

*Ispit traje 2.5 sata. Nije dozvoljena upotreba literature.*

1. (15) a) Objasniti osnovni algoritam (npr. u pseudokodu) koji reguliše trenutni broj instanci web i worker uloga cloud aplikacije. Komunikacija web i workera je asinhrona.  
b) Objasniti kako se iz PHP-a može regulisati broj instanci uloge u Azure oblaku.  
c) Pretpostavimo da cloud aplikacija ima dve instance web uloge (A1 i A2) i dve instance worker uloge (B1 i B2). Pretpostavimo da se klijent K1 ulogovao na cloud aplikaciju (što je preduslov da mu se prihvate zahtevi) i do je sada slao zahteve instanci A1. Ako “padne” instanca A1, a klijent K1 posle toga pošalje novi zahtev, da li će taj zahtev biti obrađen? Obrazložiti odgovor.  
d) U čemu je razlika između definicije i konfiguracije Azure servisa?
2. (15) a) Navesti različite jedinice mere koje se koriste pri proceni troškova.  
b) Definisati pojam kupe neizvesnosti procene.  
c) Na koji način se formula za optimalno vreme razvoja softvera može koristiti da se odredi potreban broj programera?  
d) Navesti nazive različitih tipova sistema prema COCOMO metodologiji. Za svaki tip dati primere aplikacija.
3. (30) Neka je data mini aplikacija „Parking servis Beograd“. Korisnici aplikacije su dežurni kontrolori, koji aplikaciji pristupaju preko tablet računara, i administratori. Za svakog korisnika pamti se korisničko ime (minimalno 8 karaktera), lozinka, ime i prezime, telefon i e-mail. Na početnoj strani korisnik treba da ima mogućnost unosa korisničkog imena i lozinke i u slučaju ispravno unetih podataka, treba omogućiti rad sa ostatkom sistema.

Ukoliko automobil nema nalepnicu za parkirališta, vozač automobila mora da putem SMS poruke kupi kartu za parking u trajanju od 1h (za zonu #1), od maksimalno 2h (za zonu #2) i od maksimalno 3h (za zonu #3). Kontrolor u sistemu može vršiti proveru parking karte, tako što će uneti određeni registarski broj automobila, nakon čega dobija listu poslednjih pet uplata za to vozilo, brojeve mobilnih telefona sa kojih su uplaćene karte i nazive zona u kojima je vozač automobila uplatio parking karte (1 - CRVENA; 2 - ŽUTA; 3 - ZELENA). Osim provere karte, kontrolor može da naplati i dopunsku kartu (kaznu) za automobil. Ako je od vremena poslednje karte proteklo više od 1h u crvenoj zoni (ili više od 2h u žutoj ili više od 3h u zelenoj zoni), u odnosu na trenutno vreme, kontroloru pored takve uplate tada treba ponuditi dodatnu opciju naplati dopunsku kartu. Kontrolor treba izborom te opcije da na posebnoj stranici dopuni još lokaciju (ulicu) na kojoj je naplaćena kazna i da potvrdi kaznu. Lokacija se dobija iz šifarnika koji sadrži spisak ulica i zonu kojoj ta ulica pripada (ulica može pripadati i dvema zonama).

Naplata dopunske parking karte, može se izvršiti i na posebnoj stranici, za one vozače koji uopšte nisu slali SMS poruke za plaćanje parkinga, unosom svih podataka o tom vozilu. Kada se kontrolor uloguje na sistem, osim opcija provera parking karte, treba mu

dozvoliti i takvu opciju - naplata dopunske karte, a takođe treba mu prikazati i zbir svih kazni koje je on napisao u tekućem mesecu.

Administrator ima iste opcije kao i kontrolor uz još neke dodatne opcije:

A1) Unos nove lokacije (ulice) i izmena zone kojoj pripada ulica.

A2) Unos nove zone i promena cene karte u određenoj zoni (cena karte za svaku zonu čuva se u posebnoj tabeli, zajedno sa cenom dopunske karte) ili dopunske karte. Dopunska karta ima istu vrednost za sve zone.

A3) Štampanje izveštaja koliko je svaki kontrolor proverio registracija i koliko je napisao kazni za određeni dan, sedmicu ili mesec (u vidu liste svih kontrolora ili listu sa dodatnim filtriranjem po imenu i prezimenu, od strane administratora).

A4) Štampanje izveštaja o prosečnom broju kazni u određenom opsegu (period od jednog do drugog datuma može slobodno uneti) po zonama, kao i o celokupnim prihodima parking servisa.

Pretpostaviti da je aplikacija napravljena tako da svaki nelegalni unos podataka ispisuje poruku o određenoj grešci.

Na osnovu datog opisa:

a) [6] Napraviti SSU *Provera parking karte* (uz mogućnost pisanja dopunske karte, od strane kontrolora), a zatim skicirati prototip cele aplikacije metodom *storyboard* (ilustracija scenarija).

b) [6] Nacrtati IE model baze podataka, sa svim entitetima i vezama, koje su potrebne za realizaciju ove aplikacije koju projektujete, prema opisanoj specifikaciji.

c) [9] Korišćenjem UML dijagrama sekvence, nacrtati sekvencu *Unos nove ulice* (uz provere da takva ulica i zona kojoj pripada, već ne postoje u bazi).

d) [9] Napisati PHP skript koji vrši ispisivanje pregled parking karti za unetu registraciju, ali tako da ispunjava uslove specifikacije zadatka.

Dozvoljeno je koristiti sve tabele koje ste definisali u modelu baze pod tačkom b).