# Uputstvo za izradu Angular projekta

## Instalacija I kreiranje projekta

Da bismo mogli da kreiramo nas Angular projekat koji ce predstavljati Frontend prvo su nam potrebni:

* **Node:**
  + <https://nodejs.org/en/download/>
* **Angular CLI:**
  + Pokrenemo cmd/terminal a zatim unesemo komande:

**npm uninstall -g @angular/cli -** uninstaliramo staru verziju

**npm install -g @angular/cli@latest –** instaliramo najnoviju verziju

Kada smo instalirali Angular sledece sto treba da uradimo je da u cmd/terminalu **predjemo u folder** u kojem zelimo da kreiramo nas Angular projekat.

**Kreiranje projekta** se izvrsava sledecom komandom:

**ng new NazivAngularProjekta**

Dobicemo upit o tome da li zelimo routing na sta mi odnosno unosimo **N**.

## Pokretanje projekta

Projekat se pokrece komandom:

**ng serve**

Ovo ce pokrenuti aplikaciju I pruziti nam URL aplikacije.

Takodje se moze dodati opcioni parametar **--o** koji radi istu stvar s tim sto automatski otvara aplikaciju u nasem default brower-u.

## Struktura projekta

Komponente predstavljaju osnovni blok strukture Angular projekta.

Postoji 3 vrste komponenti koje su nama trenutno od znacaja:

* **Klasicne komponente:**

ng g c uplata -s --skip-tests

ng g c uplata/uplata-form -s --skip-tests

* **Service:**

ng g s shared/uplata --skip-tests

* **Modalna klasa:**

ng g class shared/uplata --type=model --skip-tests

Svaka komponenta se sastoji iz 4 fajla (HTML, CSS, TS – typescript i test fajla).

Za sad nam nisu potrebni test fajlovi tako da ih preskacemo sa **--skip-tests.**

Za klasicne komponente nam nisu bili potrebni .css falovi tako da smo I njih preskocili sa –**s.**

## Podesavanje projekta

U **tsconfig.json** u prvi blok dodamo novi red:

"strictPropertyInitialization": false,

U **app.module.ts dodamo**:

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

Takodje dodamo HttpClientModule u imports deo.

Za objekat koji saljemo/primamo iz base kreiramo modalnu komponentu (export klasa).

Primer modalne klase:

export class Osoba{

id:number = 0;

ime:string = “”;

prezime:string = “”;

}

Dodajemo **HttpClientModule** I **FormsModule** u imports blok unutar **app.module.ts**.

U parametre konstruktora nase service komponente dodajemo **private http:HttpClient** I importujemo HttpClient pomocu **import {HttpClient} from '@angular/common/http';**

*Sve ovo ide u klasu unutar nase service typescript komponente za trenutn model.*

formData: Osoba = new Osoba ();

list:Osoba[];

readonly baseURL = 'https://localhost:brojPorta/api/NazivKontroleraBackend-a';

### **POST, GET I DELETE metode**

postOsoba (){

return this.http.post(this.baseURL, this.formData);

}

getOsobe(){

this.http.get(this.baseURL)

.toPromise()

.then(res => this.list = res as Osoba[]);

}

deleteOsoba(id:number){

return this.http.delete(`${this.baseURL}/${id}`);

}

Svi HTML fajlovi komponenti mogu se “ugnjezditi” u **app.component.html** kao I u druge komponente.

Ovo se omogucava unosenjem HTML tag-a sa nazivom koji se nalazi u Selector-u unutar @Componente typescript fajla koji ugnjezdavamo.

Primer:

<div>

<h2> Unos nove osobe </h2>

**<app-osoba-forma> </app-osoba-forma>**

</div>

U ovom slucaju ceo sadrzaj unutar app-osoba-forma komponente ce biti unet u ovaj deo gde ce zameniti <app-osoba-forma> </app-osoba-forma>.

### Izgled forme za unos novog objekta (binding forme sa objektom)

<form **#form="ngForm" (submit)="funkcijaZaObraduPostZahteva(form)"**>

<input type="hidden" **name="id"** **[value]="service.formData.id"** />

<input **name="ime"** **#ime="ngModel”** **[(ngModel)]="service.formData.ime"**>

<button **type="submit"**>Unesi</button>

</form>