

Zadatak

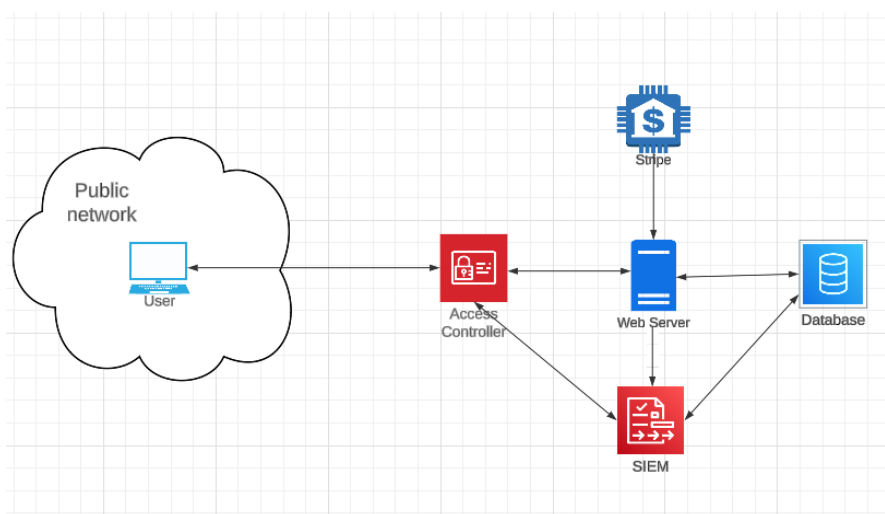
Realizovati informacijski sistem osiguravajuće kuće, koji se sastoji od dvije aplikacije: *administratorske* i *klijentske*. Sistem omogućava klijentima pregled, kupovinu i upravljanje polisama osiguranja (životno, putno, osiguranje imovine, itd.). Administratorska aplikacija omogućava upravljanje korisničkim nalogima i uslugama koje se nude klijentima.

Novi klijent se registruje na sistem, a, nakon registracije, može se prijaviti na sistem, pri čemu se koristi dvo-faktorska autentikacija. U prvom koraku klijent unosi korisničko ime i lozinku, nakon čega dobija verifikacioni kod putem *e-mail*-a, koji je potrebno unijeti u narednom koraku prijave.

Potrebno je implementirati Single Sign On (SSO) mehanizam, tako da korisnici (administratori, zaposleni), koji imaju pristup objema aplikacijama, mogu da se autentikuju u bilo kojoj aplikaciji i da imaju pristup čitavom sistemu.

Klijenti mogu da kupe određenu polisu, pri čemu je za potrebe simulacije plaćanja neophodno izvršiti integraciju nekog od postojećih sistema za plaćanje (*Stripe*, *PayPal sandbox* i sl.). Plaćanje se vrši isključivo u testnom režimu, bez generisanja stvarnih naplata, a transakcijske podatke je potrebno sačuvati u bazi podataka na odgovarajući način. Prilikom uspješne kupovine, klijent će dobiti kupljenu polisu u PDF formatu, putem email-a.

Na slici 1 je prikazan jedan primjer arhitekture sistema. *Access Controller* je komponenta preko koje korisnici vrše interakciju sa sistemom (autentikacija i autorizacija korisnika, praćenje sesije, detekcija sigurnosno osjetljivih zahtjeva itd). Ukoliko detektuje potencijalno maliciozan zahtjev (na primjer, unos znatno većeg iznosa za uplatu, itd), *Access Controller* može automatski da zatvori sesiju sa korisnikom. SIEM komponenta je zadužena za praćenje i logovanje svih sigurnosno osjetljivih zahtjeva. Potrebno je konfigurisati *Nagios* za monitoring svih ključnih servisa.



Slika 1 – Arhitektura sistema

Sve detalje zadatka koji nisu precizno specificirani realizovati na proizvoljan način. Detalje korisničkog interfejsa realizovati na proizvoljan način. Sistem treba da bude *web* baziran. Dozvoljena je upotreba proizvoljnog programskog jezika, kao i proizvoljnih tehnologija neophodnih za realizaciju tehničkih detalja.

Studenti koji uspješno odbrane projektni zadatak stiču pravo izlaska na usmeni dio ispita. Prije odbrane, potrebno je postaviti kompletan izvorni kod projektnog zadatka na *moodle*. Projektni zadatak važi od prvog termina januarsko-februarskog ispitnog roka 2025. godine i vrijedi do objavljivanja sljedećeg projektnog zadatka.