Sistem za podršku donošenja odluka pri dijagnosticiranju pacijenata

Praktikum - Napredne web tehnologije

Opis teme

Studenti: Golić Merisa (1191/16611)

Nogo Armin (1192/16617)

Palavra Adi (1129/16615)

Peljto Mirnes (1130/16602)

Sarajevo, Mart 2017.g.

1. Opis teme

Korisnicima će biti omogućen prikaz najvjerovatnijih dijagnoza na osnovu unesenih simptoma, te će doktorima biti omogućeno bilježenje svih postavljenih dijagnoza za pacijente. Pored toga, logovanim korisnicima (doktorima) će biti prikazana preporučena doza na osnovu unosa dodatnih parametara koji se tiču pacijenta. Jedan korisnik (administrator) će upravljati svim korisnicima sistema (doktorima).

1. Moduli

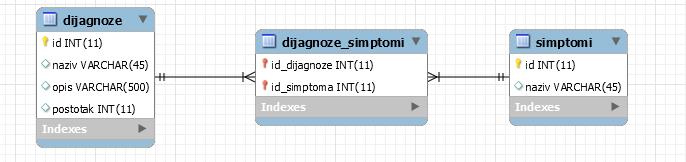
Svaki modul predstavlja dio mikroservisa.

1. Modul za dijagnosticiranje

Na osnovu unesenih simptoma vrši se uspostavljanje dijagnoze, te prikaz najvjerovatnijih dijagnoza sa procentom poklapanja simptoma.

Nakon što korisnik unese (odabere) simptome za koje želi izvršiti dijagnosticiranje, dobice prikaz dijagnoza koje odgovaraju tim simptomima. Biće omogućen prikaz svih dijagnoza koje odgovaraju barem jednom od navedenih simptoma, ali će se moći postaviti i određeni prag poklapanja kako bi se prikaz suzio samo na one dijagnoze čije je procentualno poklapanje stvarnih simptoma sa unesenim simptomima veće od odabranog praga. Prilikom odabira neke od ponuđenih dijagnoza, prikazaće se i kraći opis date dijagnoze (link na stranicu na kojoj se nalazi više informacija o datoj dijagnozi). Ovaj modul će biti osnova za drugi i treći modul. Potvrdom neke od prikazanih dijagnoza kao i unosom dodatnih informacija o pacijentu - ime, datum rođenja, spol, dodaće se informacije u bazu podataka.

ERD1:

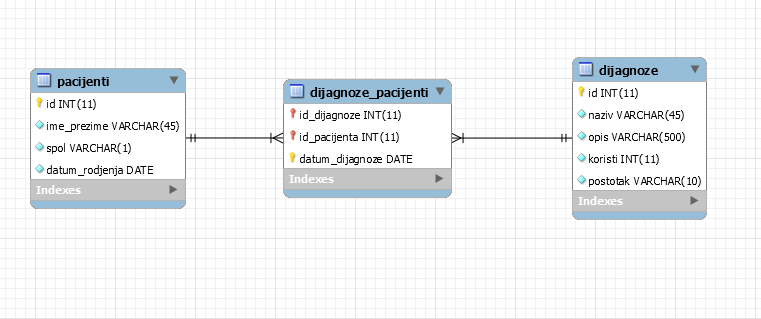


1. Modul za praćenje dijagnoza

Logovani korisnik (doktor) ima pristup spisku svih svojih pacijenata, kao i pregled historije dijagnoza za svakog pacijenta.

Nakon što se korisnik loguje, biće mu omogućen prikaz svih dijagnoza koje je on potvrdio svojim pacijentima, tj. biće mu prikazana lista svih pacijenata koje je on dijagnosticirao, te odabirom jednog od njih će se prikazati i sve dijagnoze tog pacijenta.

ERD2:

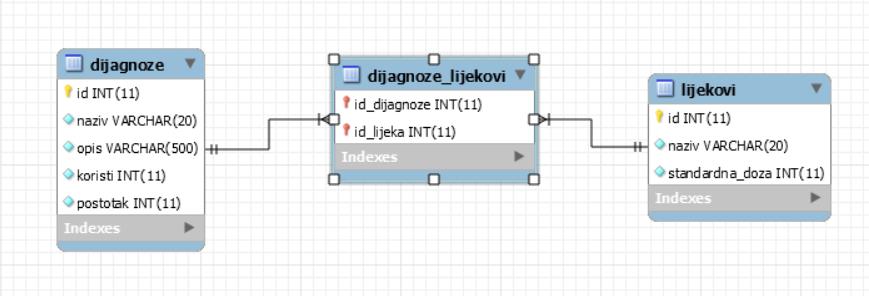


1. Modul za određivanje terapije

Za svaku dijagnozu pacijenta određuje se terapija (lijek i doza), na osnovu unesenih parametara dijagnoze.

Kada se potvrdi dijagnoza od strane logovanog korisnika (doktora) biće omogućeno određivanje doze lijeka koji je preporučen za datu dijagnozu. Potrebno je da korisnik unese određene parametre koji pružaju dodatne informacije o pacijentu (rezultate nekih nalaza i sl.), na osnovu kojih će se vršiti određivanje tačne doze lijeka za datog pacijenta.

ERD3:

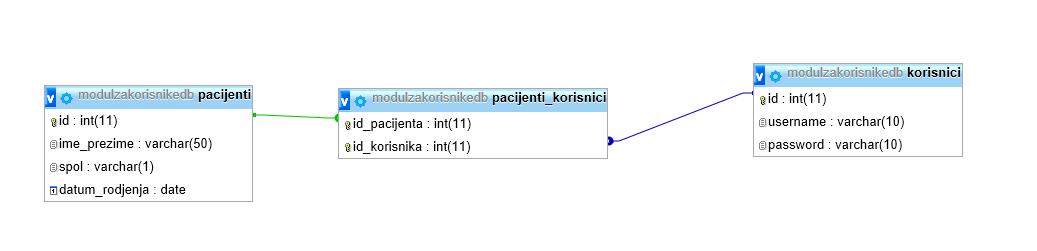


1. Modul za korisnike

Ovaj modul omogućava upravljanje korisnicima sistema, tj. dodavanje novih ili brisanje starih korisnika.

Jedan korisnik (koji predstavlja administratora sistema) će upravljati svim ostalim korisnicima sistema, tj. on će biti zadužen za dodavanje novih korisnika, eventualno brisanje korisnika koji više ne koriste sistem ili kojima su oduzete privilegije.

ERD4:



S obzirom da je tabela dijagnoze zajednička za više modula, ona je duplicirana u modulima za dijagnosticiranje, praćenje dijagnoza i određivanje terapije. Da bi se osigurala konzistentnost podataka, prilikom dodavanja/izmjene/brisanja novih slogova u tabelu dijagnoze u modulu za dijagnosticiranje, vršiće se komunikacija sa ostalim modulima te izvršavati promjene na dupliciranim tabelama. Isto vrijedi i za tabelu pacijenti koja je zajednička za modul za praćenje dijagnoza i za modul za korisnike. Tabela je primarna u modulu za praćenje dijagnoza, dok će se komuniciranjem osigurati da ispravni podaci budu i u modulu za korisnike.

1. Sigurnost – autentifikacija i autorizacija

JSON Web token

JSON web token je standard (RFC 7591) koji definiše kompaktan i nezavisan način za siguran prenos informacija između subjekata, koristeći JSON objekte. Ove informacije mogu biti verifikovane i može im se vjerovati jer su digitalno potpisane. JWT-i su male veličine i mogu se prenostiti putem URL-a, POST zahtjeva ili u zaglavlju HTTP zahtjeva, što znači da se štedi i na vremenu prenosa.

Prednosti JWT autentifikacije i autorizacije

Autentifikacija: Jednom kada se korisnik prijavi na sistem, svaki naredni zahtjev sadrži JWT, što omogućava korisniku da pristupi rutama, servisima i resursima koji su dozvoljeni koristeći taj token.

Razmjena informacija: JSON Web tokeni su dobar način prenosa sigurnog prenosa informacija između subjekata, zato što mogu biti digitalno potpisani, koristeći RSA enkripciju parom javnih ili privatnih ključeva. S obzirom da se potpis dobiva na osnovu zaglavlja i podataka, može se provjeriti da li se sadržaj poruke mijenjao.

Struktura JSON Web tokena

JSON web token sadrži tri dijela radvojena tačkom:

* Zaglavlje
* Podaci
* Potpis

Primjer izgleda JWT-a : eyJhbGciOiJIUzUxMiJ9.eyJzdWIiOiJtZ29saWMiLCJleHAiOjE0OTUxMTU0OTB9.DkQeWrxaKwxbXcLFDRtKJAXyPuPLYFWUD6KstpyUpNqdsbEbvnB1BarZ2KhLQBWr1lNm4NQaVz-H79qEk3MLg

Način funkcionisanja JWT-a

Pri autentifikaciji, kada se korisnik prijavi sa svojim podacima, JSON Web token se šalje korisniku koji bi ga trebao spasiti lokalno. Kad god korisnik želi pristupiti zaštićenoj ruti ili resursu, user agent šalje spremljeni JWT u zaglavlju Autorizacija koristeći Bearer šemu. Server će za zaštićene rute provjeravati da li u zaglavlju postoji ispravan JWT, te na osnovu toga odobravati pristup.

Poređenje JWT-a sa drugim tehnikama

U poređenju sa drugim sličnim tehnikama zaštite, kao što su Simple Web Token ili Security Assertion Markup Language Tokens, JSON Web token je manje opširan jer ne zahtjeva XML, što mu omogućava lakše prenošenje putem HTTP zahtjeva, te upotrebljava RSA algoritam enkripcije koji omogućava digitalni potpis. Također, JSON objekte je puno jednostavnije digitalno potpisati, s obzirom na njihovu jednostavnost, za razliku od XML dokumenata, što ovu tehniku čini još poželjnijom. JSON se sve više koristi u mnogim programskim jezicima zbog jednostavnog direktnog mapiranja na objekate, dok XML nema unaprijed određenu mogućnost mapiranja dokumenta na objekte.

Poznatiji protokoli autentifikacije i autorizacije kao što su kerberos, cas ili Oauth, koriste tokene prema referenci, što znači da se prilikom prijave korisnika na sistem, u modulu za autentifikaciju dodjeljuje tom korisniku token, a zatim se prilikom svakog pristupa određenom resursu vrši komunikacija sa modulom za autentifikaciju te vrši provjera ispravnosti tog tokena. Ovakav pristup pogodan je za monolitne aplikacije, dok bi bio poguban za mikrosrvisne aplikacije, s obzirom da povećava komunikaciju između modula, kompleksnost koda i infrastrukture, te ne skalira dobro povećanjem broja modula. Zbog toga je JWT u potpunosti prilagođen mikroservisnoj arhitekturi.

JWT koristi tokene prema vrijednosti, što znači da se pamti samo potrebne informacije za autentifikaciju i autorizaciju, te se mogu provjeriti odmah u modulu od kojeg se zatijevaju podaci, samim tim smanjujući komunikaciju prema autentifikacijskom servisu, te omogućavajući njegov jednostavniji dizajn