JOVANA KRŽANOVIĆ 2018200030

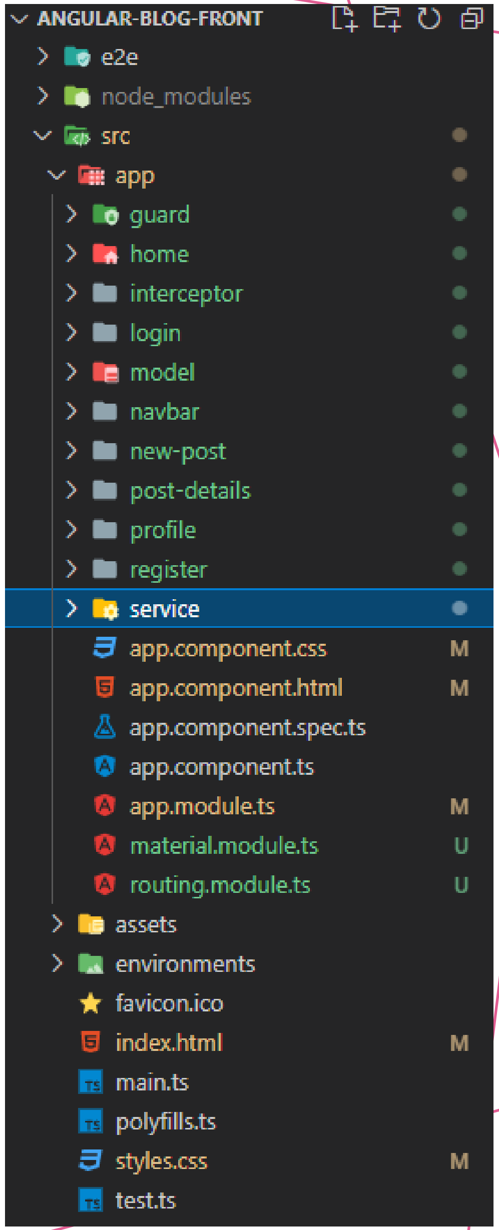
WEB BLOG APPLICATION

UVOD

- U okviru prezentacije biće prikazana implementacija frontend dela veb blog aplikacije korišćenjem Angular framework-a. Svrha navedene aplikacije je da omogući registrovanim korisnicima da objavljuju blogove, pretražuju sadržaje blogova postavljenih od strane drugih korisnika, dodavanje blogova u listu omiljenih i ostavljanje komentara. Takođe korisnici platforme imaju mogućnost da pregledaju korisničke naloge drugih korisnika, njihove liste omiljenih i objavljenih blogova. Korisniku se nudi i opcija ažuriranja podataka , kao i brisanje naloga.

STRUKTURA

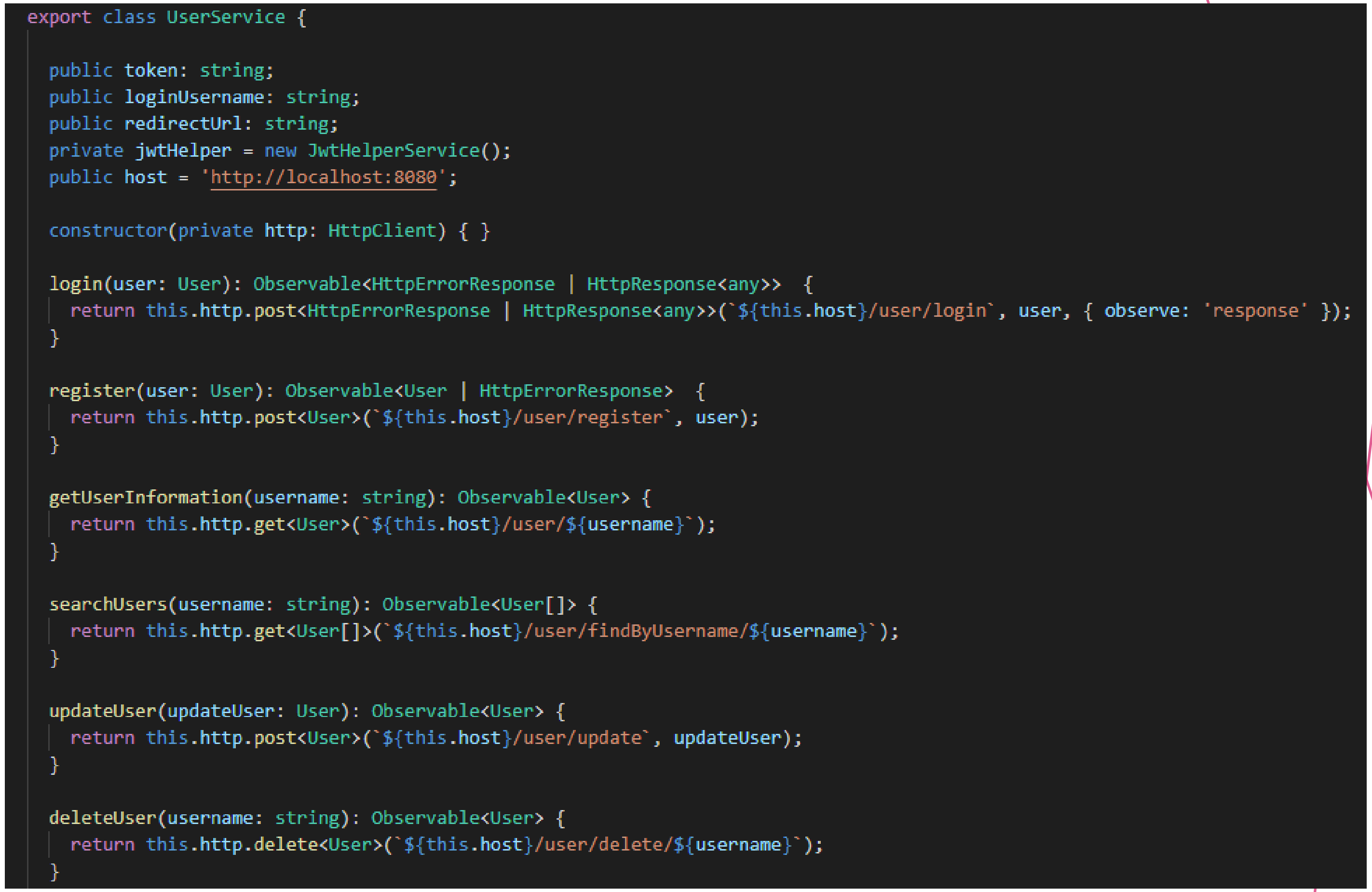
- Aplikacija sadrži:
- 7 komponenti - home, login, navbar, new-post, post-details, profile i register
- 3 servisa - user, post i auth.guard
- 3 modela - user, post i comment
- 1 interceptor - authentication
- Korena (root komponenta), app.module, material.module i routing.module.



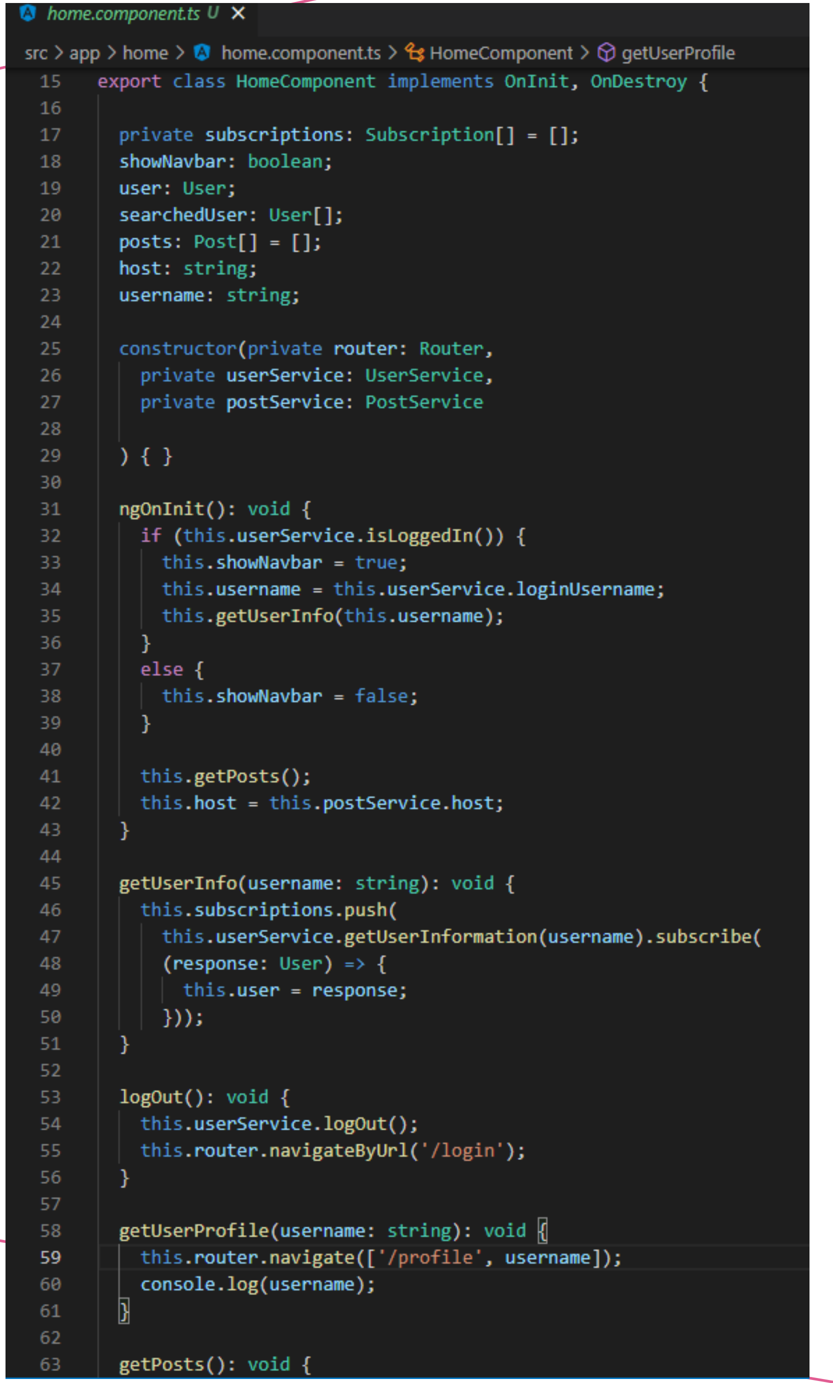
```
ANGULAR-BLOG-FRONT
  e2e
  node_modules
  src
    app
      guard
      home
      interceptor
      login
      model
      navbar
      new-post
      post-details
      profile
      register
      service
      app.component.css          M
      app.component.html         M
      app.component.spec.ts
      app.component.ts
      app.module.ts              M
      material.module.ts         U
      routing.module.ts          U
    assets
    environments
    favicon.ico
    index.html                   M
    main.ts
    polyfills.ts
    styles.css                   M
    test.ts
```

SERVISI

- Uloga servisa je komunikacija sa backend-om, tj, API-jem preko pristupnih tačaka (endpoints). Takođe se koriste i za smeštanje logine authentifikacije korisnika aplikacije (login(), logout(), čitanje, čuvanje i brisanje tokena dohvaćenih sa backend-a, proveravanje da li je korisnik ulogovan metodom isLoggedIn() koja predstavlja neki oblik sesije koji će omogućiti promenu sadržaja na stranicama u odnosu na stanje same metode.



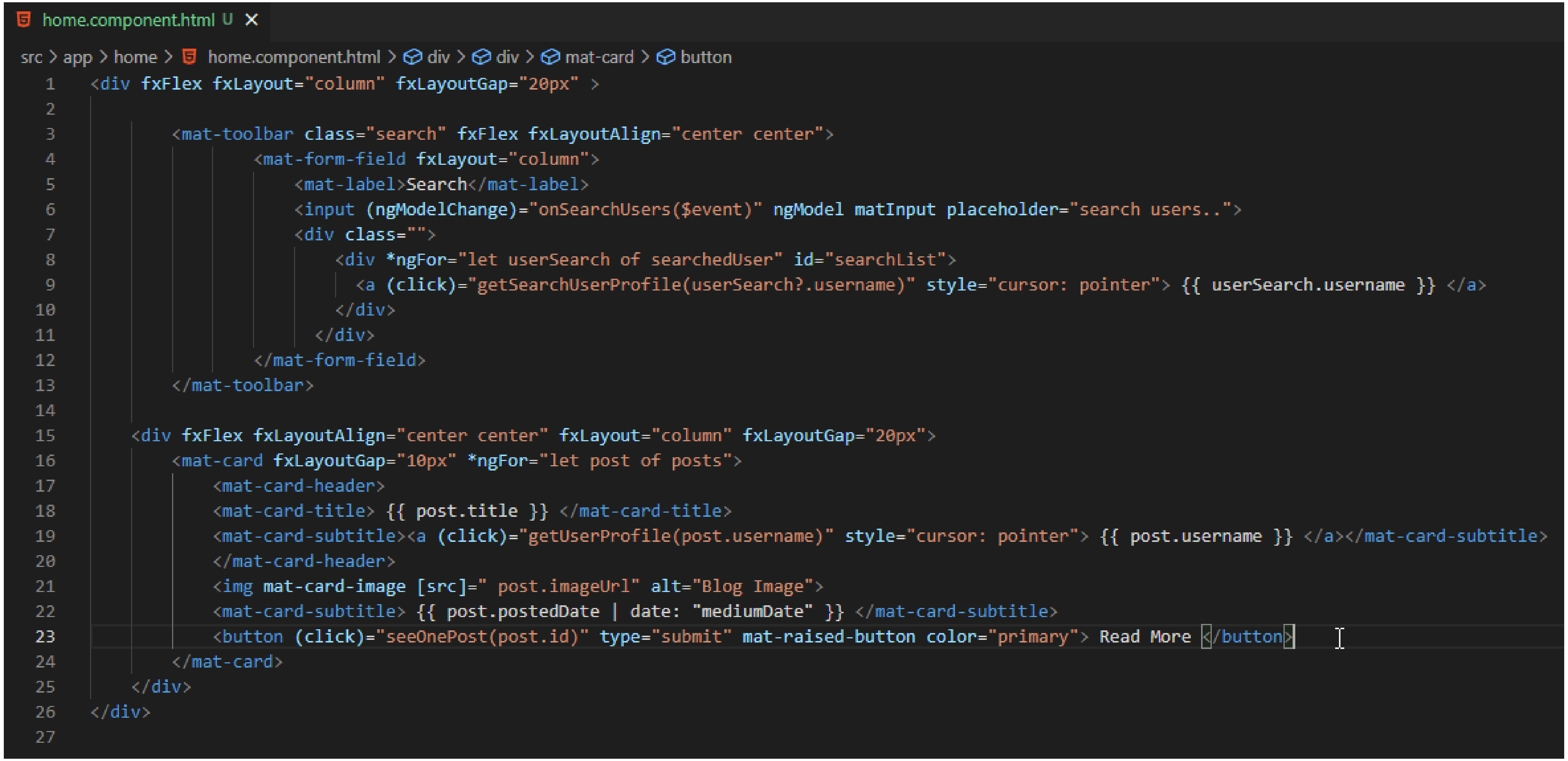
```
export class UserService {

  public token: string;
  public loginUsername: string;
  public redirectUrl: string;
  private jwtHelper = new JwtHelperService();
  public host = 'http://localhost:8080';

  constructor(private http: HttpClient) { }

  login(user: User): Observable<HttpErrorResponse | HttpResponse<any>>  {
    return this.http.post<HttpErrorResponse | HttpResponse<any>>(`${this.host}/user/login`, user, { observe: 'response' });
  }

  register(user: User): Observable<User | HttpErrorResponse>  {
    return this.http.post<User>(`${this.host}/user/register`, user);
  }

  getUserInformation(username: string): Observable<User> {
    return this.http.get<User>(`${this.host}/user/${username}`);
  }

  searchUsers(username: string): Observable<User[]> {
    return this.http.get<User[]>(`${this.host}/user/findByUsername/${username}`);
  }

  updateUser(updateUser: User): Observable<User> {
    return this.http.post<User>(`${this.host}/user/update`, updateUser);
  }

  deleteUser(username: string): Observable<User> {
    return this.http.delete<User>(`${this.host}/user/delete/${username}`);
  }
```


home.component.ts U ×
src > app > home > home.component.ts > HomeComponent > getUserProfile

```
export class HomeComponent implements OnInit, OnDestroy {

  private subscriptions: Subscription[] = [];
  showNavbar: boolean;
  user: User;
  searchedUser: User[];
  posts: Post[] = [];
  host: string;
  username: string;

  constructor(private router: Router,
    private userService: UserService,
    private postService: PostService

  ) { }

  ngOnInit(): void {
    if (this.userService.isLoggedIn()) {
      this.showNavbar = true;
      this.username = this.userService.loginUsername;
      this.getUserInfo(this.username);
    }
    else {
      this.showNavbar = false;
    }

    this.getPosts();
    this.host = this.postService.host;
  }

  getUserInfo(username: string): void {
    this.subscriptions.push(
      this.userService.getUserInformation(username).subscribe(
      (response: User) => {
        this.user = response;
      }));
  }

  logOut(): void {
    this.userService.logOut();
    this.router.navigateByUrl('/login');
  }

  getUserProfile(username: string): void {
    this.router.navigate(['/profile', username]);
    console.log(username);
  }

  getPosts(): void {
```

KOMPONENTE

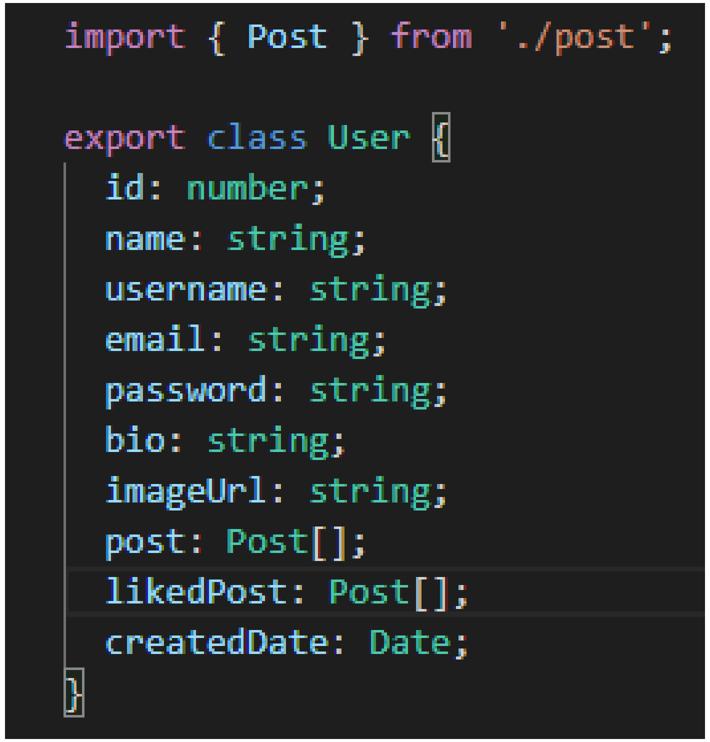
- Komponente čine odvojene celine projekta, koje se sastoje od HTML i CSS datoteka koji pružaju vizuelni izgled aplikacije, i TypeScript datoteke koja sadrži pozadinsku logiku na client-side strani. TypeScript datoteke u okviru komponenti pozivaju funkcije iz određenih servisa, smeštaju informacije dobijene iz tih metoda u određene promenljive koje se kao takve upisuju u određene delove HTML koda, odakle će se i samom korisniku prikazati sadržaj.


home.component.html U ×
src > app > home > home.component.html > div > div > mat-card > button

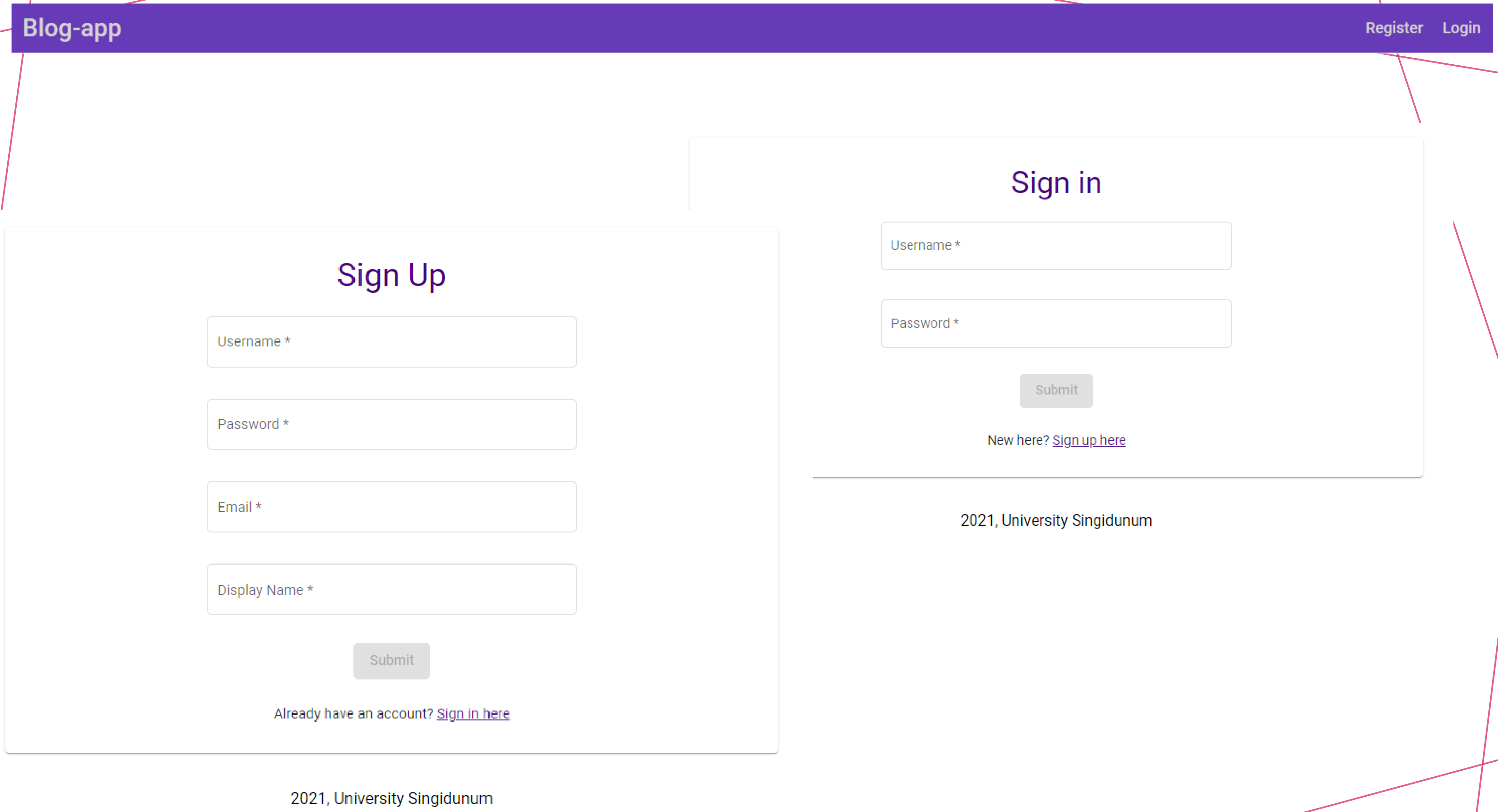
```html
<div fxFlex fxLayout="column" fxLayoutGap="20px" >

        <mat-toolbar class="search" fxFlex fxLayoutAlign="center center">
                <mat-form-field fxLayout="column">
                    <mat-label>Search</mat-label>
                    <input (ngModelChange)="onSearchUsers($event)" ngModel matInput placeholder="search users..">
                    <div class="">
                        <div *ngFor="let userSearch of searchedUser" id="searchList">
                          <a (click)="getSearchUserProfile(userSearch?.username)" style="cursor: pointer"> {{ userSearch.username }} </a>
                        </div>
                      </div>
                </mat-form-field>
        </mat-toolbar>

    <div fxFlex fxLayoutAlign="center center" fxLayout="column" fxLayoutGap="20px">
        <mat-card fxLayoutGap="10px" *ngFor="let post of posts">
            <mat-card-header>
            <mat-card-title> {{ post.title }} </mat-card-title>
            <mat-card-subtitle><a (click)="getUserProfile(post.username)" style="cursor: pointer"> {{ post.username }} </a></mat-card-subtitle>
            </mat-card-header>
            <img mat-card-image [src]=" post.imageUrl" alt="Blog Image">
            <mat-card-subtitle> {{ post.postedDate | date: "mediumDate" }} </mat-card-subtitle>
            <button (click)="seeOnePost(post.id)" type="submit" mat-raised-button color="primary"> Read More </button>
        </mat-card>
    </div>
</div>

```

MODELI

- Modeli predstavljaju preslikavanje entiteta, tj. tabela definisanih u bazi podataka, u koje se smeštaju podaci pristigli sa backend-a.



```
import { Post } from './post';

export class User {
  id: number;
  name: string;
  username: string;
  email: string;
  password: string;
  bio: string;
  imageUrl: string;
  post: Post[];
  likedPost: Post[];
  createdDate: Date;
}
```



Sign Up

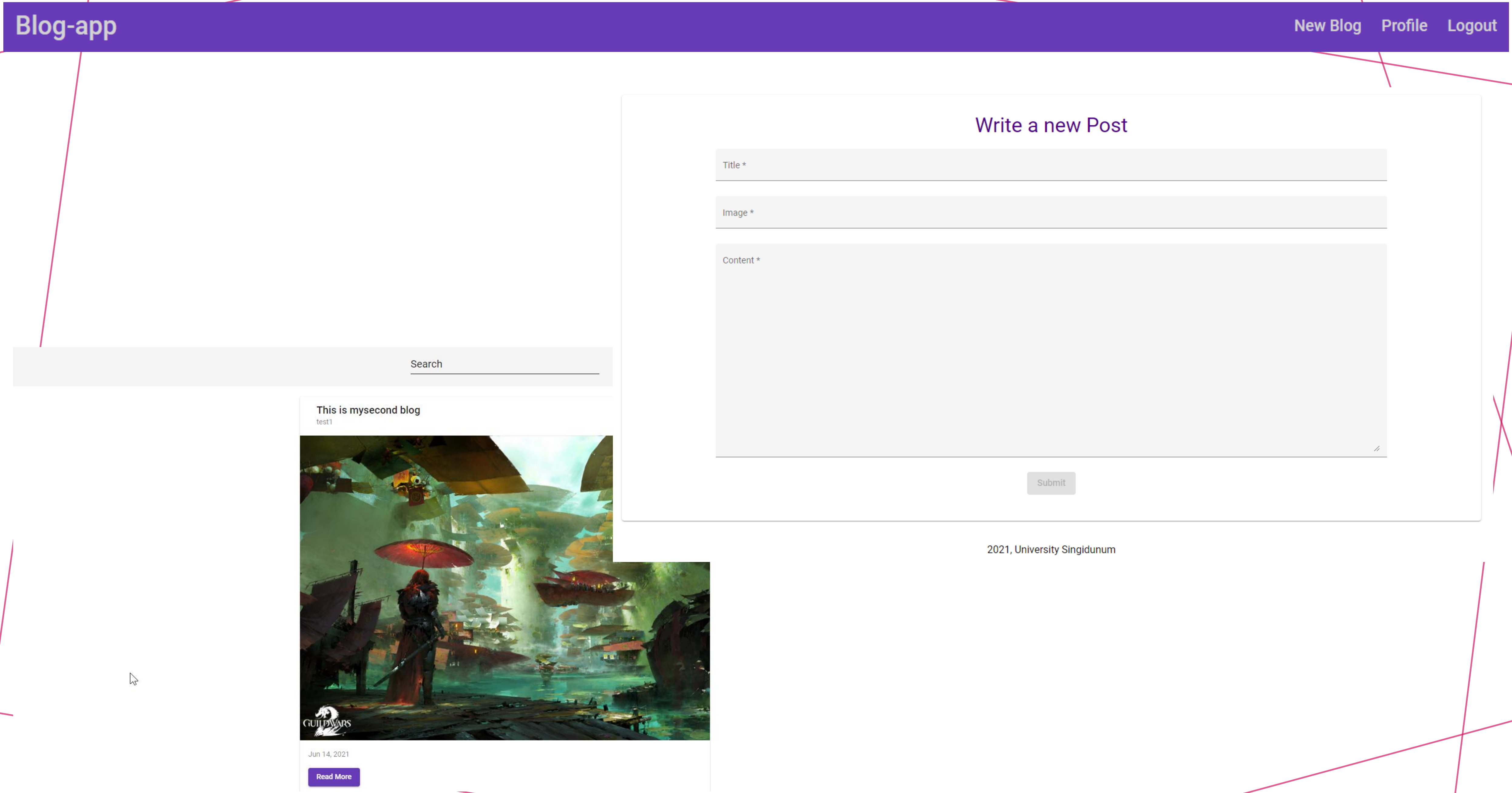
Already have an account? Sign in here

2021, University Singidunum

Sign in

New here? Sign up here

2021, University Singidunum



Write a new Post

Title *

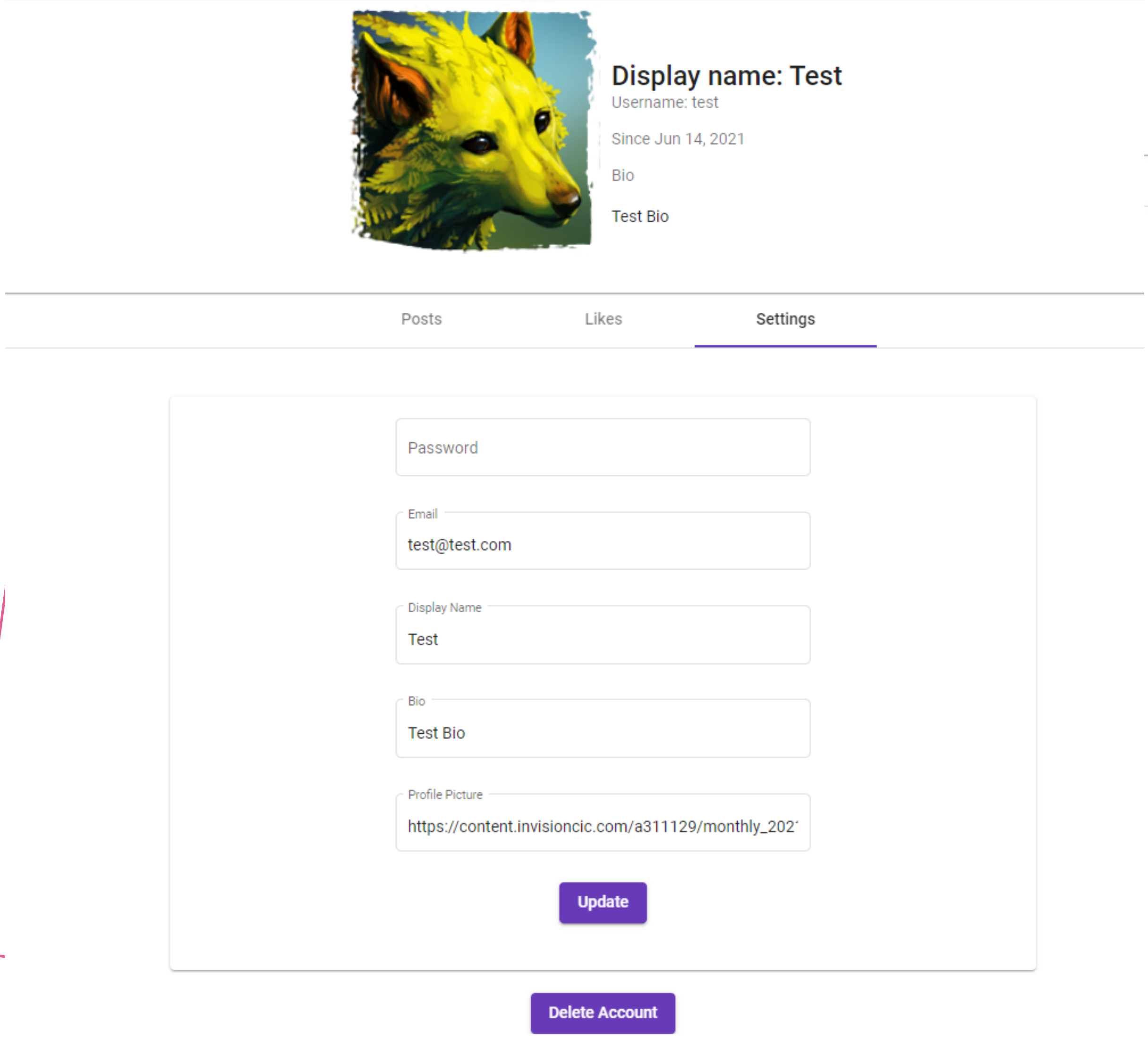
Image *

Content *

This is mysecond blog
test1

Jun 14, 2021

2021, University Singidunum



Display name: Test

Username: test

Since Jun 14, 2021

Bio

Test Bio

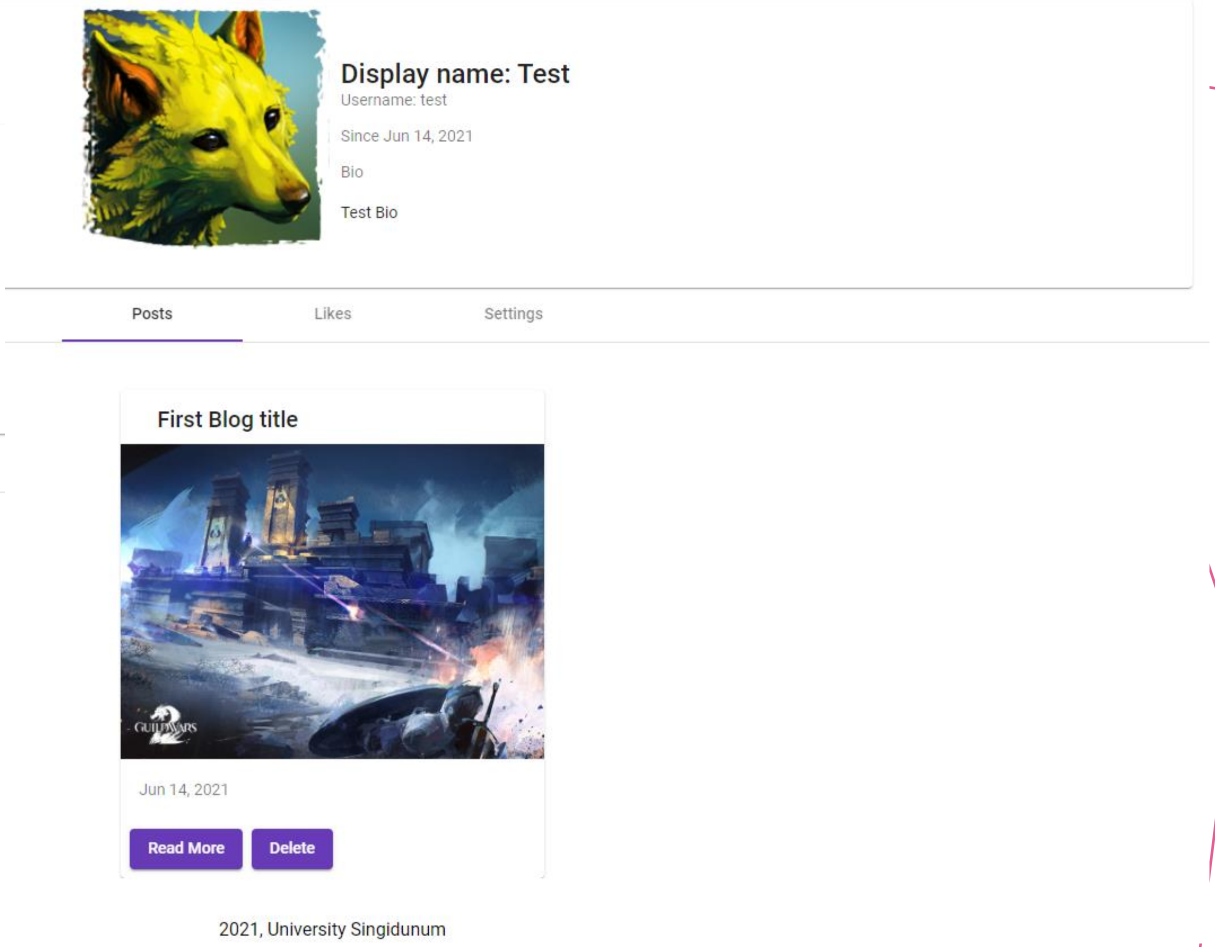
Password

Email: test@test.com

Display Name: Test

Bio: Test Bio

Profile Picture: https://content.invisioncic.com/a311129/monthly_202'



Display name: Test

Username: test

Since Jun 14, 2021

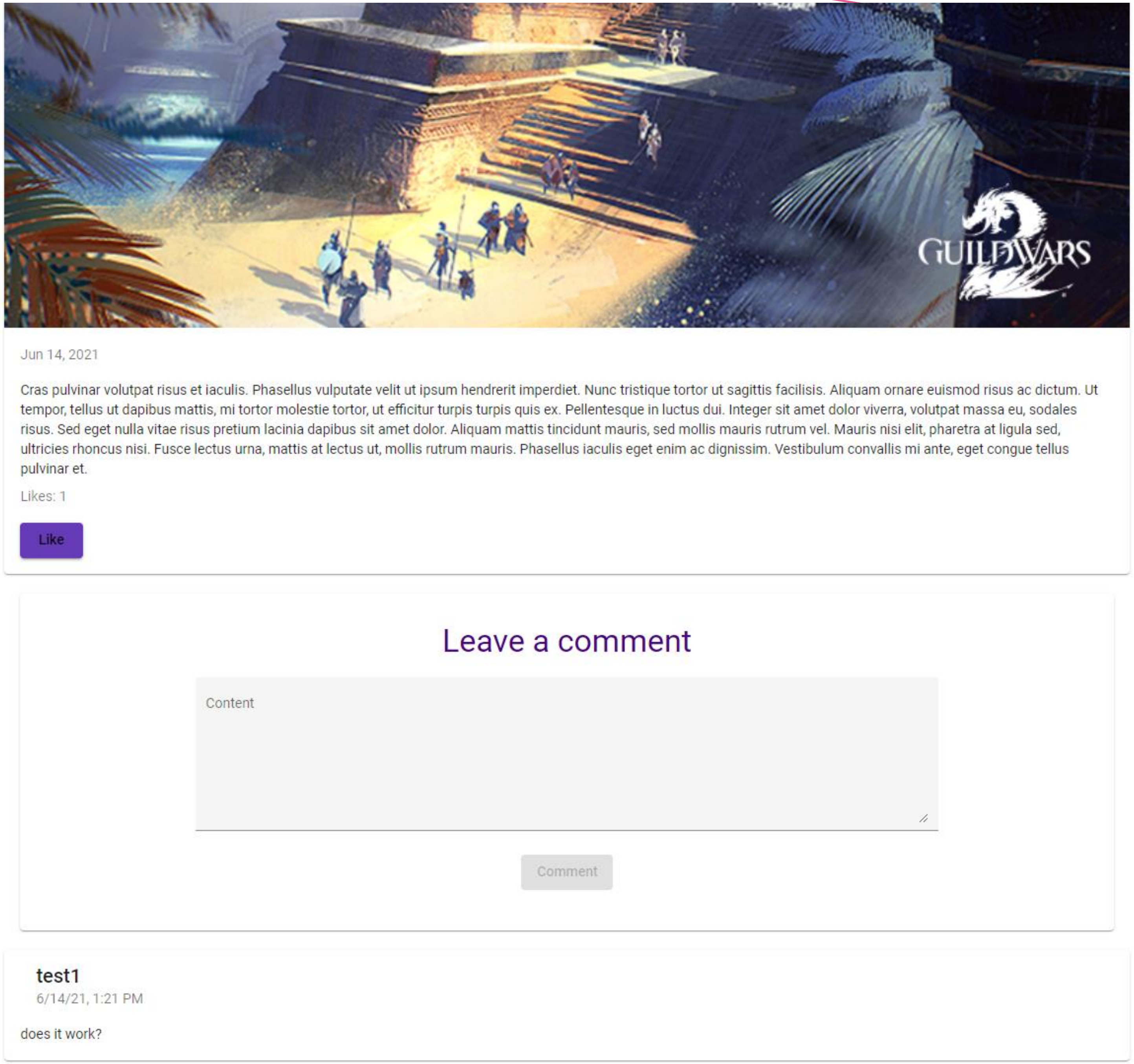
Bio

Test Bio

First Blog title

Jun 14, 2021

2021, University Singidunum



Jun 14, 2021

Cras pulvinar volutpat risus et iaculis. Phasellus vulputate velit ut ipsum hendrerit imperdiet. Nunc tristique tortor ut sagittis facilisis. Aliquam ornare euismod risus ac dictum. Ut tempor, tellus ut dapibus mattis, mi tortor molestie tortor, ut efficitur turpis turpis quis ex. Pellentesque in luctus dui. Integer sit amet dolor viverra, volutpat massa eu, sodales risus. Sed eget nulla vitae risus pretium lacinia dapibus sit amet dolor. Aliquam mattis tincidunt mauris, sed mollis mauris rutrum vel. Mauris nisi elit, pharetra at ligula sed, ultricies rhoncus nisi. Fusce lectus urna, mattis at lectus ut, mollis rutrum mauris. Phasellus iaculis eget enim ac dignissim. Vestibulum convallis mi ante, eget congue tellus pulvinar et.

Likes: 1

Leave a comment

test1
6/14/21, 1:21 PM

does it work?