

## Projektni zadatak

- maj 2020 -

Za potrebe države potrebno je napraviti jednostavan registar osoba koje su bile u potencijalnom kontaktu sa osobom koja je nosilac virusa. Sve osobe moraju da budu povezane sa centralnim registrom. Zbog toga je potrebno napraviti nekoliko aplikacija koje će se ovom prilikom koristiti.

Svaka osoba na svom računaru treba da koristi *desktop* aplikaciju. Prilikom pokretanja aplikacije prvo se otvara dijalog (*popup*) u kojem korisnik treba da se prijavi na token server. Prilikom prve prijave na token server (ili nakon odjave i ponovne prijave) korisnik unosi ime, prezime i jedinstveni matični broj koji se šalje na token server. Nakon prijave na token server i ako su podaci validni, korisnik dobija jedinstveni token koji se čuva u sklopu njegove aplikacije i na token serveru. Nakon dobijanja tokena, otvara se prozor u kojem korisnik unosi svoju lozinku koja se čuva u sklopu njegove aplikacije i služi za lokalnu prijavu u aplikaciju. Svaki naredni put prilikom prijave i prilikom slanja podataka u centralni registar zahtijeva se token od korisnika te se verifikuje putem token servera. Nakon prijave, u glavom prozoru aplikacije nalazi se meni sa opcijama za zatvaranje aplikacije (lokalne odjave sa sistema), odjavu iz registra, pregled korišćenja aplikacije i promjenu lozinke i prikaz mape evidentiranih pozicija. U centralnom dijelu aplikacije nalaze se opcije za komunikaciju sa medicinskim osobljem, za slanje podataka o trenutnoj lokaciji i za slanje medicinske dokumentacije. Sa lijeve strane centralnog dijela aplikacije nalazi se prostor za razmjenu poruka sa medicinskim osobljem. U centralnom dijelu centralnog prostora nalazi se forma u koju korisnik unosi koordinate trenutne lokacije (u tekstualna polja ili odabirom na mapi) na kojoj se nalazi i tačan period kada je bio na toj lokaciji. Na desnoj strani centralnog dijela nalazi se forma za slanje medicinskih fajlova medicinskom osoblju (do 5 dokumenata). Ispod menija, a iznad ostalih dijelova aplikacije, korisnici imaju liniju za dobijanje automatskog obavještenja o tome da su potencijalno bili izloženi virusu i dugme za pristup formi sa detaljima na kojoj se prikazuje fotografija mape sa koordinatama na kojoj se desio potencijalni kontakt. Prilikom pregleda korišćenja aplikacije, korisniku se u novom prozoru prikazuje tabela sa podacima o prijavama na sistem (vrijeme prijave, vrijeme odjave i ukupno aktivno vrijeme za tu sesiju kao razlika ova dva vremena). Pored toga korisnik ima mogućnost prikaza svih pozicija koje je evidentirao na mapi u poslednjih  $n$  dana (pristup iz menija).

Centralni registar ima svoju *desktop* aplikaciju koju koristi medicinsko osoblje. Ova aplikacija ima listu svih registrovanih osoba koje se identifikuju preko jedinstvenog tokena putem token servera (nema ličnih podataka), pregled lokacija osobe (ista mapa kao u aplikaciji za osobe) koji se aktivira izborom jedne osobe. Osobe se prije toga pretražuju po tokenu (*LIKE* pretraga). *Chat* komunikacija se može vršiti sa osobom iz medicinskog osoblja koja je trenutno slobodna. Pri tome server dodjeljuje narednu slobodnu osobu iz medicinskog osoblja po redu. Medicinsko osoblje nakon završetka komunikacije ima mogućnost prekida *chat*-a sa trenutnom osobom i samim tim se vraća u red raspoloživih osoba za komunikaciju. Medicinsko osoblje takođe ima mogućnost da označi token određene osobe tj. osobu da: nije zaražena, da je potencijalno zaražena ili da je zaražena. Kada se se detektuje/prijavi potencijalno zaražena ili zaražena osoba, osobi se automatski šalje i fotografija mape sa lokacijom na kojoj je ostvaren potencijalni kontakt. Pored toga, aplikacija ima i polje u kojem se izlistavaju potencijalni slučajevi kontakta, a koje server automatski osvježava svakih  $n$  sekundi. Pri tome se evidentiraju oni slučajevi osoba koje su imale potencijalno međusobno rastojanje manje od  $k$  metara u vremenskom intervalu od  $p$  minuta i pri čemu je jedna od osoba potencijalno zaražena ili zaražena (vrijednosti  $n$ ,  $p$  i  $k$

su konfigurabilne i konfiguriraju se na strani centralnog registra). Aplikacija sve osobe tj. tokene koji su ostvarili potencijalni kontakt označava kao potencijalno zaražene, te im šalje automatsku poruku o istom. Medicinsko osoblje ima i mogućnost blokade aktivnih naloga osoba nakon čega osobe više ne mogu vršiti interakciju sa centralnim registrom.

Opcije za prijavu na sistem (dobijanje tokena), izlistavanje svih aktivnih tokena, provjeru da li određeni token postoji te deaktivaciju tokena dostupne su preko SOAP servisa. Pri tome se evidencija čuva na posebnom serveru namijenjenom samo za tokene. Svi podaci se čuvaju u *in-memory* bazi podataka (npr. Redis). Osim toga, u *in-memory* bazi podataka, a u sklopu centralnog registra čuvaju se i ažuriraju podaci o evidentiranim lokacijama osoba i tipu osoba i dostupni su preko REST servisa. Chat funkcionalnost obavlja se preko *Socket* servera na siguran način. U okviru istog servera potrebno je implementirati opcije za pojedinačnu razmjenu tekst poruka između osobe i medicinske osobe koja je slobodna i za grupnu razmjenu poruka između medicinskog osoblja.

Čuvanje, prikaz informacija i preuzimanje dokumenata potrebno je implementirati kao RMI aplikaciju. Dokumenti svih korisnika čuvaju se na posebnom fajl serveru. Potrebno je definisati strukturu i način čuvanja fajlova kako bi sistem bio efikasan. Voditi računa o količini prenesenih podataka u procesu slanja dokumenata i po potrebi iskoristiti mehanizme komprimovanja istih. Slanje poruka između medicinskog osoblja potrebno je implementirati kao *multicast*. Svo medicinsko osoblje u tom slučaju dobija i prikazuje poruke. Svako dobijeno obavještenje se serijalizuje na fajl sistemu osobe. Potrebno je implementirati najmanje 4 različita načina serijalizacije, pri čemu se svaka poruka serijalizuje na drugi način (1. poruka – 1. način, ... 4. poruka – 4. način, 5. poruka – 1. način...).

Mapu je moguće implementirati i u obliku matrice dimenzija  $n \times m$ . Sve parametre koji se koriste u GUI aplikacijama i drugim programima potrebno je definisati u odgovarajućim *properties* fajlovima. Za obradu izuzetaka potrebno je koristiti *Logger* klasu. Dozvoljeno je napraviti skriptu ili program koji popunjava početni skup podataka kako bi se olakšalo testiranje sistema. Studenti trebaju implementirati mehanizam za generisanje tokena. Dovoljno je da to bude jedinstven UUID za svakou osobu.

Studenti dio projekta mogu da realizuju i u drugom programskom jeziku. U tom slučaju prilikom predaje projektnog zadatka potrebno je dostaviti dokument u kojem će biti navedeno koji su dijelovi projektnog zadatka rađeni u drugom programskom jeziku, uputstvo za pokretanje i listu dodatnih softvera sa verzijama (ukoliko su isti potrebni za pokretanje).

