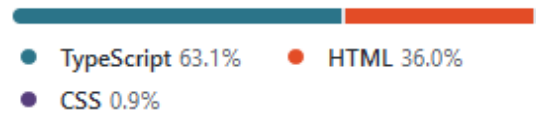


# Angular Framework Example

## Languages



master **EW-008.STUD-ETF-IP.035\_AAA\_AngularArchiveCode / web-application.002 /** Go to file Add file ▾ ...

mirko2021 IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni ...

8290ae4 on 10 Jun History

..

src

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

angular.json

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

master **EW-008.STUD-ETF-IP.035\_AAA\_AngularArchiveCode / web-application.002 / src /** Go to file Add file ▾ ...

mirko2021 IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni ...

8290ae4 on 10 Jun History

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni  
izvedbeni dio - Aplikacija 002.

..

app

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

assets

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

environments

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

favicon.ico

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

index.html

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

main.ts

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

polyfills.ts

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

proxy.conf.json

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

styles.css

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

test.ts

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

master **EW-008.STUD-ETF-IP.035\_AAA\_AngularArchiveCode / web-application.002 / src / assets /** Go to file Add file ▾ ...

mirko2021 IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni ...

8290ae4 on 10 Jun History

..

app\_messages

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

.gitkeep

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

master

EW-008.STUD-ETF-IP.035\_AAA\_AngularArchiveCode / web-application.002 / src / assets / app\_messages /

Go to file

Add file

...

mirko2021

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

...

8290ae4 on 10 Jun

History

..

file\_not\_found.txt

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

mirko2021

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

...

8290ae4 on 10 Jun

History

..

app

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

assets

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

environments

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

favicon.ico

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

index.html

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

main.ts

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

polyfills.ts

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

proxy.conf.json

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

styles.css

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

test.ts

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

mirko2021

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

...

8290ae4 on 10 Jun

History

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

izvedbeni dio - Aplikacija 002.

..

flight-service

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

flight

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

index

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

messages-service

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

messages

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

reservations-service

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

reservations

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

app-routing.module.ts

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

app.component.css

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

app.component.html

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

app.component.spec.ts

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

app.component.ts

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

app.module.ts

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

master

EW-008.STUD-ETF-IP.035\_AAA\_AngularArchiveCode / web-application.002 / src / app / flight-service /

Go to file

Add file

...

mirko2021

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

...

8290ae4 on 10 Jun

History

..

flight.service.spec.ts

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

flight.service.ts

IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni

2 months ago

mirko2021 IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni ...			8290ae4 on 10 Jun	History
..				
flight.component.css	IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni	2 months ago		
flight.component.html	IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni	2 months ago		
flight.component.spec.ts	IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni	2 months ago		
flight.component.ts	IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni	2 months ago		

mirko2021 IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni ...			8290ae4 on 10 Jun	History
..				
index.component.css	IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni	2 months ago		
index.component.html	IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni	2 months ago		
index.component.spec.ts	IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni	2 months ago		
index.component.ts	IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni	2 months ago		

mirko2021 IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni ...			8290ae4 on 10 Jun	History
..				
messages.service.spec.ts	IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni	2 months ago		
messages.service.ts	IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni	2 months ago		

mirko2021 IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni ...			8290ae4 on 10 Jun	History
..				
messages.component.css	IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni	2 months ago		
messages.component.html	IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni	2 months ago		
messages.component.spec.ts	IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni	2 months ago		
messages.component.ts	IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni	2 months ago		

mirko2021 IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni ...			8290ae4 on 10 Jun	History
..				
reservations.service.spec.ts	IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni	2 months ago		
reservations.service.ts	IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni	2 months ago		

mirko2021 IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni ...			8290ae4 on 10 Jun	History
..				
reservations.component.css	IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni	2 months ago		
reservations.component.html	IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni	2 months ago		
reservations.component.spec.ts	IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni	2 months ago		
reservations.component.ts	IP.001-Aplikacija za zaposlene-klijentski dio. Za ispit i konacni	2 months ago		

## angular.json

```
68     "serve": {
69       "builder": "@angular-devkit/build-angular:dev-server",
70       "options": {
71         "browserTarget": "web-application:build",
72         "proxyConfig": "src/proxy.conf.json"
73       },
74       "configurations": {
75         "production": {
76           "browserTarget": "web-application:build:production"
77         }
78       }
79     },
```

Proxy configuration detail

For server interdomain domain communication  
Services and scropts, non XSS without this servers

## proxy.conf.json

```
1  {
2    "/flights/*":{
3      "target": "http://localhost:8080",
4      "secure": false,
5      "logLevel": "debug"
6    },
7    "/reservations/*":{
8      "target": "http://localhost:8080",
9      "secure": false,
10     "logLevel": "debug"
11   },
12   "/messages/*":{
13     "target": "http://localhost:8080",
14     "secure": false,
15     "logLevel": "debug"
16   }
17 }
```

Могуће директно кориштење сервиса, али и страница сервиса  
Кроз интерфејс ангулар апликације у тип скрипт коду или кроз УРЛ

## index.html

```
1  <!doctype html>
2  <html lang="en">
3  <head>
4    <meta charset="utf-8">
5    <title>WebApplication</title>
6    <base href="/">
7    <meta name="viewport" content="width=device-width, initial-scale=1">
8    <link rel="icon" type="image/x-icon" href="favicon.ico">
9  </head>
10 <body>
11   <app-root></app-root>
12 </body>
13 </html>
```

## app.module.ts

```
1  import { NgModule } from '@angular/core';
2  import { BrowserModule } from '@angular/platform-browser';
3
4  import { AppRoutingModule } from './app-routing.module';
5  import { AppComponent } from './app.component';
6  import { MessagesComponent } from './messages/messages.component';
7  import { FlightComponent } from './flight/flight.component';
8  import { ReservationsComponent } from './reservations/reservations.component';
9  import { IndexComponent } from './index/index.component';
10 import { FormsModule, ReactiveFormsModule } from '@angular/forms';
11 import { HttpClientModule } from '@angular/common/http';
12 import { FlightService } from './flight-service/flight.service';
13
14 @NgModule({
15   declarations: [
16     AppComponent,
17     MessagesComponent,
18     FlightComponent,
19     ReservationsComponent,
20     IndexComponent
21   ],
22   imports: [
23     BrowserModule,
24     AppRoutingModule,
25     FormsModule,
26     ReactiveFormsModule,
27     HttpClientModule,
28   ],
29   providers: [FlightService],
30   bootstrap: [AppComponent]
31 })
32 export class AppModule { }
```

## app.components.ts

```
1  import { Component } from '@angular/core';
2  import { FormBuilder, FormGroup, Validators } from '@angular/forms';
3
4  @Component({
5    selector: 'app-root',
6    templateUrl: './app.component.html',
7    styleUrls: ['./app.component.css']
8  })
9  export class AppComponent {
10    title = 'web-application';
11  }
```

## app.components.html

```
10 <style>
11 :host {
12   font-family: -apple-system, BlinkMacSystemFont, "Segoe UI", Roboto, Helvetica, Arial, sans-serif, "Apple Color Emoji", "Segoe UI Emoji", "Segoe UI Symbol";
13   font-size: 14px;
14   color: #333;
15   box-sizing: border-box;
16   -webkit-font-smoothing: antialiased;
17   -moz-osx-font-smoothing: grayscale;
18 }
19
20 h1,
21 h2,
22 h3,
23 h4,
24 h5,
25 h6 {
26   margin: 8px 0;
27 }
--
```

Исјечак из странице  
Задржани само стилови  
Са мјеста новокреиране апликације

## app-routing.module.ts

```
1  import { NgModule } from '@angular/core';
2  import { RouterModule, Routes } from '@angular/router';
3  import { FlightComponent } from '../flight/flight.component';
4  import { IndexComponent } from '../index/index.component';
5  import { MessagesComponent } from '../messages/messages.component';
6  import { ReservationsComponent } from '../reservations/reservations.component';
7
8  const routes: Routes = [
9    {
10     path: 'ui_messages',
11     component: MessagesComponent
12   },
13   {
14     path: 'ui_flights',
15     component: FlightComponent
16   },
17   {
18     path: 'ui_reservations',
19     component: ReservationsComponent
20   },
21   {
22     path: 'index.html',
23     component: IndexComponent
24   },
25   {
26     path: '',
27     component: IndexComponent
28   }
29 ];
30
31 @NgModule({
32   imports: [RouterModule.forRoot(routes)],
33   exports: [RouterModule]
34 })
35 export class AppRoutingModule { }
```

## flight-service.ts

```
1  import { Injectable } from '@angular/core';
2  import { HttpClient, HttpHeaders } from '@angular/common/http';
3
4  @Injectable({
5    providedIn: 'root'
6  })
7  export class FlightService {
8    public httpClient: HttpClient;
9    constructor(httpClient: HttpClient) {
10      this.httpClient = httpClient;
11    }
12
13    insert(insert_data: any){
14      this.httpClient.post<any>('/flights/put', JSON.stringify(insert_data)).subscribe(data => {
15        if(data.success) alert('Подаци о лету су успјешно адаптриани или постављени.');
```

```
16        else alert('Подаци о лету нису успјешно адаптриани или постављени.');
```

```
17        window.location.reload();
18      });
19    }
20    update(update_data: any){
21      this.httpClient.post<any>('/flights/update', JSON.stringify(update_data)).subscribe(data => {
22        if(data.success) alert('Подаци о лету су успјешно измјењени.');
```

```
23        else alert('Подаци о лету нису успјешно измјењени.');
```

```
24        window.location.reload();
25      });
26    }
27    delete(delete_data: any){
28      this.httpClient.post<any>('/flights/delete', JSON.stringify(delete_data)).subscribe(data => {
29        if(data.success) alert('Подаци о лету су успјешно избрисани.');
```

```
30        else alert('Подаци о лету нису успјешно избрисани.');
```

```
31        window.location.reload();
32      });
33    }
34  }
```



## messages-services.ts

```
1  import { Injectable } from '@angular/core';
2
3  @Injectable({
4    providedIn: 'root'
5  })
6  export class MessagesService {
7
8    constructor() { }
9  }
```

## reservation-services.ts

```
1  import { HttpClient } from '@angular/common/http';
2  import { Injectable } from '@angular/core';
3
4  @Injectable({
5    providedIn: 'root'
6  })
7  export class ReservationsService {
8    public httpClient: HttpClient;
9
10   constructor(httpClient: HttpClient) {
11     this.httpClient = httpClient;
12   }
13
14   public insert(data: any): void{
15     this.httpClient.post<any>('/reservations/put', JSON.stringify(data)).subscribe(data => {
16       if(data.success) alert('Подаци о резервацији су успјешно адаптирани или постављени.');
```

```

45
46 public uploadFile(data: any, article_transport_spec_file: File | null): void{
47     if(article_transport_spec_file===null) {alert('Датотека спецификације превоза робе није подигнута на сервер.');
```

## index.component.ts

```

1  import { Component, OnInit } from '@angular/core';
2
3  @Component({
4      selector: 'app-index',
5      templateUrl: './index.component.html',
6      styleUrls: ['./index.component.css']
7  })
8  export class IndexComponent implements OnInit {
9
10     constructor() { }
11
12     ngOnInit(): void {
13     }
14 }
```

```

1  <ul>
2    <li><a target="_blank" [routerLink]='["/ui_messages"]'>Поруке</a></li>
3    <li><a target="_blank" [routerLink]='["/ui_flights"]'>Летови</a></li>
4    <li><a target="_blank" [routerLink]='["/ui_reservations"]'>Резервације</a></li>
5    <li><a target="_blank" [routerLink]='["/index.html"]'>Индекс</a></li>
6  </ul>
7  <ul>
8    <li><a target="_blank" [routerLink]='["/flights/list"]'>Листа летова</a></li>
9    <li><a target="_blank" [routerLink]='["/reservations/list"]'>Листа резервација</a></li>
10   <li><a target="_blank" [routerLink]='["/messages/list/read"]'>Листа прочитаних порука</a></li>
11   <li><a target="_blank" [routerLink]='["/messages/list/non_read"]'>Листа непрочитаних порука</a></li>
12 </ul>
13 <ul>
14   <li><a target="_blank" [routerLink]='["/assets/app_messages/file_not_found.txt"]'>Not found</a></li>
15   <li><a target="_blank" [href]='["http://www.google.com"]'>Google</a></li>
16 </ul>

```

### flight.component.ts

```

1  import { HttpClient } from '@angular/common/http';
2  import { Component, OnInit } from '@angular/core';
3  import { FormBuilder, FormGroup, Validators } from '@angular/forms';
4  import { FlightService } from '../flight-service/flight.service';
5
6  @Component({
7    selector: 'app-flight',
8    templateUrl: './flight.component.html',
9    styleUrls: ['./flight.component.css']
10  })
11  export class FlightComponent implements OnInit {
12    public flightForm: FormGroup;
13    public formBuilder: FormBuilder;
14    public httpClient: HttpClient;
15
16    public oldFlyId : string = "";
17    public neoFlyId : string = "";
18    public flyDate : string = "";
19    public username : string = "";
20    public password : string = "";
21    public relation : string = "";
22    public flightService: FlightService;
23    public flights : any[] = [];

```

```
25     constructor(formBuilder: FormBuilder, httpClient: HttpClient, flightService: FlightService) {
26         this.formBuilder = formBuilder;
27         this.httpClient = httpClient;
28         this.flightService = flightService;
29         this.flightForm = this.formBuilder.group({
30             old_fly_id: ['', Validators.required],
31             neo_fly_id: ['', Validators.required],
32             fly_date: ['', Validators.required]
33         });
34     }
35
36     ngOnInit(): void {
37         this.flightForm = this.formBuilder.group({
38             old_fly_id: ['', Validators.required],
39             neo_fly_id: ['', Validators.required],
40             fly_date: ['', Validators.required]
41         });
42         this.list();
43     }
44
45     submitInsert(){
46         var data = {
47             username: this.username,
48             password: this.password,
49             fly_id: this.oldFlyId,
50             fly_date: this.flyDate,
51             relation: this.relation
52         }
53         this.flightService.insert(data);
54     }
55
56     submitErase(){
57         var data = {
58             username: this.username,
59             password: this.password,
60             fly_id: this.oldFlyId
61         }
62         this.flightService.delete(data);
63         this.password = '';
64     }
```

```
66     submitUpdate(){
67         var data = {
68             username: this.username,
69             password: this.password,
70             old_fly_id: this.oldFlyId,
71             neo_fly_id: this.neoFlyId,
72             fly_date : this.flyDate,
73             relation : this.relation
74         }
75         this.flightSerivce.update(data);
76         this.password = '';
77     }
78
79     list(){
80         this.httpClient.get('/flights/list').subscribe(data => {
81             var dJson = JSON.parse(JSON.stringify(data));
82             this.flights = dJson;
83         });
84     }
85
86     toForm(flight: any): void{
87         var fly_id   = document.getElementById('old_fly_id') as HTMLInputElement;
88         var fly_date = document.getElementById('fly_date') as HTMLInputElement;
89         var relation = document.getElementById('relation') as HTMLInputElement;
90         fly_id.value = flight.flightId;
91         fly_date.value= flight.flightDate;
92         relation.value = flight.relation;
93         this.oldFlyId = flight.flightId;
94         this.flyDate = flight.flightDate;
95         this.relation = flight.relation;
96     }
97 }
```

## flight.component.html

```
1 <h2>Flight works!</h2>
2 <hr>
3 <form [formGroup]="flightForm">
4   <div class="flight-form">
5     <label for="username">Занослени : </label><br>
6     <input type="text" id='username' name="username" [(ngModel)]='username' [ngModelOptions]='{standalone: true}' required><br><br>
7     <label for="password">Лозинка : </label><br>
8     <input type="password" id='password' name="password" required [(ngModel)]='password' [ngModelOptions]='{standalone: true}'><br><br>
9     <label for="old_fly_id">Идентификација летова : </label><br>
10    <input type="text" id='old_fly_id' name="old_fly_id" [(ngModel)]='oldFlyId' [ngModelOptions]='{standalone: true}' required><br><br>
11    <label for="neo_fly_id">Нова идентификација летова : </label><br>
12    <input type="text" id='neo_fly_id' name="neo_fly_id" [(ngModel)]='neoFlyId' [ngModelOptions]='{standalone: true}' required><br><br>
13    <label for="fly_date">Времена/датуми летова : </label><br>
14    <textarea required id='fly_date' name='fly_date' [(ngModel)]='flyDate' [ngModelOptions]='{standalone: true}'></textarea><br><br>
15    <label for="relation">Релација : </label><br>
16    <textarea required id='relation' name='relation' [(ngModel)]='relation' [ngModelOptions]='{standalone: true}'></textarea><br>
17  </div>
18  <br>
19  <button type="submit" name='submit_insert' (click)='submitInsert()'>Додавање</button>
20  <button type="submit" name='submit_erase' (click)='submitErase()'>Брисање</button>
21  <button type="submit" name='submit_update' (click)='submitUpdate()'>Измена</button>
22 </form>
23 <hr>
24
25 <table>
26   <tr>
27     <th>Лет</th>
28     <th>Врејеме</th>
29     <th>Релација</th>
30   </tr>
31   <tr *ngFor='let flight of flights' (click)='toForm(flight)'>
32     <td>{{ flight.flightId }}</td>
33     <td>{{ flight.flightDate }}</td>
34     <td>{{ flight.relation }}</td>
35   </tr>
36 </table>
```

## messages.component.ts

```
1  import { HttpClient } from '@angular/common/http';
2  import { Component, OnInit } from '@angular/core';
3
4  @Component({
5    selector: 'app-messages',
6    templateUrl: './messages.component.html',
7    styleUrls: ['./messages.component.css']
8  })
9  export class MessagesComponent implements OnInit {
10    public httpClient: HttpClient;
11    public read_messages : any[] = [];
12    public non_read_messages : any[] = [];
13    public username: string = "";
14    public password: string = "";
15
16    constructor(httpClient: HttpClient) {
17      this.httpClient = httpClient;
18    }
19
20    ngOnInit(): void {
21      this.list_read_messages();
22      this.list_non_read_messages();
23    }
24
25    list_read_messages(): void{
26      this.httpClient.get('/messages/list/read').subscribe(data => {
27        var dJson = JSON.parse(JSON.stringify(data));
28        this.read_messages = dJson;
29      });
30    }
31
32    list_non_read_messages(): void{
33      this.httpClient.get('/messages/list/non_read').subscribe(data => {
34        var dJson = JSON.parse(JSON.stringify(data));
35        this.non_read_messages = dJson;
36      });
37    }
38  }
```

## messages.component.html

```
1 <h2>Messages works!</h2>
2 <hr>
3 <label for="username">Занослени : </label><br>
4 <input type="text" id="username" name="username" [(ngModel)]="username" [ngModelOptions]="{standalone: true}" required><br><br>
5 <label for="password">Лозинка : </label><br>
6 <input type="password" id="password" name="password" required [(ngModel)]="password" [ngModelOptions]="{standalone: true}"><br><br>
7 <hr>
8 <p>НЕПРОЧИТАНЕ ПОРУКЕ</p>
9 <div *ngFor="let reservation of non_read_messages">
10   <table class="data_table">
11     <tr>
12       <td style="width: 100px">Порука: </td>
13       <td><code>{{ reservation.message.messageId }}</code></td>
14     </tr>
15     <tr>
16       <td style="width: 100px">Корисник: </td>
17       <td><code>{{ reservation.message.username }}</code></td>
18     </tr>
19     <tr>
20       <td style="width: 100px">Створено: </td>
21       <td><code>{{ reservation.fmt.created }}</code></td>
22     </tr>
23     <tr>
24       <td style="width: 100px">Промјењено: </td>
25       <td><code>{{ reservation.fmt.changed }}</code></td>
26     </tr>
27     <tr>
28       <td style="width: 100px">Е-Пошта: </td>
29       <td><code><a style="text-decoration: none;" *ngIf="reservation.email.address != ''" [href]="['mailto:']+reservation.email.address">
30         </tr>
31       <tr><td colspan="2">Садржај: </td></tr>
32       <tr><td colspan="2"><code style="white-space: pre;">{{ reservation.message.messageContent }}</code></td></tr>
33     </table>
34     <button (click)="sentRead(reservation.message.messageId)">Прочитано</button>
35     <br><br>
36 </div>
```

```
~{{ reservation.email.address }}</a></code></td>
```

```
<p>ПРОЧИТАНЕ ПОРУКЕ</p>
<div *ngFor="let reservation of read_messages">
  <table class="data_table">
    <tr>
      <td style="width: 100px">Порука: </td>
      <td><code>{{ reservation.message.messageId }}</code></td>
    </tr>
    <tr>
      <td style="width: 100px">Корисник: </td>
      <td><code>{{ reservation.message.username }}</code></td>
    </tr>
    <tr>
      <td style="width: 100px">Створено: </td>
      <td><code>{{ reservation.fmt.created }}</code></td>
    </tr>
    <tr>
      <td style="width: 100px">Промјењено: </td>
      <td><code>{{ reservation.fmt.changed }}</code></td>
    </tr>
    <tr>
      <td style="width: 100px">Е-Пошта: </td>
      <td><code><a style="text-decoration: none;" *ngIf="reservation.email.address != ''" [href]="['mailto:']+reservation.email.address">
    </tr>
    <tr><td colspan="2">Садржај: </td></tr>
    <tr><td colspan="2"><code style="white-space: pre;">{{ reservation.message.messageContent }}</code></td></tr>
  </table>
  <br>
</div>
```



## reservations.components.ts

```
1 import { HttpClient } from '@angular/common/http';
2 import { Component, OnInit, Output } from '@angular/core';
3 import { FormBuilder, FormGroup, Validators } from '@angular/forms';
4 import { ReservationsService } from '../reservations-service/reservations.service';
5
6 @Component({
7   selector: 'app-reservations',
8   templateUrl: './reservations.component.html',
9   styleUrls: ['./reservations.component.css']
10 })
11 export class ReservationsComponent implements OnInit {
12   public reservationsForm: FormGroup;
13   public formBuilder: FormBuilder;
14   public httpClient: HttpClient;
15   public reservationsService: ReservationsService;
16   public reservations : any[] = [];
17
18   public username: string = "";
19   public password: string = "";
20   public old_reservation_id: string = "";
21   public neo_reservation_id: string = "";
22   public fly_id: string = "";
23   public target_username: string = "";
24   public article_description: string = "";
25   public article_transport_spec_file: File| null = null;
26   public place_count: number = -1;
27
28   constructor(formBuilder: FormBuilder, httpClient: HttpClient, reservationsService: ReservationsService) {
29     this.formBuilder = formBuilder;
30     this.httpClient = httpClient;
31     this.reservationsService = reservationsService;
32     this.reservationsForm = this.formBuilder.group({
33       place_count: [Validators.required, Validators.min(-1)]
34     });
35   }
36 }
```

```
37     ngOnInit(): void {
38         this.list();
39     }
40
41     submitInsert(): void{
42         var data = {
43             username: this.username,
44             password: this.password,
45             reservation_id : this.old_reservation_id,
46             fly_id: this.fly_id,
47             target_username: this.target_username,
48             article_description: this.article_description,
49             article_transport_spec_file: this.article_transport_spec_file,
50             place_count: this.place_count
51         }
52         this.reservationsService.insert(data);
53     }
54
55     submitErase(): void{
56         var data = {
57             username: this.username,
58             password: this.password,
59             reservation_id : this.old_reservation_id,
60         }
61         this.reservationsService.delete(data);
62     }
63
64     submitResetFile(): void{
65         var data = {
66             username: this.username,
67             password: this.password,
68             reservation_id : this.old_reservation_id,
69         }
70         this.reservationsService.resetFile(data);
71     }
```

```

submitUploadFile(): void{
    var data = {
        username: this.username,
        password: this.password,
        reservation_id : this.old_reservation_id,
    }
    this.reservationsService.uploadFile(data, this.article_transport_spec_file);
}

submitUpdate(): void{
    var data = {
        username: this.username,
        password: this.password,
        reservation_id : this.old_reservation_id,
        neo_reservation_id: this.neo_reservation_id,
        fly_id: this.fly_id,
        target_username: this.target_username,
        article_description: this.article_description,
        article_transport_spec_file: this.article_transport_spec_file,
        place_count: this.place_count
    }
    this.reservationsService.update(data);
}

handleFileInput(event: any): void {
    let fileInput = document.getElementById('article_transport_spec_file') as HTMLInputElement;
    var files = fileInput.files;
    if(files==null)
        this.article_transport_spec_file = null;
    else if(files.length==1)
        this.article_transport_spec_file = files.item(0);
    else
        this.article_transport_spec_file = null;
}

list(){
    this.httpClient.get('/reservations/list').subscribe(data => {
        var dJson = JSON.parse(JSON.stringify(data));
        this.reservations = dJson;
    });
}

```

```
115     toForm(reservation: any): void{
116         var old_reservation_id = document.getElementById('old_reservation_id') as HTMLInputElement;
117         var fly_id = document.getElementById('fly_id') as HTMLInputElement;
118         var target_username = document.getElementById('target_username') as HTMLInputElement;
119         var article_description = document.getElementById('article_description') as HTMLInputElement;
120         var place_count = document.getElementById('place_count') as HTMLInputElement;
121         old_reservation_id.value = reservation.reservationId;
122         fly_id.value = reservation.flyId;
123         target_username.value = reservation.username;
124         article_description.value = reservation.articleDescription;
125         place_count.value = reservation.placeCount;
126         this.old_reservation_id = reservation.reservationId;
127         this.fly_id = reservation.flyId;
128         this.target_username = reservation.username;
129         this.article_description = reservation.articleDescription;
130         this.place_count = reservation.placeCount;
131     }
132
133     generateLink(reservation_id: string):string{
134         return "/reservations/download_ast?reservation_id="+encodeURIComponent(reservation_id);
135     }
136 }
```

## reservations.components.html

```
1 <h2>Reservations works!</h2>
2 <hr>
3 <form [formGroup]="reservationsForm">
4   <label for="username">Занослени : </label><br>
5   <input type="text" id="username" name="username" [(ngModel)]="username" [ngModelOptions]="{standalone: true}" required><br><br>
6   <label for="password">Лозинка : </label><br>
7   <input type="password" id="password" name="password" required [(ngModel)]="password" [ngModelOptions]="{standalone: true}"><br><br>
8   <label for="old_reservation_id">Идентификација резервације : </label><br>
9   <input type="text" id="old_reservation_id" name="old_reservation_id" [(ngModel)]="old_reservation_id" [ngModelOptions]="{standalone: true}" required><br><br>
10  <label for="neo_reservation_id">Нова идентификација резервације : </label><br>
11  <input type="text" id="neo_reservation_id" name="neo_reservation_id" [(ngModel)]="neo_reservation_id" [ngModelOptions]="{standalone: true}" required><br><br>
12  <label for="fly_id">Идентификација лета : </label><br>
13  <input type="text" id="fly_id" name="fly_id" [(ngModel)]="fly_id" [ngModelOptions]="{standalone: true}" required><br><br>
14  <label for="target_username">Циљни корисник : </label><br>
15  <input type="text" id="target_username" name="target_username" [(ngModel)]="target_username" [ngModelOptions]="{standalone: true}" required><br><br>
16  <label for="place_count">Број мјеста за путнике : </label><br>
17  <input type="text" id="place_count" name="place_count" [(ngModel)]="place_count" [ngModelOptions]="{standalone: true}" required class=""><br><br>
18  <label for="article_description">Опис робе за превоз : </label><br>
19  <textarea id="article_description" name="article_description" [(ngModel)]="article_description" [ngModelOptions]="{standalone: true}" required><br><br></textarea><br><br>
20  <label for="article_transport_spec_file">Датотека спецификације превоза робе : </label><br>
21  <input type="file" id="article_transport_spec_file" name="article_transport_spec_file" (change)="handleFileInput($event.target)" required><br><br><br>
22  <button type="submit" name="submit_insert" (click)="submitInsert()">Додавање</button>
23  <button type="submit" name="submit_erase" (click)="submitErase()">Брисање</button>
24  <button type="submit" name="submit_update" (click)="submitUpdate()">Измјена</button>
25  <button type="submit" name="submit_upload_file" (click)="submitUploadFile()">Постављање датотеке</button>
26  <button type="submit" name="submit_reset_file" (click)="submitResetFile()">Поништавање датотеке</button>
27 </form>
28 <hr>
29 <table>
30   <tr>
31     <th>Резервација</th>
32     <th>корисник</th>
33     <th>лет</th>
34     <th>број путника</th>
35     <th>опис робе</th>
36     <th>спецификација транспорта</th>
37   </tr>
38   <tr *ngFor="let reservation of reservations">
39     <td (click)="toForm(reservation)">{{ reservation.reservationId }}</td>
40     <td (click)="toForm(reservation)">{{ reservation.username }}</td>
41     <td (click)="toForm(reservation)">{{ reservation.flyId }}</td>
42     <td (click)="toForm(reservation)">{{ reservation.placeCount }}</td>
43     <td (click)="toForm(reservation)">{{ reservation.articleDescription }}</td>
44     <td><a [href]="generateLink(reservation.reservationId)" target="_blank">{{ reservation.fmt.astf_name }}</a></td>
45   </tr>
46 </table>
```

Activate Windows  
Go to Settings to activate W

## messages.componnet.css

```
1 td, th {
2   border: 0px solid black;
3   padding : 0px 0px 0px 0px;
4   text-align: left;
5 }
6
7 table.data_table{
8   border: 1px solid black;
9   background-color:lightcyan;
10  width:100%;
11 }
```

## reservations.component.css

```
1  td, th {  
2    border: 1px solid black;  
3    padding : 5px 5px 5px 5px;  
4    text-align: left;  
5  }
```

## flights.component.css

```
1  td, th {  
2    border: 1px solid black;  
3    padding : 5px 5px 5px 5px;  
4    text-align: left;  
5  }
```