ts

```

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

import { NavService } from '../services/nav.service';
@Component({
  selector: 'app-footer',
  templateUrl: './footer.component.html',
  styleUrls: ['./footer.component.css']
})
export class FooterComponent implements OnInit {
  pages: Object;
  categories: Object;
  constructor(private service : NavService) {
  }
  ngOnInit() {
    this.service.getPages().subscribe( pages => { this.pages = pages; });
    this.service.getCategories().subscribe( categories => { this.categories =
categories; });
  }
}

```

home.component.html

```

<!--
SECTION
hot -->

<div class="section">
  <!-- container -->
  <div class="container">
    <!-- row HOME HOT POST-->
    <div id="hot-post" class="row hot-post">
      <div class="col-md-8 hot-post-left">
        <!-- post -->
        <div class="post post-thumb" *ngFor="let dest of destinations | sortDesc:
'date' | slice:0:2">
          <a class="post-img" routerLink="/destination/{{dest.id}}"></a>
          <div class="post-body">
            <div class="post-category">
              <a

```

```

routerLink="/destinations/category/{{dest.category.path}}">{{dest.category.name}}</a>
    </div>
    <h3 class="post-title title-lg"><a
routerLink="/destination/{{dest.id}}">{{dest.title}}</a></h3>
    <ul class="post-meta">
        <li>{{dest.date | date: 'mediumDate'}}</li>
    </ul>
</div>
</div>
<!-- /post -->
</div>
<div class="col-md-4 hot-post-right">

    <!-- post -->
    <div class="post post-thumb" *ngFor="let dest of destinations | sortDesc:
'date' | slice:2:6">
        <a class="post-img" routerLink="/destination/{{dest.id}}"></a>
        <div class="post-body">
            <div class="post-category">
                <a
routerLink="/destinations/category/{{dest.category.path}}">{{dest.category.name}}</a>
            </div>
            <h3 class="post-title"><a
routerLink="/destination/{{dest.id}}">{{dest.title}}</a></h3>
            <ul class="post-meta">
                <li>{{dest.date | date: 'mediumDate'}}</li>
            </ul>
        </div>
    </div>
<!-- /post -->
</div>
</div>
<!-- /row HOME HOT POST-->
</div>
<!-- /container -->
</div>
<!-- /SECTION hot -->

```

home.component.ts

```

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

import { DestinationsService } from '../services/destinations.service';
@Component({
    selector: 'app-home',
    templateUrl: './home.component.html',
    styleUrls: ['./home.component.css']

```

```

    })
    export class HomeComponent implements OnInit {
        destinations: Object;
        constructor(private service : DestinationsService) {
        }
        ngOnInit() {
            this.service.getDestinations().subscribe( destinations => { this.destinations =
destinations; });
        }
    }
}

```

nav.component.html

```

<header
id="header">

```

```

        <!-- NAV -->
        <div id="nav">
            <!-- Top Nav -->
            <div id="nav-top">
                <div class="container">
                    <!-- social -->
                    <ul class="nav-social">
                        <li><a href="https://www.facebook.com/"><i
class="fa fa-facebook"></i></a></li>
                        <li><a href="https://www.twitter.com/"><i
class="fa fa-twitter"></i></a></li>
                        <li><a href="https://www.google.com/"><i
class="fa fa-google-plus"></i></a></li>
                        <li><a href="https://www.instagram.com/"><i
class="fa fa-instagram"></i></a></li>
                    </ul>
                    <!-- /social -->
                    <!-- logo -->
                    <div class="nav-logo">
                        <a routerLink="/" class="logo"></a>
                    </div>
                    <!-- /logo -->
                    <!-- search & aside toggle -->
                    <div class="nav-btns">
                        <button class="aside-btn"><i class="fa fa-
bars"></i></button>
                    </div>
                    <!-- /search & aside toggle -->
                </div>
            </div>
            <!-- /Top Nav -->
            <!-- Main Nav -->
            <div id="nav-bottom">
                <div class="container">

```

```

<!-- nav -->
<ul class="nav-menu">
    <li *ngFor="let page of pages" class="has-
dropdown">
        <a
routerLink="{{page.path}}">{{page.title}}</a>
    </li>
</ul>
<!-- /nav -->
</div>
</div>
<!-- /Main Nav -->
<!-- Aside Nav -->
<div id="nav-aside">
    <ul class="nav-aside-menu">
        <li *ngFor="let page of pages">
            <a
routerLink="{{page.path}}">{{page.title}}</a>
        </li>
    </ul>
    <button class="nav-close nav-aside-
close"><span></span></button>
</div>
<!-- /Aside Nav -->
</div>
<!-- /NAV -->
</header>

```

nav.component.ts

```

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

import { NavService } from '../services/nav.service';
@Component({
    selector: 'app-nav',
    templateUrl: './nav.component.html',
    styleUrls: ['./nav.component.css']
})
export class NavComponent implements OnInit {
    //variables
    pages: Object;
    categories: Object;
    constructor(private service : NavService ) {

    }
    ngOnInit() {
        this.service.getPages().subscribe( pages => { this.pages = pages; });
    }
}

```

```
}
```

not-found.component.html

```
<div
id="notfound"
@fade>

  <div class="notfound">
    <div class="notfound-404">
      <h1>:</h1>
    </div>
    <h2>404 - Page not found</h2>
    <p>The page you are looking for might have been removed had its name changed or
is temporarily unavailable.</p>
    <a routerLink="/">Back to Home Page</a>
  </div>
</div>
```

not-found.component.ts

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

import { fade } from '../animations';
@Component({
  selector: 'app-not-found',
  templateUrl: './not-found.component.html',
  styleUrls: ['./not-found.component.css'],
  animations: [
    fade
  ]
})
export class NotFoundComponent implements OnInit {
  constructor() { }
  ngOnInit() {
  }
}
```

pipes

shorten.pipe.ts

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

@Pipe({
  name: 'shorten'
})
```

```

export class ShortenPipe implements PipeTransform {
  transform(value: any, limit: number){
    if(value.length > limit){
      return value.substr(0, limit) + "...";
    }
    return value;
  }
}

```

sort-asc.pipe.ts

```

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

@Pipe({
  name: 'sortAsc'
})
export class SortAscPipe implements PipeTransform {
  transform(array: any, field: string): any[] {
    if (!Array.isArray(array)) {
      return;
    }
    array.sort((a: any, b: any) => {
      if (a[field] < b[field]) {
        return -1;
      } else if (a[field] > b[field]) {
        return 1;
      } else {
        return 0;
      }
    });
    return array;
  }
}

```

sortDesc.pipe.ts

```

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

@Pipe({
  name: 'sortDesc'
})
export class SortDescPipe implements PipeTransform {
  transform(array: any, field: string): any[] {
    if (!Array.isArray(array)) {
      return;
    }
  }
}

```

```

        array.sort((a: any, b: any) => {
            if (a[field] > b[field]) {
                return -1;
            } else if (a[field] < b[field]) {
                return 1;
            } else {
                return 0;
            }
        });
        return array;
    }
}

```

services

destinations.service.ts

```

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

import { HttpClient } from '@angular/common/http';
@Injectable()
export class DestinationsService {
    servicePath = "http://localhost:4200/assets/json/";
    constructor(private service : HttpClient){
    }
    getDestinations(){
        var destinations = this.service.get(this.servicePath + "destinations.json");
        return destinations;
    }
    getComments() {
        var comments = this.service.get(this.servicePath + "comments.json");
        return comments;
    }
    saveComment(data) {
        this.service.post(this.servicePath + 'comments.json',
data).subscribe(responseData => {});
    }
}

```

nav.service.ts

```

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

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

```

```

export class NavService {
  servicePath = "http://localhost:4200/assets/json/";
  constructor(private service : HttpClient){
  }
  getPages(){
    var pages = this.service.get(this.servicePath + "pages.json");
    return pages;
  }
  getCategories(){
    var categories = this.service.get(this.servicePath + "categories.json");
    return categories;
  }
}

```

animations.ts

```

import { trigger,
transition, style,
animate } from
'@angular/animations';

export let fade = trigger('fade', [
  transition('void => *',[
    style({ opacity: 0}),
    animate(1000)
  ])
])
export let slide = trigger('slide', [
  transition(':enter', [
    style({transform: 'translateX(-100px)'}),
    animate(1000)
  ]),
  transition(':leave', [
    animate(1000, style({transform: 'translateX(-100%)'}))
  ])
]);

```

assets/json

categories.json

```

[
  {
    "id": 1,
    "name": "Serbia",
    "path": "serbia"
  },
  {
    "id" : 2,
    "name" : "Italy",
    "path": "italy"
  }
]

```



```

    },
    {
        "id": 3,
        "name" : "England",
        "path": "england"
    },
    {
        "id": 4,
        "name" : "France",
        "path": "france"
    },
    {
        "id": 5,
        "name" : "Spain",
        "path": "spain"
    }
]

```

comments.json

```

[
    {
        "id": 1,
        "message": "I attended this event last year, it's wonderful. Greetings!",
        "name": "John",
        "email": "john@gmail.com",
        "created": "2019-12-10T18:25:43",
        "postId": 7
    },
    {
        "id": 2,
        "message": "Manchester is beautiful in winter ...",
        "name": "maria",
        "email": "maria@gmail.com",
        "created": "2019-12-10T18:26:43",
        "postId": 7
    },
    {
        "id": 4,
        "message": "Novi Sad, najlepsi grad!",
        "name": "Zile",
        "email": "zile@gmail.com",
        "created": "2019-12-10T18:26:43",
        "postId": 6
    }
]

```

destinations.json

```

[

```

```

{
  "id": 1,
  "image": "rome-1.jpg",
  "title": "The Good and Bad of Rome: A Quick Overview",
  "description": "No city is perfect – which is why we like to talk about some of the good
and the not-so-good aspects of Rome. The Good: The Atmosphere and History: Rome has more
spectacular museums, churches, squares, fountains, and ancient ruins per square foot than just
about anywhere else in the world. Everywhere you go, you’ll see something amazing. Art: Rome
certainly has no shortage of art and museums – from contemporary to ancient. Good Food: You
don’t have to spend a fortune to eat well in Rome. All it takes is a little research! Walking
The Streets and Side Streets: Rome has been growing and changing for over 3,000 years so its
streets also grew organically – which means lots of little alleys and side streets that are a joy
to explore. For once, you’ll love getting lost. The Nightlife: Like much of Southern Europe,
Italians eat late and stay up even later. This means that restaurants, bars, and cafés are open
late into the night (and then people head to the clubs even later). The Not-So-Good Tourist
Hoards: Tourists everywhere! Rome attracts millions of visitors each year, so expect to be
surrounded by other people wanting to see all the sights. Expect long lines for most attractions –
try to book tickets in advance so you can skip the multi-hour lines. Summer is the busiest time of
the year, so try visiting in the spring or fall to help escape the crowds. Pickpockets: Rome
is known for being one of the worst cities, if not the worst city when it comes to pickpockets and
other petty theft. It’s the worst at the Termini Station (Rome’s main train station), on the
metro, and on buses. Be extra vigilant at all times. If you can, try not to carry anything
valuable. Tourist Restaurants: Rome has a ton of amazing restaurants serving great food at
fair prices. It also has just as many crappy restaurants serving overpriced junk to unknowing
visitors. Do a little research and try to go a few streets away from the tourist sites and you’ll
find plenty of great options. Dirty: Rome is pretty dirty. I mean, it’s a big city so it
shouldn’t be too surprising. But a lot of people are surprised to see trash and graffiti all over
the place. Hot Summers: The temperatures in Rome can be fairly brutal in the middle of summer.
Many hostels and hotels don’t have AC so that can pose a problem for some people.",
  "category": "italy",
  "date": "2019-12-09"
},
{
  "id": 2,
  "image": "paris-1.jpg",
  "title": "The Best Things To Do In Paris | Over 50 Unforgettable Experiences",
  "description": "We lived in Paris for nearly two years so we know first-hand that there are
so many things to do in Paris. In fact, Paris is jam-packed with so many museums, sights,
entertainment, shopping, cool neighborhoods, and restaurants that it’s hard to narrow it all down.
But fear not because we’re here to help show you what to do in Paris! If you only visit one museum
in Paris we recommend you make it the Musee d’Orsay. It might not be as grand or famous as the
Louvre, but it has an excellent (and manageable) collection of art with a heavy emphasis on
impressionist paintings from Renoir, Monet, van Gogh, and other masters. Furthermore, the Musee
d’Orsay is housed in an old art nouveau train station so it makes a beautiful space to view art.
The d’Orsay is one of the most popular things to do in Paris so it does get busy. We recommend
getting there early or you can visit later in the day on Thursdays because they have extended
hours (9:45 pm). The Musee du Louvre is one of the biggest and most famous museums in the world.
This place is insane so don’t even try “seeing it all” because your brain will melt from museum
overload. The collection covers art and artifacts that span thousands of years including Egyptian
antiquities, Greek and Roman sculptures, Islamic art, and paintings from the 13th century to 1848.
Because of its popularity, the Louvre draws huge crowds so it is highly recommended that you

```

arrive early or use a Museum Pass or pre-book your tickets so you can skip the ticket line. You can also visit Wednesday or Friday night because they have extended hours (9:45 pm). Also, if nothing else, we recommend at least walking around the Louvre because the outside of Louvre is beautiful and the grounds are impressive. ",

```
        "category" : "france",
        "date": "2019-12-10"
    },
    {
        "id": 3,
        "image": "london-1.jpg",
        "title": "London Travel Guide | How To Visit London on a Budget ",
        "description": "London is truly one of the world's greatest cities, and you owe it to yourself to spend at least a few days exploring its streets. It's easily one of our favorite places to visit. One of the things that make London is great is that it has just about everything – history out the wazoo, amazing architecture, theater, art and museums, world-class nightlife, excellent shopping, friendly people, top-notch public transportation, and just about anything else you can imagine. Plus, London is the perfect first stop for your journey through Europe because it's an “easy” introduction to Europe. London isn't cheap, so we recommend budgeting £45-£80/day if you're on a backpacker's budget. You can check out our London City Price Guide for a more in-depth cost breakdown, but we recommend over-budgeting to be safe.",
        "category" : "england",
        "date": "2019-12-11"
    },
    {
        "id": 4,
        "image": "cannes-1.jpg",
        "title": "Things to do in Cannes",
        "description": "The port: Cannes flashy port is a place where you can check out all the luxury yachts that's owned by both millionaires and celebrities. Buy an ice cream and stroll along the harbor and admire the luxury yachts! Festival de Cannes: The world's most famous and largest film festival takes place in Cannes every year in the month of May. If you're in town during this festival, you can count on to see a lot of familiar faces since many of the world's most famous persons attend this event! Stroll around Old Town: Above the harbor and the movie palace, you will find the cozy medieval district of Cannes. Between the narrow streets and beautiful old buildings lies several boutiques, excellent restaurants, and cozy cafés. When you're in Cannes old town, you have to stop by the famous Clock Tower, Tour du Suquet, which is a hallmark of Cannes along with Palace de la Castre. Iles de Lerins: les de Lérins is an island group of four that lies about 15 minutes by boat from La Croisette in central Cannes. These pristine islands are all natural and offer a beautiful environment with an unusual history. The two largest islands are Sainte-Marguerite and Saint Honorat. You can smell the scent of pine and eucalyptus while exploring the island of Sainte-Marguerite. This is best done by following the marked passages that lead you to the most scenic and historic sights. The island also offers cozy restaurants, great picnic spots and stunning bays with crystal clear water. This island is perfect for those who want to relax and enjoy the tranquility and beauty of nature. Saint Honorat is an island owned by the French monks. The monks live on this particular island all the year round and are also very active. Here the monks manufacture both liqueurs and wines and also offer delightful wine tasting tours and meditation for those who are interested.",
        "category" : "france",
        "date": "2019-12-12"
    },
    {
```

```

        "id": 5,
        "image": "barcelona-1.jpg",
        "title": "48 Hours in Barcelona",
        "description": "If you want to spend a weekend abroad, Barcelona is an excellent choice, as
it's one of the cultural capitals of Europe. As the home of Gaudi, the city has several
architectural highlights that you simply can't find anywhere else in the world. The city is also a
great place for fashionistas to fill their bags with the latest trends. Thus, there are plenty of
things to do when having 48 hours in Barcelona! It's very worth visiting Barcelona, Spain for a
long weekend to immerse yourself in everything the city has to offer. Trying out everything might
be a tight squeeze, but that has never stopped us from trying. So, let's find out what you should
definitely do during your weekend in Barcelona to get the most out of your vacation. Your 48 hours
in Barcelona start with figuring out the best way to get from airport to the city center of
Barcelona. Be ready for an unforgettable 48 hours in Barcelona! 1. The Plaça Reial is a historic
square in the heart of Barcelona. If you want to dine at a stunning location for an affordable
price, then the Plaça Reial is the place to be. At the Plaça Reial you will find a dozen
restaurants, each with their own cuisine and outdoor terraces. 2. Gaudi has had a strong influence
on the architecture of Spain and this is nowhere more apparent than in Park Güell. Not only was
Park Güell designed by Gaudi, but he also actually lived in it. Taking a stroll through the city
park, while enjoying the unique architecture that perfectly blends in with surrounding nature, is
the perfect way to escape the busy city life of Barcelona. 3. Barcelona is a city that's still
very much alive in the evening hours. Due to the warm climate, the evening can be an excellent
time to go out and enjoy a drink. One attraction that's only available in the evening is the Magic
Fountain, which is a light show in front of the National Palace. 4. In the heart of Barcelona, you
will find the Mercat de la Boqueria, which is a traditional food market. As one of the biggest
fresh food markets in Spain, it's a good place to mingle among the locals looking to get some
fresh fruits, nuts or fish. 5. The best way to appreciate the skyline of Barcelona is by taking
the Montjuïc Cable Car. From this cable car, you get panoramic views over the city as you head
upwards to your destination. Where are you going you ask? Well, you're going up the Montjuïc,
which is a hill in front of the coast of Barcelona. At the top, you will find the most defensible
position of Barcelona, Montjuïc Castle. This historical castle has a classic walking bridge, large
walls, bastions and a great view over the city and harbor of Barcelona. 6. The Ramblas is the most
famous street in Barcelona for a good reason. It's a long shopping street in the heart of
Barcelona. It leads from the Plaça de Catalunya all the way to the harbor. 7. If you ever received
a postcard from Barcelona, then chances are high it had the Sagrada Familia on it. The Sagrada
Familia is often jokingly called the church that is never finished, as people have been working on
it for over a century now. In fact, the biggest spire has yet to be finished. ** hope these ideas
have given you some inspiration for a weekend in Barcelona. The great thing about Barcelona is
that there are always more things to discover. Just remember to take it easy and give yourself
some time to take it all in, then you are bound to have a great time in this marvelous city. And I
definitely recommend spending more than 48 hours in Barcelona.  ",
        "category" : "spain",
        "date": "2019-12-13"
    },
    {
        "id": 6,
        "image": "novisad-1.jpg",
        "title": "Welcome to Novi Sad",
        "description": "Novi Sad is a chipper town with all the spoils and none of the stress of
the big smoke. Locals sprawl in pretty parks and outdoor cafes, and laneway bars pack out nightly.
The looming Petrovaradin Fortress keeps a stern eye on proceedings, loosening its tie each July to
host Serbia's largest music festival. You can walk to all of Novi Sad's attractions from the

```

happening pedestrian thoroughfare, Zmaj Jovina, which stretches from the main square (Trg Slobode) to Dunavska. Novi Sad isn't nicknamed the 'Athens of Serbia' for nothing: its history as a vibrant, creative city continues today in its established galleries, alternative music scene and a vibe that's generally more liberal than that of other Serbian cities. Novi Sad is 2019's European Youth Capital, and in 2021, it will become the first non-EU city to spend a year with the prestigious title of European Capital of Culture.

1. Petrovaradin Fortress: Towering over the river on a 40m-high volcanic slab, this mighty citadel, considered Europe's second-biggest fortress (and one of its best preserved), is aptly nicknamed 'Gibraltar on the Danube'. Constructed using slave labour between 1692 and 1780, its dungeons have held notable prisoners including Karađorđe (leader of the first Serbian uprising against the Turks and founder of a royal dynasty) and Yugoslav president Tito. Have a good gawk at the iconic clock tower: the size of the minute and hour hands are reversed so far-flung fisherfolk can tell the time. Within the citadel walls, a museum offers insight into the site's history; it can also arrange tours (in English; 3500RSD) of Petrovaradin's 16km of creepy, but cool, unlit underground tunnels (katakombe). Petrovaradin hosts Novi Sad's wildly popular EXIT Festival each July.

2. Štrand: One of Europe's best by-the-Danube beaches, this 700m-long stretch morphs into a city of its own come summertime, with bars, stalls and all manner of recreational diversions attracting thousands of sun- and fun-seekers from across the globe. It's also the ultimate Novi Sad party venue, hosting everything from local punk gigs to EXIT raves. It's great for kids (watch them by the water: the currents here are strong), with playgrounds, trampolines and dozens of ice-cream and fast-food stalls.",

```

    "category" : "serbia",
    "date": "2019-12-14"
  },
  {
    "id": 7,
    "image": "manchester-1.jpg",
    "title": "Manchester Christmas Markets",
    "description": "Manchester's award-winning Christmas Markets 2019 return this winter offering mouth-watering international food, drinks and delicately crafted Yuletide gifts in the heart of this buzzing city. With more than 300 beautifully festooned stalls and chalets spread across the city centre, the Christmas Markets are free and accessible to all. Soak up the atmosphere and follow the market trail as it winds through the city centre, tempting you with gifts, crafts, jewellery, clothes and toys. European and local producers offer everything from fine amber jewellery, handcrafted leather bags and top-quality bonsai trees, to handmade Belgian kitchenware, framed photo prints and French soaps. And when planning a visit, don't miss the amazing array of food and drink! Manchester's Christmas Markets are famous for traditional bratwurst, but visitors can also treat themselves to Hungarian goulash, Spanish paella or a good old-fashioned hog roast. Plenty of vegetarian options will be on offer, and for those with a sweet tooth the choices are even more mouth-watering. ",
    "category" : "england",
    "date": "2019-12-15"
  }
]

```

pages.json

```

[
  {
    "id": 1,
    "title": "Home",

```

```
        "path": ""
    },
    {
        "id": 2,
        "title": "Destinations",
        "path": "destinations"
    },
    {
        "id": 3,
        "title": "Contact",
        "path": "contact"
    },
    {
        "id": 4,
        "title": "About us",
        "path": "about-us"
    },
    {
        "id": 5,
        "title": "Author",
        "path": "author"
    }
]
```

Visoka škola strukovnih studija za informacione i komunikacione tehnologije

Visoka ICT škola, Beograd, decembar 2019

Predmet Praktikum

student: Marko Mutavdžić 125/16