Akademija tehničko-umetničkih strukovnih Studija

Odsek Visoka ICT škola

Collēctiō

https://nikola-project-418a3.web.app/

Praktikum Web Programiranje   Nikola Gutić 137-19

Beograd, Avgust 2022

Sadržaj

Uvod

Korišćene tehnologije

- HTML
- CSS
- TypeScript
- Angular
- Bootstrap
- Firebase

Okruženja

Za izradu projekta korišten je Visual Studio Code.

Opis projekta

Projekat predstavlja online enciklopediju ili raznovrsne online kolekcije koje korisnici mogu da posjećuju i o kojim mogu da čitaju i nauče nešto novo.

Organizacija

Struktura JSON fajla

```json
"categories": [
  {
    "id": 1,
    "name": "Botanic"
  },
  {
    "id": 2,
    "name": "Animals"
  },
  {
    "id": 3,
    "name": "Insects"
  },
  {
    "id": 4,
    "name": "History"
  }
],
```

```json
"collectionItems": [
  {
    "id": 1,
    "image": "fungi1.jpg",
    "infos": [
      1,
      2
    ],
    "model": "AmanitaMuscaria",
    "title": "Amanita Muscaria"
  },
  {
    "id": 2,
    "image": "fungi2.jpg",
    "infos": [
      4,
      5
    ],
    "model": "AmanitaRubescens",
    "title": "Amanita Rubescens"
  },
```

```json
"collections": [
  {
    "backgroundImage": "banner1.jpg",
    "categoryId": 1,
    "id": 1,
    "image": "fungi1.jpg",
    "items": [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12],
    "title": "Mushrooms",
    "userId": 1
  },
  {
    "backgroundImage": "banner2.jpg",
    "categoryId": 3,
    "id": 2,
    "image": "insect7.jpg",
    "items": [ ⋯
    ],
    "title": "Butterflies",
    "userId": 2
```

```json
"itemInfos": [
  {
    "content": "A saprobic species, Lycope
    grow in fairy rings. Typical habitats
    in Tamil Nadu, India. The puffball som
    "id": 1,
    "title": "First Info"
  },
  {
    "content": "Aglia tau, the tau emperor
    Pacific coast. The species was first d
    "id": 2,
    "title": ""
  },
```

```json
"users": [
  {
    "email": "nikola@gmail.com",
    "firstName": "Nikola",
    "id": 1,
    "lastName": "Gutic"
  },
  {
    "email": "ana@gmail.com",
    "firstName": "Ana",
    "id": 2,
    "lastName": "Ana"
  }
]
```

Organizaciona šema
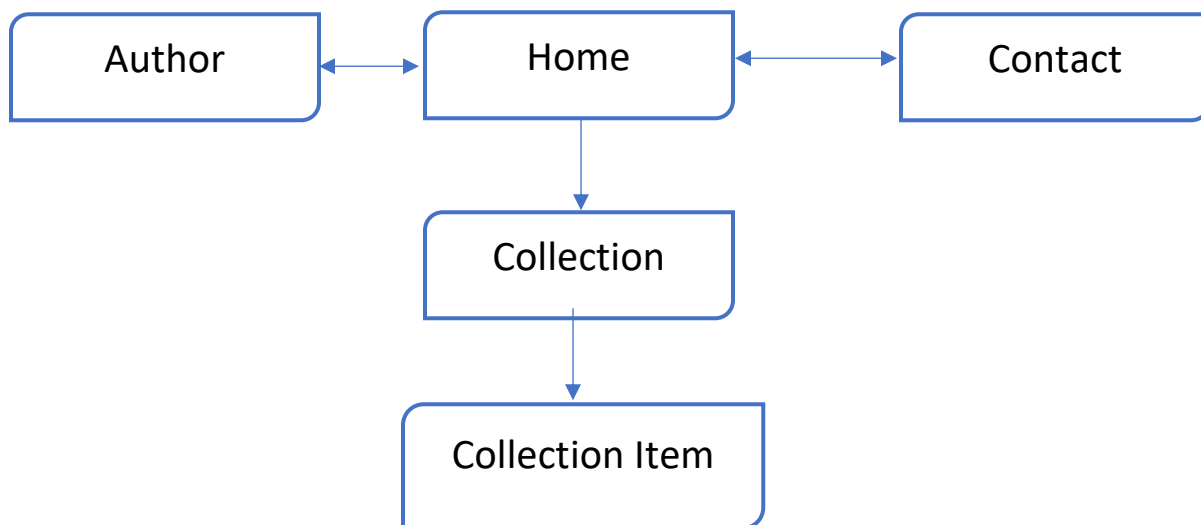
Struktura projekta

App
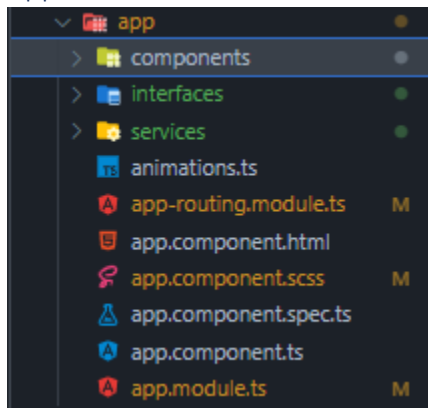
Components
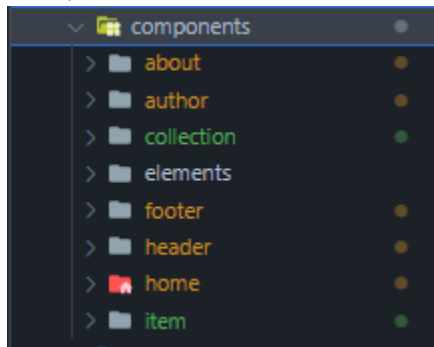
Interfaces
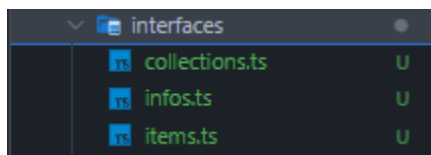
Services
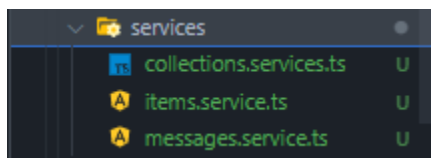
Opis i slike stranica

Home

Na Home stranici prikazane su sve kolekcije, klikom na neku od kolekcija otvara se nova stranica sa svim njenim elementima.

Collection

Na stranici kolekcije prikazani su autor, naziv kolekcije i svi njeni elementi, klikom na neki od elemenata prikazuje se detaljniji prikaz.

Collection item

Element kolekcije može sadržati dodatne informacije kao i 3D model.

Contact

Kontakt stranica sadrži formu sa proverom elemenata formulara i obradom grešaka. Ukoliko je validacija prošla dugme za slanje se enable-uje. Slanje mail-a je realizovano pomoću Formspree-a.

Kodovi

Banner component

Html

```
<div class="banner">
    <div class="heading" @fade>
        <h1>collēctiō </h1>
        <h3>The encyclopedia of everything</h3>
        <span class="opis">discover, learn & share your work</span>
    </div>
</div>
```

Scss

```
.banner {
    height: 50vh;
    width: 100%;
```

```scss
    display: flex;
    justify-content: center;

    .heading {
        text-align: center;
        padding-top: 8%;
        font-family: 'Bagerich';
        z-index: 99;

        h1 {
            z-index: 99;
            font-size: 13vw;
        }

        .opis {
            font-family: "Agrandir";
            z-index: 99;
            font-stretch: extra-condensed;
        }
    }

    @media only screen and (max-width: 992px) {
        .heading {
            padding-top: 20vh;
        }
    }
}
```

Ts

```typescript
import { $animations } from 'src/app/animations';
import { Component, OnInit } from '@angular/core';

@Component({
  selector: 'app-banner',
  templateUrl: './banner.component.html',
  styleUrls: ['./banner.component.scss'],
  animations: [$animations]
})
export class BannerComponent implements OnInit {
```

```
  constructor() { }

  ngOnInit(): void {
  }

}
```

Collections component

Html

```
<div class="table-fungi">
    <div class="row">
        <div class="col-12 d-flex flex-wrap justify-content-center">
            <app-single-collection class="col-lg-2 col-md-3 col-12 ms-3 me-3 mb-5
mt-3" *ngFor="let collection of collections | async" [data]="collection"></app-
single-collection>
        </div>
    </div>
</div>
```

Scss

```
.table-fungi{
    margin-top: 100px;
    width: 100%;
    padding: 0px 5%;
    z-index: 60;
}
```

Ts

```
import { CollectionsServices } from './../../../services/collections.services';
import { Component, OnInit } from '@angular/core';
import { Observable } from 'rxjs';
```

```
import { Collections } from 'src/app/interfaces/collections';

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

  collections!: Observable<Collections[]>;
  data!: Collections[];

  constructor(private collectionServices: CollectionsServices) { }

  ngOnInit(): void {
    this.getCollections();
  }

  getCollections(): void {
    this.collections = this.collectionServices.getCollections();
  }
}
```

Single Collection component

Html

```
<div class="fungi-content" [ngStyle]="animateBlock" style="background-image:
url('../../../../../assets/images/{{data.image}}'); background-position: center;
               background-repeat: no-repeat;
               background-size: cover;">
    <div class="layer" [ngStyle]="hoverLayer">
    </div>
    <span class="content">
        <a aria-current="page" [routerLink]="['/collection',
transformUrl(data.title)]" routerLinkActive="active">
            <p class="info" [ngStyle]="animateText">{{data.title}}</p>
        </a>
    </span>
</div>
```

Scss

```scss
.fungi-content {
    position: relative;
    background-position: center;
    background-repeat: no-repeat;
    background-size: cover;
    height: 240px;
    display: flex;
    align-items: center;
    justify-content: center;
    z-index: 60;

    box-shadow: rgba(0, 0, 0, 0.2) 0px 60px 40px -7px;

    .layer {
      position: absolute;
      top: 0;
      left: 0;
      width: 100%;
      height: 100%;
    }

    .content {
      display: flex;
      align-items: center;
      justify-content: center;
      width: 100%;
      .info{
        p{
          width: 100%;
        }
      }
      a {
        width: 100%;
        p {
          font-weight: 300;
          width: 100%;
          z-index: 99;
        }
      }
    }
  }
```

Ts

```ts
import { Component, OnInit, ElementRef, HostListener, Input } from
'@angular/core';

@Component({
  selector: 'app-single-collection',
  templateUrl: './single-collection.component.html',
  styleUrls: ['./single-collection.component.scss']
})
export class SingleCollectionComponent implements OnInit {


  @Input() data!: any;

  isVisible = false;
  brojac!: number;
  interval!: any;

  constructor(public el: ElementRef<HTMLElement>) {
  }

  public transformUrl(data: string): string {
    let newString = data.replace(/\s/g, '-');
    return newString;
  }

  @HostListener('mouseenter')
  onMouseEnter() {
    this.isVisible = true;
    this.brojac = 1;
    clearInterval(this.interval);
    this.interval = setInterval(() => {
      if (this.brojac < 1.3) {
        this.brojac += 0.01
      }
      else {
        clearInterval(this.interval);
      }
    }, 5);

  }
```

```
@HostListener('mouseleave')
onMouseLeave() {
  this.isVisible = false;
  clearInterval(this.interval);
  this.interval = setInterval(() => {
    if (this.brojac > 1) {
      this.brojac -= 0.01
    }
    else {
      clearInterval(this.interval);
    }
  }, 5);
}

ngOnInit(): void {

}

get animateText() {
  if (this.isVisible) {
    return {
      "text-align": "center",
      "animation-name": "fontZoom",
      "animation-duration": "0.5s",
      "animation-fill-mode": "forwards"
    }
  }
  return {
    "text-align": "center",
    "animation-name": "fontOut",
    "animation-duration": "0.5s",
    "animation-fill-mode": "forwards"
  }
}

get animateBlock() {
  const scaler = this.brojac;
  if (this.isVisible) {
    return {
      "transform": "scale(" + scaler + "," + scaler + ")",
      "animation-name": "borderCurver",
      "animation-duration": "0.5s",
      "animation-fill-mode": "forwards",
      "z-index": "10"
    }
```

```
        }
    return {
      "transform": "scale(" + scaler + "," + scaler + ")",
      "animation-name": "borderCurverOut",
      "animation-duration": "0.5s",
      "animation-fill-mode": "forwards",
      "z-index": "5"
    }
  }

  get hoverLayer() {
    if (this.isVisible) {
      return {
        "animation-name": "hoverLayer",
        "animation-duration": "0.5s",
        "animation-fill-mode": "forwards"
      }
    }
    return {
      "animation-name": "hoverLayerExit",
      "animation-duration": "0.10s",
      "animation-fill-mode": "forwards"
    }
  }

  get borderCurver() {
    if (this.isVisible) {
      return {
        "animation-name": "borderCurver",
        "animation-duration": "0.5s",
        "animation-fill-mode": "forwards"
      }
    }
    return {
      "animation-name": "borderCurverOut",
      "animation-duration": "0.5s",
      "animation-fill-mode": "forwards"
    }
  }

}
```

Header component

Html

```html
<nav class="navbar fixed-top  navbar-expand-lg navbar-light bg-light">
  <div class="contain" [ngStyle]="headerOverlay">
    <a class="navbar-brand" routerLink="/author"
routerLinkActive="active">NIKOLAGUTIC</a>
    <button class="navbar-toggler navButton" type="button" data-bs-
toggle="collapse"
      data-bs-target="#navbarSupportedContent" aria-
controls="navbarSupportedContent" aria-expanded="false"
      aria-label="Toggle navigation" (click)="animateButton()"
[ngStyle]="buttonAnimation">
      <span class="navbar-toggler-icon"></span>
    </button>
    <div class="collapse navbar-collapse" id="navbarSupportedContent">
      <ul class="navbar-nav me-auto mb-2 mb-lg-0 navi">
        <li class="nav-item">
          <a class="nav-link active" aria-current="page" routerLink="/"
routerLinkActive="active">Home</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" routerLink="/about"
routerLinkActive="active">Contact</a>
        </li>
      </ul>
    </div>
  </div>
</nav>
```

Scss

```scss
.navbar-light .navbar-toggler {
    color: rgba(0, 0, 0, 0.55);
    border-color: rgba(0, 0, 0, 0) !important;

    &:focus {
        border: none !important;
        box-shadow: none !important;
    }
```

```css
}

.navbar {
    padding: 0px 150px;
    font-family: Agrandir;
    color: black;
    width: 100%;

    .contain {
        box-shadow: rgba(0, 0, 0, 0.15) 1.95px 1.95px 2.6px;
        backdrop-filter: blur(2px);

        position: absolute;
        height: 10vh;
        top: 0px;
        left: 0px;
        width: 100%;
        display: flex;
        justify-content: space-between;
        align-items: center;
        padding: 0px 10%;
    }


    .navi {
        display: flex;
        align-items: center;
        width: 100%;
        justify-content: end;
        flex-direction: row;

        li {
            padding: 15px 20px;
            font-size: 1.2rem;
        }
    }
}

.navbar-brand {
    color: #212529;
    font-style: italic;
}
```

Ts

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

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

  public clicked: boolean = false;
  public triggerAnimation!: boolean;
  public scrollValue!: number;

  constructor() { }

  ngOnInit(): void {
    this.animateHeader();
  }

  //OnScroll trigger za promenu stilova header-a
  animateHeader() {
    window.addEventListener('scroll', () => {
      this.scrollValue = window.pageYOffset;
      if (this.scrollValue > 100) {
        this.triggerAnimation = true;
      }
      else {
        this.triggerAnimation = false;
      }
    })
  }

  animateButton() {
    this.clicked = !this.clicked;
  }

  //Dodavanje stilova i animacija menu-button-u
  get buttonAnimation() {
    if (this.clicked) {
      return {
        "animation-name": "animateMenuIn",
        "animation-duration": "3s",
```

```
          "animation-fill-mode": "forwards"
        }
      }
      return {
        "animation-name": "animateMenuOut",
        "animation-duration": "3s",
        "animation-fill-mode": "forwards"
      }
    }

    //stilovi
    get headerOverlay() {
      if (this.triggerAnimation) {
        return {
          "animation-name": "headerIn",
          "animation-duration": "0.5s",
          "animation-fill-mode": "forwards"
        }
      }
      return {
        "animation-name": "headerOut",
        "animation-duration": "0.5s",
        "animation-fill-mode": "forwards"
      }
    }

    get aTagColor() {
      if (this.triggerAnimation) {
        return {
          "color": "#ffedc5"
        }
      }
      return {
        "color": "#212529"
      }
    }
}
```

Footer component

Html

```html
<div class="container foot">
    <div class="row">
        <div class="col-lg-6 col-md-6 col-12 copy ">
            <ul class="navbar-nav me-auto mb-2 mb-lg-0 navi">
                <li class="nav-item">
                    <a class="nav-link active" aria-current="page" routerLink="/"
routerLinkActive="active">Home</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" routerLink="/main"
routerLinkActive="active">About</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" routerLink="/author"
routerLinkActive="active">Author</a>
                </li>
            </ul>
            <p>NIKOLAGUTIC&#169;2022</p>
        </div>
        <div class="col-lg-6 col-md-6 col-12">
            <h1>collēctiō</h1>
            <ul>
                <li>COPYWRITHING     </li><a
href="">Nikola Gutic</a><br>
                <li>DESIGN, DEVELOP  </li><a href="">Nikola
Gutic</a><br>
            </ul>
        </div>
    </div>
</div>
```

Scss

```scss
.container {
    padding: 50px;
    margin-top: 50px;
    height: 40vh;
```

```css
    font-family: Agrandir;
h1 {
    font-family: 'Bagerich';
    font-size: 3rem;
}

ul {
    padding: 0px;

    li {
        display: inline;
    }

    justify-content: flex-end;
}

a {
    display: inline;
    color: black;
    text-decoration: underline;
}

.copy {
    display: flex;
    flex-direction: column;
    justify-content: flex-end;

    a {
        text-decoration: none;
    }

    p {
        margin-top: 30px;
    }

    .navi {
        width: 100%;
    }
}

@media only screen and (max-width: 768px) {

    h1,
    ul,
    p {
```

```
            text-align: center !important;
        }
    }
}
```

Ts

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

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

  constructor() { }

  ngOnInit(): void {
  }

}
```

Collection component

Html

```
<div class="row">
    <div class="background-image col-12"
        style="background-
image:url('../../../assets/images/{{data.backgroundImage}}')">
    </div>
    <div class="image-info-container">
        <div class="collection-image">
            <img src="../../../assets/images/{{data.image}}" alt="{{data.image}}"
srcset="">
        </div>
    </div>
```

```html
</div>
<div class="row">
    <div class="image-info-container">
        <div class="collection-info">
            <div class="author">
                <h1 class="title">{{data.title}}</h1>
                <p *ngIf="author">By : {{author.firstName + " " +
author.lastName}}</p>
                <p>{{data.description}}</p>
            </div>
        </div>
    </div>
</div>
<div class="container">
    <div class="row">
        <div class="col-12 d-flex flex-wrap justify-content-around">
            <app-collection-item class="col-lg-3 col-md-6 col-sm-6 col-12"
*ngFor="let item of items" [itemData]="item"
                [title]="collectionTitle">
            </app-collection-item>
        </div>
    </div>

</div>
```

Scss

```scss
.background-image {
    height: 50vh;
    background-position: center;
    background-repeat: no-repeat;
    background-size: cover;

    img {
        height: 100%;
        width: 100%;

    }
}

.row {
    position: relative;
```

```css
}

.collection-info {
    font-family: "Agrandir";
    padding-top: 2rem;

    .title {
        font-family: "Bagerich";
        font-size: 4rem;
    }
}

.image-info-container {
    width: 100%;
    padding: 0px 10%;
}

.collection-image {
    position: absolute;
    bottom: -10%;
    width: 200px;
    border: 5px solid white;
    overflow: hidden;
    display: flex;
    align-items: center;
    justify-content: center;

    img {
        width: 100%;
    }
}

@media only screen and (max-width: 992px) {
    .image-info-container {
        display: flex;
        justify-content: center;

        .collection-info {
            text-align: center;
        }
    }
}
```

Ts

```typescript
import { Component, OnInit } from '@angular/core';
import { AngularFireAction } from '@angular/fire/compat/database';
import { ActivatedRoute } from '@angular/router';
import { Observable } from 'rxjs';
import { Items } from 'src/app/interfaces/items';
import { CollectionsServices } from 'src/app/services/collections.services';

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

  collectionTitle!: string;
  sub!: any;
  data!: any;
  author!: any;
  items = new Array<any>();

  itemData!: Items[];

  authorObs!: Observable<unknown[]>;
  collection!: Observable<unknown[]>;
  itemsObs!: Observable<unknown[]>[];

  constructor(private route: ActivatedRoute, private collectionServices:
CollectionsServices) { }

  ngOnInit(): void {
    this.getCollectionName();
    this.getCollection(this.collectionTitle);
  }

  ngAfterViewInit() {
    this.valueChecker();
  }

  valueChecker() {
    if (this.data == undefined) {
      this.valueChecker();
    }
```

```
      this.getAuthor(this.data.userId);
      this.getItems(this.data.items);
  }

  getCollectionName() {
    this.sub = this.route.params.subscribe(params => {
      this.collectionTitle = params['title'].replaceAll('-', ' ');
    });
  }

  getCollection(name: string) {
    this.collection = this.collectionServices.find(name);
    this.collection.subscribe(val => this.data = val[0]);
  }

  getAuthor(id: string) {
    this.authorObs = this.collectionServices.author(id);
    this.authorObs.subscribe(val => this.author = val[0]);
  }

  getItems(itemIds: []) {
    this.itemsObs = this.collectionServices.items(itemIds);
    this.itemsObs.forEach(element => {
      element.subscribe(val => this.items.push(val[0]));
    });
  }
}
```

Collection item component

Html

```html
<div class="card">
    <div class="content">
        <div class="imgBx">
            <img src="../../assets/images/{{itemData.image}}">
        </div>
    </div>
    <div class="item-info">
```

```
        <a [routerLink]="['/collection',transformUrl(title),
transformUrl(itemData.title)]">
            <h5>{{itemData.title}}</h5>
        </a>
    </div>
</div>
```

Scss

```scss
.card {
    position: relative;
    height: 300px;
    background: rgba(255, 255, 255, 0.05);
    margin: 20px;
    display: flex;
    justify-content: center;
    align-items: center;
    overflow: hidden;

    box-shadow: 0px 10px 1px rgba(221, 221, 221, 1), 0 10px 20px rgba(204, 204,
204, 1);

    .content {
        position: relative;
        display: flex;
        width: 100%;
        justify-content: center;
        align-items: center;
        flex-direction: column;
        transition: 0.5s;

        .imgBx {
            position: relative;
            width: 100%;
            height: 300px;
            overflow: hidden;

            img {
                position: absolute;
                inset: 0px;
                box-sizing: border-box;
                padding: 0px;
```

```
                border: none;
                margin: auto;
                display: block;
                width: 100%;
            }


        }
    }

    .item-info {
        position: absolute;
        bottom: -50px;
        width: 100%;
        height: 50px;
        transition: 0.5s;
        background-color: #fff;

        display: flex;
        flex-direction: column;
        justify-content: center;
        align-items: center;

        font-family: "Bagerich";
        h5{
            text-align: center;
        }
        a{
            color: #212529;
        }
    }

    &:hover .item-info {
        transform: translateY(-50px);
    }

}
```

Ts

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

@Component({
```

```
  selector: 'app-collection-item',
  templateUrl: './collection-item.component.html',
  styleUrls: ['./collection-item.component.scss']
})
export class CollectionItemComponent implements OnInit {

  @Input() itemData!: any;
  @Input() title!: string;

  constructor() { }

  ngOnInit(): void {
  }
  public transformUrl(data: string): string {
    return data.replace(/\s/g, '-');
  }
}
```

Author component

Html

```html
<div class="main">
    <div class="row">
        <div class="col-2">
            <div class="side side-left">
                <p>less.less.less.less.</p>
            </div>
        </div>
        <div class="col-8">
            <div class="info">
                <div class="row d-flex justify-content-center">
                    <div class="col-md-6 col-10 image">
                        <img src="../../../assets/images/author1.jpeg">
                    </div>
                    <div class="col-md-6 col-10 author">
                        <h1>Hello I'm Nikola</h1>
                        <p>I am studying at the College of Information and
Communication Technologies, majoring in
                            Internet
```

```
                            Technology and I enjoy everything this college has to
offer, web3 enthusiast and I work at A
                            Tavola, a
                            great startup company that develops technology based
on computer vision, designed for collection,
                            analysis and data use for improving restaurant front
and back of the house operations, as
                            well
                            as to optimize overall operations and efficiency of
the restaurants.</p>
                    </div>
                </div>
            </div>
        </div>
        <div class="col-2">
            <div class="side side-right">
                <p>less.less.less.less.</p>
            </div>
        </div>
    </div>
</div>
```

Scss

```scss
.main {
    .side {
        font-size: 10rem;
        font-family: Bagerich;
    }

    .side-left {
        p {
            position: relative;
            top: -50px;
            transform: rotate(90deg) scale(1, -1);
        }
    }

    .side-right {
        p {
            position: relative;
            top: -50px;
```

```scss
        transform: rotate(90deg);
    }
}

.info {
    padding-top: 150px;
    width: 100%;
    margin-bottom: 150px;
}

.image {
    img {
        width: 80%;
        float: right;
    }
}

.author {
    h1 {
        font-size: 5rem;
        font-family: Bagerich;

    }

    p {
        text-align: justify;
    }
}
}
```

Ts

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

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

```
  public infoBlock!: any;
  constructor() { }

  ngOnInit(): void {
  }

  click(event: { currentTarget: any; }) {
    var idAttr = event.currentTarget.id
    this.change(idAttr);
  }
  change(id: string) {
    var selector = "#info" + id;
    this.infoBlock = document.querySelector(selector);
    if (this.infoBlock.classList.contains('none')) {
      this.infoBlock.classList.remove('none');
      this.infoBlock.classList.add('block');
    }
    else {
      this.infoBlock.classList.add('none');
      this.infoBlock.classList.remove('block');
    }
  }
}
```

About (Contact) component

Html

```
<div class="container">
    <div class="about-main">
        <div class="col-6">
            <h1>Contact Us</h1>
            <p>Feel free to contact us and we will get back to you as soon as we
can.</p>
        </div>
    </div>
    <div class="contact-area">
        <div class="container">
            <form [formGroup]="contactForm" (ngSubmit)="onSubmit()">
                <div class="row">
```

```html
                  <div class="col-sm-6 col-12">
                      <mat-form-field appearance="outline">
                          <mat-label>Your Name <span class="req">*</span>
</mat-label>

                          <input matInput formControlName="name" />

                          <div *ngIf="name?.invalid && (name?.dirty ||
name?.touched)" class="alert alert-danger">
                              <mat-error>
                                  <ng-container
*ngIf="name?.errors?.['required']">
                                      Your full name is required.
                                  </ng-container>
                              </mat-error>
                          </div>
                      </mat-form-field>
                  </div>
                  <div class="col-sm-6 col-12">
                      <mat-form-field appearance="outline">
                          <mat-label>Email <span class="req">*</span></mat-
label>

                          <input matInput formControlName="email" />
                          <div *ngIf="email?.invalid && (email?.dirty ||
email?.touched)" class="alert alert-danger">
                              <mat-error>
                                  <ng-container
*ngIf="email?.errors?.['required']">
                                      Email is required.
                                  </ng-container>
                                  <ng-container
*ngIf="email?.errors?.['minlength']">
                                      The field is under min Len.
                                  </ng-container>
                              </mat-error>
                          </div>
                      </mat-form-field>
                  </div>
              </div>
              <div class="row">
                  <div class="col-12">
                      <mat-form-field appearance="outline">
                          <mat-label>Message <span class="req">*</span></mat-
label>

                          <textarea name="msg" id="msg" matInput
formControlName="msg">
```

```html
                                    </textarea>
                                    <div *ngIf="msg?.invalid && (msg?.dirty ||
msg?.touched)" class="alert alert-danger">
                                        <mat-error>
                                            <ng-container
*ngIf="msg?.errors?.['required']">
                                                Message is required.
                                            </ng-container>
                                        </mat-error>
                                    </div>
                            </mat-form-field>
                        </div>
                    </div>
                    <div class="row">
                        <div class="col-12">
                            <button class='send' mat-flat-button color="primary"
(ngSubmit)="onSubmit()"
                                [disabled]="contactForm.invalid">send</button>
                        </div>
                    </div>
                </form>
            </div>
        </div>
</div>
```

Css

```css
input {
    background-color: rgba(0, 0, 0, 0);
    outline: none;
    border: none;
    border-bottom: 1.3px solid black;
    width: 100%;
}


mat-label {
    margin-top: 20px;
    display: block;
}

.alert-danger {
```

```css
    background-color: rgba(0, 0, 0, 0);
    border: none;
    height: 30px;
    padding: 0px;
    margin: 0px;
}

.req {
    color: brown;
}

#msg {
    width: 100%;
    background-color: rgba(0, 0, 0, 0);
    padding: 20px;

    &:focus {
        outline: none;
    }
}

.about-main {
    padding: 0px 12px;
    margin: 20vh 0px 10vh 0px;
    h1{
        font-family: Bagerich;
        font-size: 5rem;
    }
}

.send {
    background-color: rgba(0, 0, 0, 0);
    border: 1px solid black;
    outline: none;
    width: 200px;
    height: 50px;

    &:focus {
        outline: none;
    }
}
```

Ts

```ts
import { MessagesService } from './../../services/messages.service';
import { Component, OnInit } from '@angular/core';
import { FormGroup, FormControl, Validators } from '@angular/forms';

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

  public contactForm = new FormGroup({
    name: new FormControl('', [Validators.required]),
    email: new FormControl('', [Validators.required, Validators.minLength(5)]),
    msg: new FormControl('', [Validators.required])
  }
  );

  constructor(private messagesServices: MessagesService) { }

  ngOnInit(): void {
  }

  onSubmit(): void {
    if (this.contactForm.valid) {
      this.messagesServices.sendEmail(this.contactForm.value);
    }
    this.contactForm.reset();
  }

  get name() { return this.contactForm.get('name'); }
  get email() { return this.contactForm.get('email'); }
  get msg() { return this.contactForm.get('msg'); }
}
```

Interfaces

Collection

```
export interface Collections {
    id : number;
    title: string;
    image: string;
    backgroundImage: string;
    userId: number;
    categoryId: number;
    items: [number];
  }
```

Infos

```
export interface Infos {
    id : number;
    title: string;
    content: string;
  }
```

Items

```
export interface Items {
    id : number;
    title: string;
    image: string;
    model: string;
    infos: [string];
  }
```

Services

Collection.services

```
import { Injectable } from '@angular/core';
import { BehaviorSubject, Observable } from 'rxjs';
```

```typescript
import { AngularFireDatabase, AngularFireList } from
'@angular/fire/compat/database';

import { Collections } from './../interfaces/collections';


@Injectable({
  providedIn: 'root'
})

export class CollectionsServices {

  public collections: Observable<Collections[]>;
  private itemsRef: AngularFireList<any>;

  singleCollection!: Observable<unknown[]>;
  collectionModel!: BehaviorSubject<any>;

  authorData!: Observable<unknown[]>;
  private itemsData!: any;

  constructor(private db: AngularFireDatabase) {

    this.itemsRef = db.list('collections');
    this.collections = this.itemsRef.valueChanges();
  }

  public find(title: string | null): Observable<unknown[]> {
    this.singleCollection = this.db.list('collections', ref =>
      ref.orderByChild('title').equalTo(title)
    ).valueChanges();

    return this.singleCollection;
  }

  public author(userId: string | null): Observable<unknown[]> {
    this.authorData = this.db.list('users', ref =>
      ref.orderByChild('id').equalTo(userId)
    ).valueChanges();

    return this.authorData;
  }

  public items(itemIds: []): Observable<unknown[]>[] {
    this.itemsData = new Array<Observable<unknown[]>>();
```

```
    itemIds.forEach(element => {
      let obj = this.db.list('collectionItems', ref =>
        ref.orderByChild('id').equalTo(element)
      ).valueChanges();
      this.itemsData.push(obj);
    });
    return this.itemsData;
  }


  public getCollections(): Observable<Collections[]> {
    return this.collections;
  }

}
```

Items.service

```
import { AngularFireDatabase } from '@angular/fire/compat/database';
import { Injectable } from '@angular/core';
import { Observable } from 'rxjs';

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

  private itemInfos!: any;
  private singleItem!: Observable<unknown[]>;;
  constructor(private db: AngularFireDatabase) {

  }

  public find(title: string | null): Observable<unknown[]> {
    this.singleItem = this.db.list('collectionItems', ref =>
      ref.orderByChild('title').equalTo(title)
    ).valueChanges();

    return this.singleItem;
  }

  public infos(infoIds: []): Observable<unknown[]>[] {
```

```
      this.itemInfos = new Array<Observable<unknown[]>>();
      infoIds.forEach((element: string | number | boolean | null) => {
        let obj = this.db.list('itemInfos', ref =>
          ref.orderByChild('id').equalTo(element)
        ).valueChanges();
        this.itemInfos.push(obj);
      });
      return this.itemInfos;
    }


}
```

Messages.service

```
import { Injectable } from '@angular/core';
import { HttpClient, HttpHeaders } from '@angular/common/http';

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

  constructor(private http: HttpClient) { }

  sendEmail(formData: any): void {
    const headers = new HttpHeaders({ 'Content-Type': 'application/json' });

    this.http.post('https://formspree.io/f/mbjbllkj',
      { name: formData.name, replyto: formData.email, message: formData.msg },
      { 'headers': headers }).subscribe(
        response => {
          console.log(response);
        }
      );
  }
}
```

Animations

```
import { animate, query, stagger, style, transition, trigger } from
"@angular/animations";

export const $animations = [

    trigger('fade', [
        transition('void=>*', [
            style({
                opacity: 0
            }),
            animate('3s ease-in-out')
        ])
    ]),
    trigger('slideSide', [
        transition('void=>*', [
            style({
                transform: "translateX(-300px)"
            }),
            animate("3s ease-in-out")
        ])
    ]),
    trigger('slideUp', [
        transition('void=>*', [
            style({
                transform: "translateY(200px)"
            }),
            animate("3s ease-in-out")
        ])
    ])
];
```

App-routing-model.ts

```
import { CollectionComponent } from
'./components/collection/collection.component';
import { AuthorComponent } from './components/author/author.component';
import { AboutComponent } from './components/about/about.component';
import { HomeComponent } from './components/home/home.component';
import { NgModule } from '@angular/core';
```

```typescript
import { RouterModule, Routes } from '@angular/router';
import { ItemComponent } from './components/item/item.component';

const routes: Routes = [
  {
    path: '',
    component: HomeComponent,
    data: { title: 'Collēctiō | Home' }
  },
  {
    path: 'collection/:title',
    component: CollectionComponent,
    data: { title: 'Collēctiō | Details' }
  },
  {
    path: 'collection/:title/:name',
    component: ItemComponent,
    data : {title: 'Collēctiō | Details'}
  },
  {
    path: 'about',
    component: AboutComponent,
    data: { title: 'Collēctiō | About' }
  },
  {
    path: 'author',
    component: AuthorComponent,
    data: { title: 'Nikola Gutic' }
  }
];

@NgModule({
  imports: [RouterModule.forRoot(routes)],
  exports: [RouterModule]
})
export class AppRoutingModule {
}
```

App.module.ts

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

```typescript
import { BrowserModule, Title } from '@angular/platform-browser';
import { NgbModule } from '@ng-bootstrap/ng-bootstrap';
import { HttpClientModule } from '@angular/common/http';

import { BrowserAnimationsModule } from '@angular/platform-browser/animations';
import { AppRoutingModule } from './app-routing.module';
import { AngularFireModule } from "@angular/fire/compat";
import { AngularFireDatabaseModule } from '@angular/fire/compat/database';

import { AppComponent } from './app.component';
import { HeaderComponent } from './components/header/header.component';
import { LogoComponent } from './components/elements/logo/logo.component';

import { BannerComponent } from './components/home/banner/banner.component';
import { HomeComponent } from './components/home/home.component';
import { environment } from 'src/environments/environment';
import { FooterComponent } from './components/footer/footer.component';
import { AboutComponent } from './components/about/about.component';
import { AuthorComponent } from './components/author/author.component';
import { ReactiveFormsModule } from '@angular/forms';
import { CollectionsComponent } from
'./components/home/collections/collections.component';
import { SingleCollectionComponent } from './components/home/collections/single-
collection/single-collection.component';
import { CollectionComponent } from
'./components/collection/collection.component';
import { CollectionItemComponent } from './components/collection/collection-
item/collection-item.component';
import { ItemComponent } from './components/item/item.component';
import { ItemInfosComponent } from './components/item/item-infos/item-
infos.component';

@NgModule({
  declarations: [
    AppComponent,
    HeaderComponent,
    LogoComponent,
    BannerComponent,
    HomeComponent,
    FooterComponent,
    AboutComponent,
    AuthorComponent,
    CollectionItemComponent,
    CollectionsComponent,
    SingleCollectionComponent,
```

```typescript
    CollectionComponent,
    ItemComponent,
    ItemInfosComponent,
  ],
  imports: [
    BrowserModule,
    AppRoutingModule,
    NgbModule,
    BrowserAnimationsModule,
    AngularFireModule.initializeApp(environment.firebaseConfig),
    AngularFireDatabaseModule,
    ReactiveFormsModule,
    HttpClientModule,
  ],
  providers: [Title],
  bootstrap: [AppComponent],
  schemas: [CUSTOM_ELEMENTS_SCHEMA]
})
export class AppModule { }
```