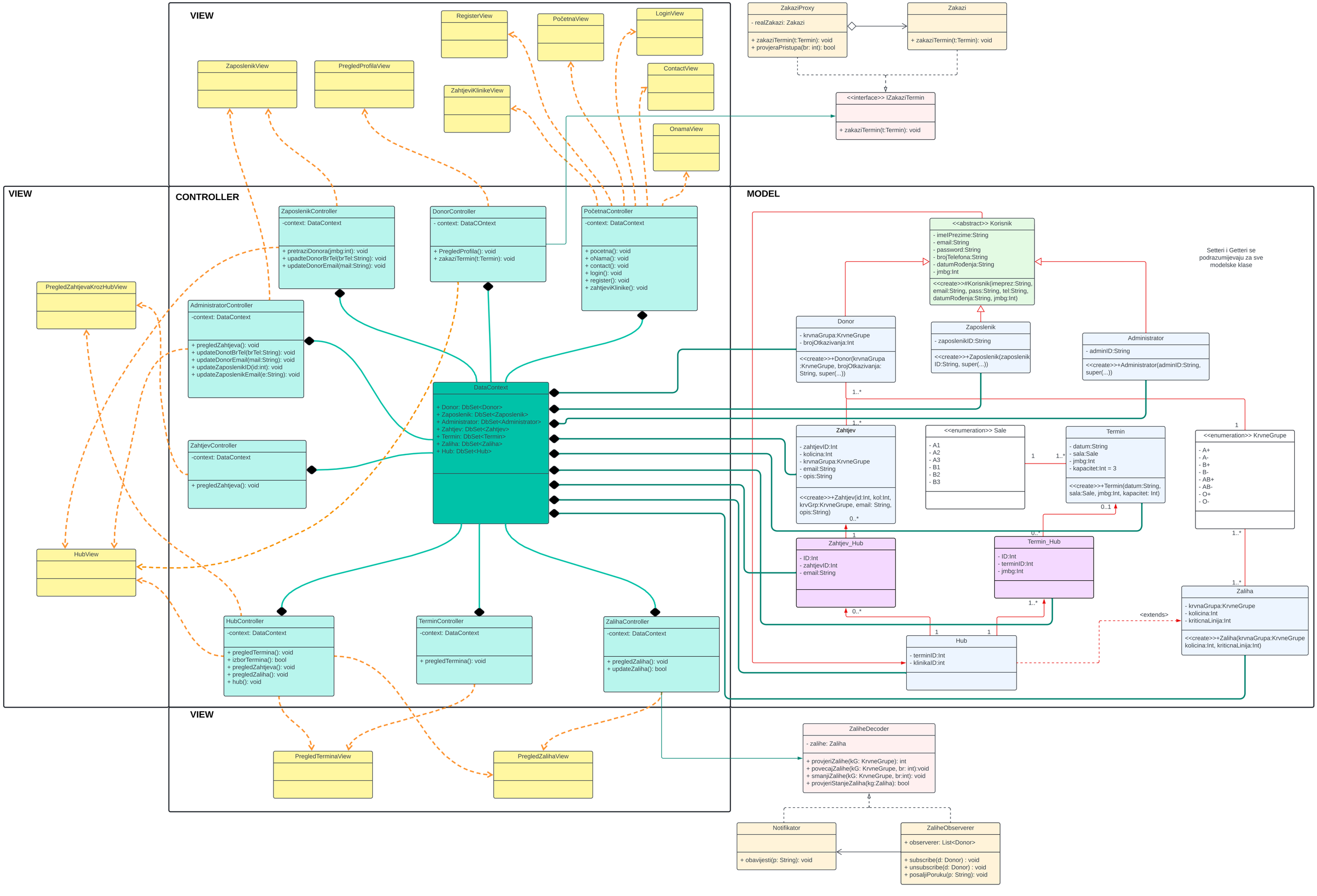
VIEW
RegisterView
PočetnaView
LoginView
ZaposlenikView
PregledProfilaView
ZahtjeviKlinikeView
ContactView
OnamaView
ZakaziProxy
- realZakazi: Zakazi
+ zakaziTermin(t:Termin): void
+ provjeraPristupa(br: int): bool
Zakazi
+ zakaziTermin(t:Termin): void
<<interface>> IZakaziTermin
+ zakaziTermin(t:Termin): void
VIEW
CONTROLLER
MODEL
PregledZahtjevaKrozHubView
ZaposlenikController
-context: DataContext
+ pretraziDonora(jmbg:int): void
+ upadteDonorBrTel(brTel:String): void
+ updateDonorEmail(mail:String): void
DonorController
- context: DataCOntext
+ PregledProfila(): void
+ zakaziTermin(t:Termin): void
PočetnaController
-context: DataContext
+ pocetna(): void
+ oNama(): void
+ contact(): void
+ login(): void
+ register(): void
+ zahtjeviKlinike(): void
AdministratorController
-context: DataContext
+ pregledZahtjeva(): void
+ updateDonotBrTel(brTel:String): void
+ updateDonorEmail(mail:String): void
+ updateZaposlenikID(id:int): void
+ updateZaposlenikEmail(e:String): void
DataContext
+ Donor: DbSet<Donor>
+ Zaposlenik: DbSet<Zaposlenik>
+ Administrator: DbSet<Administrator>
+ Zahtjev: DbSet<Zahtjev>
+ Termin: DbSet<Termin>
+ Zaliha: DbSet<Zaliha>
+ Hub: DbSet<Hub>
ZahtjevKontroller
-context: DataContext
+ pregledZahtjeva(): void
HubView
HubController
-context: DataContext
+ pregledTermina(): void
+ izborTermina(): bool
+ pregledZahtjeva(): void
+ pregledZaliha(): void
+ hub(): void
TerminController
-context: DataContext
+ pregledTermina(): void
ZalihaController
-context: DataContext
+ pregledZaliha(): void
+ updateZaliha(): bool
<<abstract>> Korisnik
- imeIPrezime:String
- email:String
- password:String
- brojTelefona:String
- datumRođenja:String
- jmbg:Int
<<create>>#Korisnik(imeprez:String, email:String, pass:String, tel:String, datumRođenja:String, jmbg:Int)
Setteri i Getteri se podrazumijevaju za sve modelske klase
Donor
- krvnaGrupa:KrvneGrupe
- brojOtkazivanja:Int
<<create>>+Donor(krvnaGrupa :KrvneGrupe, brojOtkazivanja: String, super(...))
Zaposlenik
- zaposlenikID:String
<<create>>+Zaposlenik(zaposlenik ID:String, super(...))
Administrator
- adminID:String
<<create>>+Administrator(adminID:String, super(...))
1..*
1..*
Zahtjev
- zahtjevID:Int
- kolicina:Int
- krvnaGrupa:KrvneGrupe
- email:String
- opis:String
<<create>>+Zahtjev(id:Int, kol:Int, krvGrp:KrvneGrupe, email: String, opis:String)
<<enumeration>> Sale
- A1
- A2
- A3
- B1
- B2
- B3
1
1..*
Termin
- datum:String
- sala:Sala
- jmbg:Int
- kapacitet:Int = 3
<<create>>+Termin(datum:String, sala:Sale, jmbg:Int, kapacitet: Int)
0..1
1
<<enumeration>> KrvneGrupe
- A+
- A-
- B+
- B-
- AB+
- AB-
- O+
- O-
1..*
0..*
1
Zahtjev_Hub
- ID:Int
- zahtjevID:Int
- email:String
0..*
Termin_Hub
- ID:Int
- terminID:Int
- jmbg:Int
1..*
1..*
Zaliha
- krvnaGrupa:KrvneGrupe
- kolicina:Int
- kriticnaLinija:Int
<<create>>+Zaliha(krvnaGrupa:KrvneGrupe kolicina:Int, kriticnaLinija:Int)
<extends>
1
1
Hub
- terminID:Int
- klinikaID:int
VIEW
PregledTerminaView
PregledZalihaView
ZaliheDecoder
- zalihe: Zaliha
+ provjeriZalihe(kG: KrvneGrupe): int
+ povecajZalihe(kG: KrvneGrupe, br: int):void
+ smanjiZalihe(kG: KrvneGrupe, br:int): void
+ provjeriStanjeZaliha(kg:Zaliha): bool
Notifikator
+ obavijesti(p: String): void
ZaliheObserverer
+ observerer: List<Donor>
+ subscribe(d: Donor) : void
+ unsubscribe(d: Donor) : void
+ posaljiPoruku(p: String): void