

# Projektni zadatak RCA

## Stack Overflow

Implementirati sistem za postavljanje pitanja i odgovora, inspirisan platformom StackOverflow. Sistem će omogućiti korisnicima da postavljaju tehnička pitanja, daju odgovore, glasaju za najbolje odgovore i komuniciraju u okviru zajednice. Sistem će biti podeljen na više servisa, uključujući zasebne servise za slanje notifikacija i proveru dostupnosti sistema.

### StackOverflowService – Neophodne funkcionalnosti (Web Role)

Korisnički interfejs je *Web Role* aplikacija koja treba da opsluži korisnika u interakciji sa platformom.

Akcije koje treba da budu podržane na korisničkom interfejsu:

- Registracija novog korisnika
- Prijava novog korisnika
- Izmena korisničkog profila
- Postavljanje pitanje
- Izmena i brisanje postavljenog pitanja (samo korisnik koji je postavio pitanje)
- Odgovor na postavljeno pitanje
- Glasanje za odgovor
- Pretraga na osnovu naslova
- Sortiranje na osnovu broja glasova i datumu postavljanja pitanja

Korisnik se registruje unoseći:

- Ime i prezime
- Pol
- Država, grad i adresa
- Email
- Lozinka
- Slika korisničkog naloga

Korisnik se prijavljuje putem:

- Email
- Lozinka

Postavljanje pitanja:

- Naslov pitanja
- Detaljan opis problema
- Slika greške/problema

### Opšti opis korisničkog iskustva

Korisnik se nakon registracije ili prijave na sistem usmerava na stranicu gde su izlistana sva pitanja koja su postavljena. Korisnik može da postavi novo pitanje, izmeni ili obriše pitanje koje je postavio. Korisnici mogu da glasaju za neki odgovor i da odgovore na postavljeno pitanje. Korisniku se u okviru postavljenog pitanja prikazuje ukupan broj odgovora koje je dobio. Autor pitanja može da izabere jedan od odgovora i označi ga kao najbolji. Svima koji su

odgovorili na pitanje stiže mejl sa obaveštenjem da je pokrenuta tema zatvorena i koji je konačan odgovor na postavljeno pitanje.

### StackOverflowService (Web Role)

Servisni sloj zadužen za opsluživanje svih korisničkih zahteva.

### NotificationService (Worker Role)

Slanje mejlova korisnicima ne sme da poremeti rad sistema. Po mogućnošću koristiti neki servis za slanje mejlova ili slati ručno (SMTP). Primer sistema za slanje mejlova jesu *Postmark* ili *SendGrid*.

Slanje mejla je neophodno implementirati kroz sledeće elemente:

- Zaseban *Worker Role* servis pod nazivom - **NotificationService** - 3 instance
- Prilikom izbora najboljeg odgovora na pitanje - **StackOverflowService** rola šalje poruku u red (naziv reda npr. *notifications*) gde ubacuje ID odgovora
- Background prilikom čitanja iz reda dobija ID odgovora na osnovu koga pronalazi pitanje na koje se odgovor odnosi. Na osnovu pitanja pronalaze se svi korisnici koji su odgovorili na to pitanje i šalje im se mejl da je pitanje zatvoreno. Telo mejla treba da sadrži informaciju da je pitanje uspešno zatvoreno, ko je autor finalnog odgovora i tekst odgovora.
- Persistovati informacije o svakoj poslatoj grupi notifikacija:  
| datum-vreme | id-odgovora | broj-poslatih-mejlova |

### HealthMonitoringService (Worker Role)

Implementirati servis za proveru dostupnosti sistema.

Zaseban *Worker Role* servis pod nazivom - **HealthMonitoringService** - 2 instance

- Na svake 4 sekunde šalje zahtev ka **StackOverflowService** i **NotificationService** /health-monitoring endpoint gde ukoliko sam zahtev prođe smatra se da je sve kako treba, a ukoliko ne prođe šalje se mejl na mejl adrese sačuvane u Azure tabeli (npr. *AlertEmails*)
- Svejedno da li je zahtev prošao ili nije, neophodno je upisati poruku u posebnu tabelu *HealthCheck* sa trenutnim datumom i vremenom, statusom da li je uspeo ili nije (OK ili NOT\_OK) i nazivom servisa:  
| datum-vreme | status | naziv-servisa |

### HealthStatusService (Web Role prikaz)

Potrebno je realizovati zasebnu *Web Role* aplikaciju (najjednostavnije u MVC-u) koja će vizuelno prikazivati dostupnost **StackOverflowService**.

Zahtevi:

- Čitanje podataka: Aplikacija treba da povlači informacije iz *HealthCheck* tabele za poslednja 3 sata
- Vizuelni prikaz: Grafički prikazati dostupnost/nedostupnost servisa (npr. pomoću grafikona ili tabela)
- Procentualna dostupnost – Izračunati i prikazati udeo vremena (u procentima) tokom koga je servis bio dostupan, odnosno nedostupan

### AdminToolsConsoleApp (ConsoleApp)

Konzolna aplikacija koja komunicira sa **HealthMonitoringService**.

Zahtevi:

- Omogućava izmenu mejlova na koje se šalju upozorenja ukoliko zahtev za proveru dostupnosti ne prođe (odnosi se na imejl adrese na koje **HealthMonitoringService** šalje upozorenja)
- Omogućava prikaz svih mejl adresa na koja se šalju upozorenja i brisanje istih

