



Dokumentacija sustava

Upute za uspješno korištenje

Irena Ilišević

1. Sadržaj

2.	Opis sustava	4
3.	Komponente i moduli	5
i.	AppModule	5
ii.	AdminLayoutModule	5
iii.	AppRoutingModule	6
iv.	ComponentsModule	6
v.	AdminLayoutComponent	7
vi.	AppComponent	7
vii.	LoginComponent	7
viii.	LogoutComponent	8
ix.	ArtiklEditComponent	8
x.	ArtiklNoviComponent	9
xi.	ArtikliComponent	9
xii.	DashboardComponent	10
xiii.	KorisnikEditComponent	10
xiv.	KorisniciComponent	11
xv.	KorisnikNoviComponent	11
xvi.	ModalComponent	11
xvii.	PartnerEditComponent	12
xviii.	PartnerNoviComponent	12
xix.	PartneriComponent	13
xx.	RacunNoviComponent	13
xxi.	RacuniComponent	15
xxii.	VrstaRacunaEditComponent	15
xxiii.	VrstaRacunaNoviComponent	16
xxiv.	VrsteRacunaComponent	16
xxv.	FooterComponent	17
xxvi.	NavbarComponent	17
xxvii.	SidebarComponent	17
4.	Klase	19
i.	Artikl	19
ii.	Partner	19
iii.	Stavka	19
iv.	VrstaRacuna	20
5.	Servisi	21
i.	Auth-guard	21
ii.	Authentication	21

6. Routing i navigacijski dijagram.....	22
---	----

2. Opis sustava

Sustav POS App namijenjen je vođenju jednostavne blagajne. Aplikacija omogućava korisniku pregled, unos i izmjenu podatka o artiklima, partnerima, korisnicima, vrstama računa i računima. Program nudi brzo i jednostavno kreiranje i izdavanje računa te njihovo ispisivanje. Sučelje je intuitivno i moderno te vrlo jednostavno za korištenje. Za pristup mogućnostima, korisnik se mora prijaviti te nakon toga može započeti s radom. Osim temeljnih funkcija dodavanja, izmjene i pregleda informacija o artiklima, korisnicima, partnerima i računima, korisniku su dostupne i opcije pretraživanja i sortiranja podataka. Osim toga, prisutan je i prikaz jednostavne statistike.

3. Komponente i moduli

Sustav se sastoji od 4 modula:

1. AppModule
2. AdminLayoutModule
3. AppRoutingModule
4. ComponentsModule

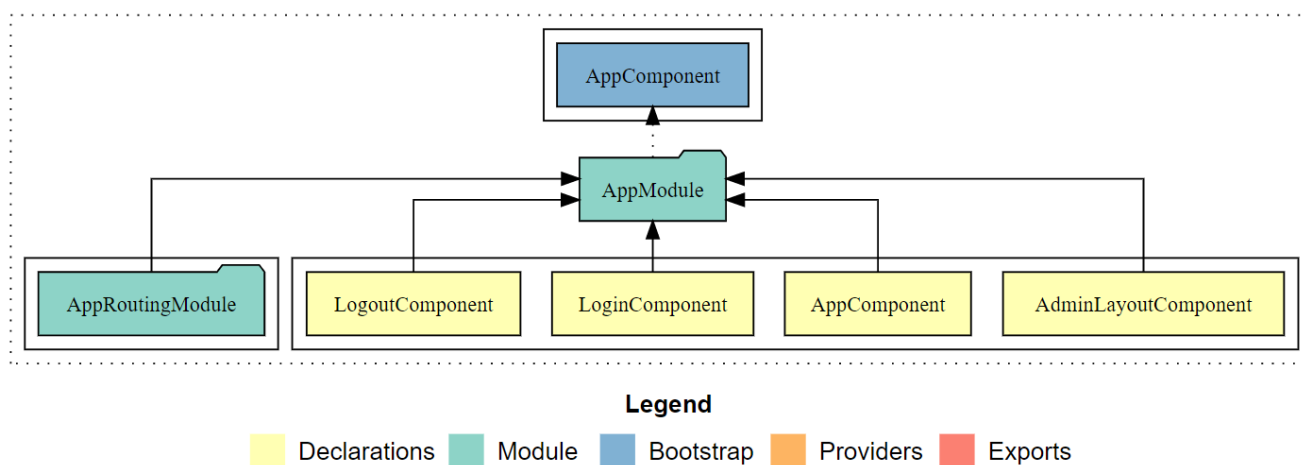
i. AppModule

Osnovni modul sustava. Deklarira ključne komponente za prijavu, odjavu, pokretanje aplikacije te komponentu za početak rada s AdminLayoutModulom.

Deklaracije:

- AdminLayoutComponent
- AppComponent
- LoginComponent
- LogoutComponent

Dijagram modula:



ii. AdminLayoutModule

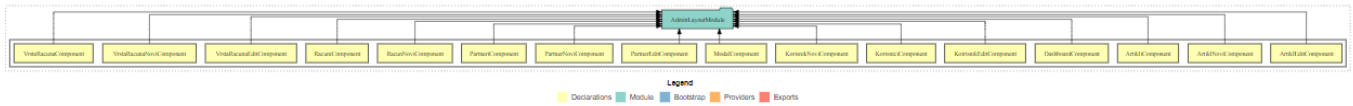
Osnovni modul za prikaz i rad s funkcionalnostima sustava. Deklarira sve komponente za rad.

Deklaracije:

- ArtiklEditComponent
- ArtiklNoviComponent
- ArtikliComponent
- DashboardComponent
- KorisnikEditComponent
- KorisniciComponent
- KorisnikNoviComponent
- ModalComponent
- PartnerEditComponent
- PartnerNoviComponent

- PartneriComponent
- RacunNoviComponent
- RacuniComponent
- VrstaNovComponent
- VrstaNoviComponent
- VrstaNovaComponent

Dijagram modula:



iii. AppRoutingModule

Osnovni modul za preusmjeravanje. Definira rute kroz aplikaciju pozivajući `AdminLayoutComponent` te rute za login i logout komponente.

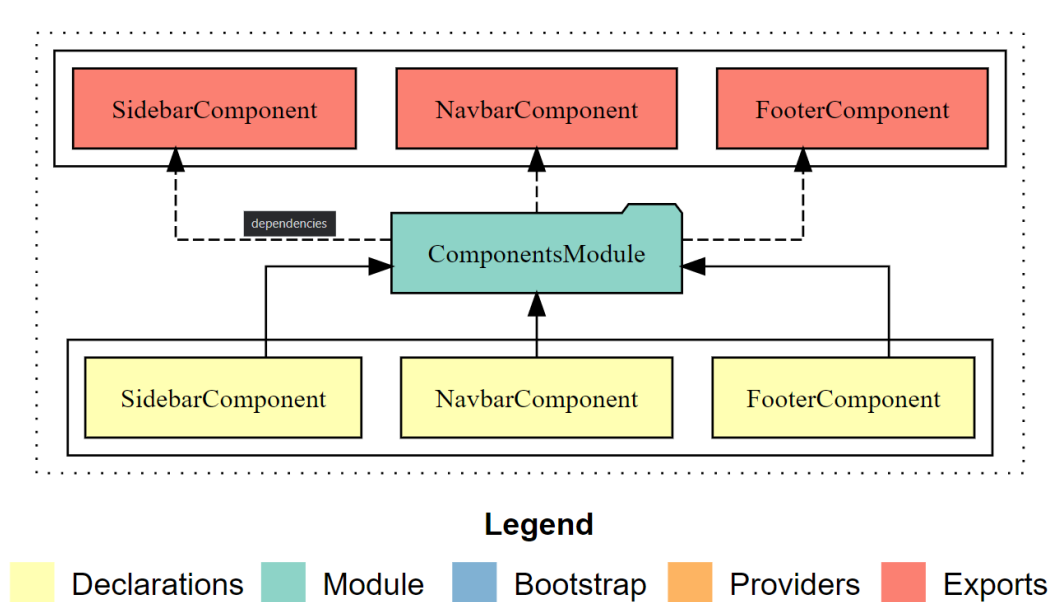
iv. ComponentsModule

Modul koji deklarira komponente navigacije i podnožja.

Deklaracije:

- FooterComponent
- NavbarComponent
- SidebarComponent

Dijagram modula:



v. AdminLayoutComponent

Početna komponenta. Definira funkcije za kretanje po glavnom izborniku i za mijenjanje dizajna prilikom klika na pojedine elemente izbornika. Definira funkcije za responzivni dizajn.

selector	app-admin-layout
styleUrls	./admin-layout.component.scss
templateUrl	./admin-layout.component.html

Atributi:

Private: _router (tip podatka: Subscription), lastPoppedUrl (tip podatka: string), yScrollStack (tip podatka: number[])

Public: location (tip podatka: Location)

Metode:

isMac() – vraća boolean – provjerava vrstu uređaja, odnosno operacijskog sustava.

isMaps() – vraća boolean, prima 1 parametar path

ngAfterViewInit() – vraća void

ngOnInit() – vraća void

runOnRouteChange() – vraća void

vi. AppComponent

Root komponenta.

selector	app-root
styleUrls	./app.component.css
templateUrl	./app.component.html

Nema atributa ni metoda.

vii. LoginComponent

Komponenta za prijavu u aplikaciju. Definira metode za provjeru korisnika.

selector	app-login
styleUrls	./login.component.css
templateUrl	./login.component.html

Atributi:

invalidLogin (tip podatka: Boolean), password (tip podatka: string), username (tip podatka: string)

Metode:

checkLogin() – vraća void – poziva servis za autentifikaciju korisnika, te u ovisnosti o odgovoru preusmjerava na početnu stranicu, odnosno ostaje na prijavi ukoliko autentifikacija nije uspjela.

ngOnInit() – vraća void

viii. LogoutComponent

Komponenta za odjavu korisnika. Poziva autentifikacijski servis i njegovu metodu logOut() i preusmjerava na stranicu za prijavu.

selector	app-logout
styleUrls	./logout.component.css
templateUrl	./logout.component.html

Nema atributa.

Metode:

ngOnInit() – vraća void – poziva autentifikacijski servis i preusmjerava.

ix. ArtiklEditComponent

Komponenta za uređivanje podataka o odabranom artiklu. Definira metode za čitanje podatka o artiklu i zapisivanje novih podataka.

selector	artikl-edit
templateUrl	./artikl-edit.component.html

Atributi:

Artikl (tip podatka: any) – pohranjuje podatke o artiklu koji se uređuje (dobiveni s API-ja),

Atributi za spremanje novih podataka o artiklu: Broj (tip podatka: number), Cijena (tip podatka: number), Id (tip podatka: any), IdArtikla (tip podatka: number), JedMjere (tip podatka: string), Naziv (tip podatka: string), Porez (tip podatka: number).

Metode:

ngOnInit() – vraća void – dohvaća podatke s API-ja o artiklu koji se uređuje prilikom inicijalizacije komponente

prikaziObavijest() – vraća void – generira obavijest o uspješnom spremanju promjena

spremiPromjene() – vraća void – šalje promijenjene podatke o artiklu na API za spremanje u bazu podataka.

Konstruktor:

Parametri: route (tip podatka: ActivatedRoute), http (tip podatka: HttpClient).

x. ArtiklNoviComponent

Komponenta za dodavanje novog artikla. Definira metode za spremanje podataka o novom artiklu.

selector	artikl-novi
templateUrl	./artikl-novi.component.html

Atributi:

broj (tip podatka: number), cijena (tip podatka: number), idArtikla (tip podatka: number), jed_mjere (tip podatka: string), naziv (tip podatka: string), porez (tip podatka: number).

Metode:

ngOnInit() – vraća void

preusmjeri() – vraća void – preusmjerava na prikaz svih artikala nakon uspješnog spremanja

prikaziObavijest() – vraća void – generira obavijest o uspješnom spremanju novog artikla

spremiPromjene() – vraća void – šalje podatke o novom artiklu na API za spremanje u bazu podataka

Konstruktor:

Parametri: route (tip podatka: ActivatedRoute), http (tip podatka: HttpClient), _router (tip podatka: Router)

xi. ArtikliComponent

Komponenta za prikaz svih artikala. Definira metode za dohvaćanje podataka o artiklima s API-ja te metode za pretraživanje i sortiranje podataka.

selector	artikli
templateUrl	./artikli.component.html

Atributi:

Artikli (tip podataka: []), Filter (tip podataka: string), key (tip podataka: string), p (tip podataka: number), reverse (tip podataka: boolean).

Metode:

ngOnInit() – vraća void – dohvaća podatke o svim artiklima s API-ja za vrijeme inicijalizacije komponente

preusmjeriEdit() – vraća void, prima parametar id_artikla – preusmjerava na uređivanje odabranog artikla

preusmjeriNovi() – vraća void – preusmjerava na kreiranje novog artikla

sort() – vraća void, prima parametar key – sortira podatke prema dobivenom ključu.

Konstruktor:

Parametri: http (tip podatka: HttpClient), _router (tip podatka: Router)

xii. DashboardComponent

Komponenta za prikaz početne stranice. Definira metode za prikaz grafa s podacima i prikaz količine podataka u obliku kartica.

selector	app-dashboard
styleUrls	./dashboard.component.css
templateUrl	./dashboard.component.html

Atributi:

artikli (tip podatka: []), brojArtikala (tip podatka: number), brojKorisnika (tip podatka: number), brojPartnera (tip podatka: number), brojRacuna (tip podatka: number), korisnici (tip podatka: []), partneri (tip podatka: []), podaci (tip podatka: []), racuni (tip podatka: [])

Metode:

Crtaj() – vraća void, prima parametre labels i series – crta grafički prikaz

ngOnInit() – vraća void – dohvaća podatke s API-ja prilikom inicijalizacije komponente

startAnimationForBarChart() – vraća void, prima parametar chart – animira graf

Konstruktor:

Parametri: http (tip podatka: HttpClient)

xiii. KorisnikEditComponent

Komponenta za uređivanje podataka o odabranom korisniku. Definira metode za čitanje podatka o korisniku i zapisivanje novih podataka.

selector	korisnik-edit
templateUrl	./korisnik-edit.component.html

Atributi:

Korisnik (tip podatka: any) – pohranjuje podatke o korisniku koji se uređuje (dobiveni s API-ja),

Atributi za spremanje novih podataka o korisniku: idKorisnika (tip podatka: number), korime (tip podatka: string), lozinka (tip podatka: string)

Metode:

ngOnInit() – vraća void – dohvaća podatke s API-ja o korisniku koji se uređuje prilikom inicijalizacije komponente

prikaziObavijest() – vraća void – generira obavijest o uspješnom spremanju promjena

spremiPromjene() – vraća void – šalje promijenjene podatke o korisniku na API za spremanje u bazu podataka.

Konstruktor:

Parametri: route (tip podatka: ActivatedRoute), http (tip podatka: HttpClient).

xiv. KorisniciComponent

Komponenta za prikaz svih korisnika. Definira metode za dohvaćanje podataka o korisnicima s API-ja te metode za pretraživanje i sortiranje podataka.

selector	<code>korisnici</code>
templateUrl	<code>./korisnici.component.html</code>

Atributi:

Korisnici (tip podataka: []), Filter (tip podataka: string), key (tip podataka: string), p (tip podataka: number), reverse (tip podataka: boolean).

Metode:

ngOnInit() – vraća void – dohvaća podatke o svim korisnicima s API-ja za vrijeme inicijalizacije komponente
 preusmjeriEdit() – vraća void, prima parametar id_korisnika – preusmjerava na uređivanje odabranog korisnika
 preusmjeriNovi() – vraća void – preusmjerava na kreiranje novog korisnika
 sort() – vraća void, prima parametar key – sortira podatke prema dobivenom ključu.

Konstruktor:

Parametri: http (tip podatka: HttpClient), _router (tip podatka: Router)

xv. KorisnikNoviComponent

Komponenta za dodavanje novog korisnika. Definira metode za spremanje podataka o novom korisniku.

selector	<code>korisnik-novi</code>
templateUrl	<code>./korisnik-novi.component.html</code>

Atributi:

idKorisnika (tip podatka: number), korime (tip podatka: string), lozinka (tip podatka: string).

Metode:

ngOnInit() – vraća void
 preusmjeri() – vraća void – preusmjerava na prikaz svih korisnika nakon uspješnog spremanja
 prikaziObavijest() – vraća void – generira obavijest o uspješnom spremanju novog korisnika
 spremiPromjene() – vraća void – šalje podatke o novom korisniku na API za spremanje u bazu podataka

Konstruktor:

Parametri: route (tip podatka: ActivatedRoute), http (tip podatka: HttpClient), _router (tip podatka: Router)

xvi. ModalComponent

Komponenta za generiranje modalnog dijaloga za prikaz print verzije računa. Definira metode za generiranje prikaza.

selector	app-modal
styleUrls	./modal.component.css
templateUrl	./modal.component.html

Atributi:

dialogRef (tip podatka: MatDialogRef<ModalComponent>), iznos (tip podatka: number), racuni (tip podatka: []), racunId (tip podatka: string), stavke (tip podatka: [])

Metode:

closeModal() – vraća void – zatvara modalni dijalog

ngOnInit() – vraća void – dohvaća podatke za prikaz s API-ja.

Konstruktor:

Parametri: dialogRef (tip podatka: MatDialogRef<ModalComponent>), http (tip podatka: HttpClient)

xvii. PartnerEditComponent

Komponenta za uređivanje podataka o odabranom partneru. Definira metode za čitanje podatka o partneru i zapisivanje novih podataka.

selector	partner-edit
templateUrl	./partner-edit.component.html

Atributi:

partner (tip podatka: any) – pohranjuje podatke o partneru koji se uređuje (dobiveni s API-ja),

Atributi za spremanje novih podataka o partneru: adresa (tip podatka: string), broj (tip podatka: number), id (tip podatka: any), idPartnera (tip podatka: number), mjesto (tip podatka: string), naziv (tip podatka: string)

Metode:

ngOnInit() – vraća void – dohvaća podatke s API-ja o partneru koji se uređuje prilikom inicijalizacije komponente

prikaziObavijest() – vraća void – generira obavijest o uspješnom spremanju promjena

premiPromjene() – vraća void – šalje promijenjene podatke o partneru na API za spremanje u bazu podataka.

Konstruktor:

Parametri: route (tip podatka: ActivatedRoute), http (tip podatka: HttpClient).

xviii. PartnerNoviComponent

Komponenta za dodavanje novog partnera. Definira metode za spremanje podataka o novom partneru.

selector	partner-novi
templateUrl	./partner-novi.component.html

Atributi:

adresa (tip podatka: string), broj (tip podatka: string), idPartnera (tip podatka: number), mjesto (tip podatka: string), naziv (tip podatka: string)

Metode:

ngOnInit() – vraća void

preusmjeri() – vraća void – preusmjerava na prikaz svih partnera nakon uspješnog spremanja

prikaziObavijest() – vraća void – generira obavijest o uspješnom spremanju novog partnera

spremiPromjene() – vraća void – šalje podatke o novom partneru na API za spremanje u bazu podataka

Konstruktor:

Parametri: route (tip podatka: ActivatedRoute), http (tip podatka: HttpClient), _router (tip podatka: Router)

xix. PartneriComponent

Komponenta za prikaz svih partnera. Definira metode za dohvaćanje podataka o partnerima s API-ja te metode za pretraživanje i sortiranje podataka.

selector	partneri
templateUrl	./partneri.component.html

Atributi:

partneri (tip podatka: []), Filter (tip podatka: string), key (tip podatka: string), p (tip podatka: number), reverse (tip podatka: boolean).

Metode:

ngOnInit() – vraća void – dohvaća podatke o svim partnerima s API-ja za vrijeme inicijalizacije komponente

preusmjeriEdit() – vraća void, prima parametar id_partnera – preusmjerava na uređivanje odabranog partnera

preusmjeriNovi() – vraća void – preusmjerava na kreiranje novog partnera

sort() – vraća void, prima parametar key – sortira podatke prema dobivenom ključu.

Konstruktor:

Parametri: http (tip podatka: HttpClient), _router (tip podatka: Router)

xx. RacunNoviComponent

Komponenta za kreiranje novog računa. Definira metode za spremanje podataka o novom računu.

selector	racun-novi
templateUrl	./racun-novi.component.html

Atributi:

Atributi stavke računa: nazivArtikla (tip podatka: string), kolicina (tip podatka: number), cijena (tip podatka: number), pdv (tip podatka: number), osnovica (tip podatka: number), iznosPDV (tip podatka: number), ukupno (tip podatka: number), broj (tip podatka: string)

Atributi računa: idRacuna (tip podatka: number), brojRacuna (tip podatka: number), datumRacuna (tip podatka: any), korisnik (tip podatka: string), vrstaRacuna (tip podatka: string), partner (tip podatka: string), napomena (tip podatka: string)

Atributi za izračunavanje iznosa računa: osnovicaRacuna (tip podatka: number), iznospdvRacuna (tip podatka: number), iznosRacuna (tip podatka: number)

Atributi za form controls: myControl (tip podatka: FormControl), myControl2 (tip podatka: FormControl), myControl3 (tip podatka: FormControl)

Atributi za podatke s API-ja: artikli (tip podatka: []), partnerItems (tip podatka: []), vrsteRacuna (tip podatka: [])

Atributi za filtrirane podatke za autocomplete: filteredOptions (tip podatka: Observable<string[]>), filteredPartners (tip podatka: Observable<string[]>)

Atributi za uređene podatke s API-ja: options (tip podatka: []), vrste (tip podatka: []), partneri (tip podatka: [])

Lista za stavke računa: stavke (tip podatka: Array<Stavka>)

Objekti: trenutnaStavka (tip podatka: Stavka), trenutniPartner (tip podatka: Partner), trenutnaVrsta (tip podatka: VrstaRacuna)

Id novokreiranog računa: returnID (tip podatka: any)

Metode:

_filter() – vraća string[], prima parametar value – filtrira podatke za autocomplete artikala

dodajStavku() – vraća void – dodaje stavku u listu svih stavki

filter() – vraća void, prima parametar value – filtrira podatke za autocomplete partnera

focusChange() – vraća void, prima parametar artikl – sprema podatke o artiklu u trenutnu stavku i popunjava input polja prema podacima o odabranom artiklu

getOptionText() – vraća any, prima parametar option – vraća prikaz broja artikla umjesto objekta za autocomplete

getOptionText2() – vraća any, prima parametar option – vraća prikaz naziva partnera umjesto objekta za autocomplete

izracunaj() – vraća void – izračunava iznos stavke

izracunajIznosRacuna() – vraća void – izračunava iznos računa

ngOnInit() – vraća void

obrisiStavku() – vraća void, prima parametar stavka – briše odabranu stavku iz liste stavki

odaberiPartnera() – vraća void, prima parametar partner – postavlja trenutnog partnera na odabranog partnera

odaberiVrstu() – vraća void, prima parametar vrsta – postavlja trenutnu vrstu na odabranu vrstu računa

prikaziObavijest() – vraća void – generira obavijest o uspješnom spremanju novog računa

spremiPromjene() – vraća any – šalje podatke o novom računu i stavkama na API za spremanje u bazu podataka

spremiRacun() – vraća any, prima parametre datum, napomena, korisnik, partner, vrstaRacuna – sprema novi račun

spremiStavke() - vraća void, prima parametar racun – sprema stavke za kreirani račun

ucitajPodatke() – vraća void, prima parametar event – učitava podatke s API-ja za autocomplete

Konstruktor:

Parametri: route (tip podatka: ActivatedRoute), http (tip podatka: HttpClient), _router (tip podatka: Router), elementRef (tip podatka: ElementRef)

xxi. RacuniComponent

Komponenta za prikaz svih računa. Definira metode za dohvaćanje podataka o računima s API-ja te metode za pretraživanje i sortiranje podataka.

selector	racuni
templateUrl	./racuni.component.html

Atributi:

partneri (tip podatka: []), Filter (tip podatka: string), key (tip podatka: string), p (tip podatka: number), reverse (tip podatka: boolean), iznos (tip podatka: number), MatDialog (tip podatka: MatDialog), odabraniRacun (tip podatka: string), stavke (tip podatka: [])

Metode:

dohvatiStavke() – vraća void, prima parametar id_racuna – dohvaća stavke odabranog računa

openModal() – vraća void, prima parametar racunId – prikazuje print verziju odabranog računa

ngOnInit() – vraća void – dohvaća podatke o svim partnerima s API-ja za vrijeme inicijalizacije komponente

preusmjeriNovi() – vraća void – preusmjerava na kreiranje novog računa

sort() – vraća void, prima parametar key – sortira podatke prema dobivenom ključu.

Konstruktor:

Parametri: http (tip podatka: HttpClient), _router (tip podatka: Router)

xxii. VrstaRacunaEditComponent

Komponenta za uređivanje podataka o odabranoj vrsti računa. Definira metode za čitanje podatka o vrsti računa i zapisivanje novih podataka.

selector	vrsta-racuna-edit
templateUrl	./vrsta-racuna-edit.component.html

Atributi:

vrsta (tip podatka: any) – pohranjuje podatke o vrsti računa koja se uređuje (dobiveni s API-ja),

Atributi za spremanje novih podataka o vrsti računa: naziv (tip podatka: string), idVrste (tip podatka: number), id (tip podatka: any), opis (tip podatka: string)

Metode:

ngOnInit() – vraća void – dohvaća podatke s API-ja o vrsti računa koja se uređuje prilikom inicijalizacije komponente

prikaziObavijest() – vraća void – generira obavijest o uspješnom spremanju promjena

spremiPromjene() – vraća void – šalje promijenjene podatke o partneru na API za spremanje u bazu podataka.

Konstruktor:

Parametri: route (tip podatka: ActivatedRoute), http (tip podatka: HttpClient).

xxiii. VrstaRacunaNoviComponent

Komponenta za dodavanje nove vrste računa. Definira metode za spremanje podataka o novoj vrsti računa.

selector	vrsta-racuna-novi
templateUrl	./vrsta-racuna-novi.component.html

Atributi:

naziv (tip podatka: string), opis (tip podatka: string), idVrste (tip podatka: number)

Metode:

ngOnInit() – vraća void

preusmjeri() – vraća void – preusmjerava na prikaz svih vrsta računa nakon uspješnog spremanja

prikaziObavijest() – vraća void – generira obavijest o uspješnom spremanju nove vrste računa

spremiPromjene() – vraća void – šalje podatke o novoj vrsti računa na API za spremanje u bazu podataka

Konstruktor:

Parametri: route (tip podatka: ActivatedRoute), http (tip podatka: HttpClient), _router (tip podatka: Router)

xxiv. VrsteRacunaComponent

Komponenta za prikaz svih vrsta računa. Definira metode za dohvaćanje podataka o vrstama računa s API-ja te metode za pretraživanje i sortiranje podataka.

selector	vrste-racuna
templateUrl	./vrste-racuna.component.html

Atributi:

vrste (tip podataka: []), Filter (tip podataka: string), key (tip podataka: string), p (tip podataka: number), reverse (tip podataka: boolean),

Metode:

ngOnInit() – vraća void – dohvaća podatke o svim vrstama računa s API-ja za vrijeme inicijalizacije komponente

preusmjeriNovi() – vraća void – preusmjerava na kreiranje nove vrste računa

sort() – vraća void, prima parametar key – sortira podatke prema dobivenom ključu.

Konstruktor:

Parametri: http (tip podatka: HttpClient), _router (tip podatka: Router)

xxv. FooterComponent

Komponenta za generiranje podnožja.

selector	app-footer
styleUrls	./footer.component.css
templateUrl	./footer.component.html

Metode:

ngOnInit() – vraća void.

xxvi. NavbarComponent

Generira gornju navigacijsku traku.

selector	app-navbar
styleUrls	./navbar.component.css
templateUrl	./navbar.component.html

Atributi:

listTitles (tip podatka: any[]), location (tip podatka: Location), mobile_menu_visible (tip podatka: any), sidebarVisible (tip podatka: boolean), toggleButton (tip podatka: any),

Metode:

getTitle() – vraća any – dohvaća naslov stranice

ngOnInit() – vraća void

sidebarClose() – vraća void – zatvara sidebar

sidebarOpen() – vraća void

sidebarToggle() – vraća void

xxvii. SidebarComponent

Generira glavni izbornik.

selector	app-sidebar
styleUrls	./sidebar.component.css

templateUrl

./sidebar.component.html

Atributi:

menuItems (tip podatka: any[]), user (tip podatka: string)

Metode:

isMobileMenu() – vraća boolean

ngOnInit() – vraća void

4. Klase

U sklopu sustava definirane su 4 klase podataka:

1. Artikl
2. Partner
3. Stavka
4. VrstaRacuna

i. Artikl

Klasa koja definira tip podataka Artikl. Sastoji se od 6 atributa.

Atributi:

- Id (tip podatka: number)
- Broj (tip podatka: number)
- Naziv (tip podatka: string)
- JedinicaMjere (tip podatka: string)
- Porez (tip podatka: number)
- Cijena (tip podatka: number)

Konstruktor:

Parametri: id (tip podatka: number), broj (tip podatka: number), naziv (tip podatka: string), jedmjere (tip podatka: string), porez (tip podatka: number), cijena (tip podatka: number)

ii. Partner

Klasa koja definira tip podatka Partner. Sastoji se od 5 atributa.

Atributi:

- Id (tip podatka: number)
- Broj (tip podatka: number)
- Naziv (tip podatka: string)
- Adresa (tip podatka: string)
- Mjesto (tip podatka: string)

Konstruktor:

Parametri: id (tip podatka: number), broj (tip podatka: number), naziv (tip podatka: string), adresa (tip podatka: string), mjesto (tip podatka: string)

iii. Stavka

Klasa koja definira tip podatka Stavka. Sastoji se od 7 atributa.

Atributi:

- Artikl (tip podatka: Artikl)
- Kolicina (tip podatka: number)
- Cijena (tip podatka: number)
- Osnovica (tip podatka: number)

- Ukupno (tip podatka: number)
- IznosPDVa (tip podatka: number)
- PDV (tip podatka: number)

iv. VrstaRacuna

Klasa koja definira tip podatka VrstaRacuna. Sastoji se od 3 atributa.

Atributi:

- Id (tip podatka: number)
- Naziv (tip podatka: string)
- Opis (tip podatka: string)

Konstruktor:

Parametri: id (tip podatka: number), naziv (tip podatka: string), opis (tip podatka: string)

5. Servisi

Sustav definira 2 servisa, oba za autentifikaciju korisnika. Servisi su:

1. auth-guard
2. authentication

i. Auth-guard

Servis za provjeru postojanja prijavljenog korisnika prilikom pristupa nekoj od funkcionalnosti.

Metode:

canActivate() – vraća boolean, prima parametre route i state – provjerava je li korisnik prijavljen, ukoliko jest može aktivirati komponentu kojoj želi pristupiti, inače je preusmjeren na stranicu za prijavu.

ii. Authentication

Servis za provjeru prijave korisnika. Definira funkcije za provjeru korisničkog imena i lozinke te u slučaju uspjeha stvara sesiju za korisnika.

Atributi:

Korisnik (tip podataka: []) – za pohranu odgovora API-ja o postojanju korisnika s danim korisničkim imenom i lozinkom

Metode:

getData() – vraća void, prima parametre username i password – šalje podatke na API za provjeru korisnika u bazi podataka

authenticate() – vraća boolean, prima parametre username i password – provjerava dobiveni odgovor od API-ja o postojanju korisnika. U slučaju uspjeha stvara sesiju.

isUserLoggedIn() – vraća boolean – provjerava ima li prijavljenog korisnika

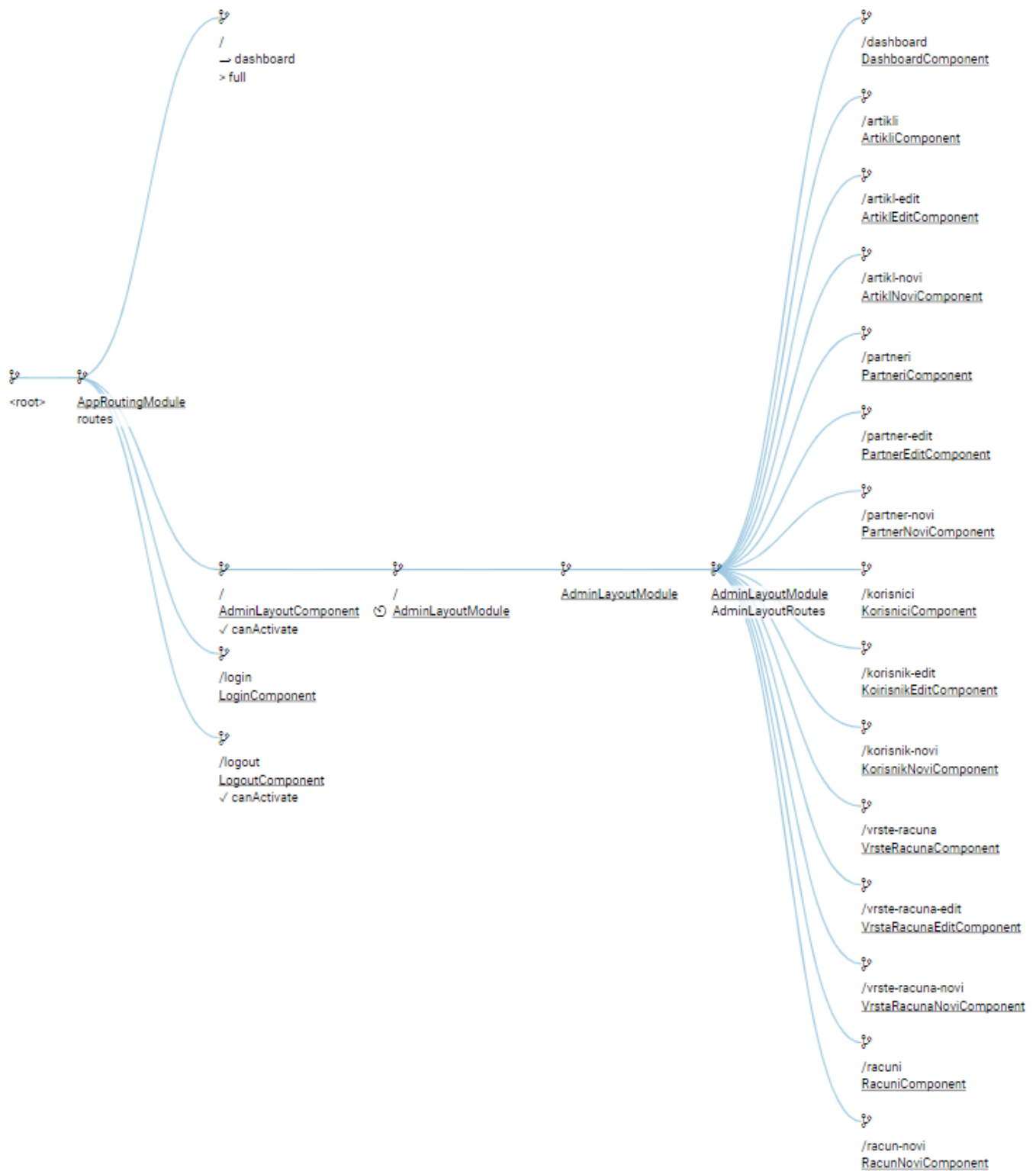
logout() – vraća void – odjavljuje korisnika, odnosno briše ga iz sesije

prikaziObavijest() – vraća void – generira obavijest kada prijava nije uspješna.

6. Routing i navigacijski dijagram

Datoteka admin-layout.routing.ts sadrži putanje do funkcionalnosti u sustavu, dakle sve rute do komponenti.

Routing dijagram:



Navigacijski dijagram:

