

DOMAĆI ZADATAK
INFORMACIONI SISTEMI 1
(13S113IS1)

školska 2021/22. godina

Uvod

Za domaći zadatak potrebno je realizovati sistem za funkcionisanje jedne banke.

Opis podataka

U sistemu banke pamte se podaci o komitentima i to naziv i adresa, kao i mesto koje je njihovo sedište. Za mesto se pamti poštanski broj i naziv. Filijala se može otvoriti u određenom mestu i za nju se pamte naziv i adresa. Komitent u banci može otvoriti račun u određenom mestu kada je njegovo stanje 0. Račun poseduje dozvoljeni minus. Ukoliko stanje pređe ispod dozvoljenog minusa račun se automatski blokira. Kada stanje pređe iznad dozvoljenog minusa račun se aktivira. Pamte se još i datum i vreme otvaranja računa, broj transakcija. Transakcija može biti prenos sume sa računa na račun, uplata novca na račun, isplata novca sa računa. Za transakciju se još pamti i datum i vreme obavljanja, iznos, redni broj u okviru računa, svrha. Ukoliko je transakcija uplata ili isplata novca pamti se još i filijala u kojoj je to obavljeno.

Opis sistema

Sistem se sastoji iz korisnička aplikacije, centralnog servera i tri podsistema. Kroz korisničku aplikaciju je potrebno omogućiti unosa svih korisničkih zahteva. Korisnička aplikacija kreira REST zahteve i šalje ih centralnom serveru i vraća rezultat poslatih zahteva. Centralni server obrađuje REST zahteve i prosleđuje ih podsystemima koristeći JMS. Podsystemi komuniciraju isključivo preko JMS. Definicije pojedinačnih delova sistema su date u nastavku.

Klijentska aplikacija

Klijentska aplikacija je java program (Java SE) koji od korisnika dobija zahteve kroz konzolu ili putem grafičkog interfejsa (po Vašem izboru), kreira REST zahteve i šalje ih centralnom serveru. Klijent svaki odgovor od centralnog servera ispisuje na konzoli ili putem grafičkog interfejsa.

Podsystem 1

Podsystem 1 u svojoj bazi podataka čuva podatke o komitentima, mestima, filijalama i njihovim međusobnim vezama. Podsystem 1 komunicira isključivo preko JMS.

Podsystem 2



Podsystem 2 u svojoj bazi podataka čuva podatke o komitentima, računima, transakcijama i njihovim međusobnim vezama. Podsystem 2 komunicira isključivo preko JMS.

Podsystem 3

Podsystem 3 u svojoj bazi podataka čuvan rezervne kopije podataka iz baza podataka podsistema 1 i 2. Čuvanje funkcioniše tako što podsystem periodično preko JMS dohvata podatke iz podsistema 1 i 2 i čuva ih u svojoj bazi kao rezervnu kopiju. Periodo čuvanje rezervne kopije se poziva na svakih 2 minuta.

Centralni server

Centralni server ne čuva nikakve podatke kod sebe već predstavlja vezu između klijente aplikacije i drugih podsistema. On omogućava izvršavanje REST upita sa krajnjim tačkama za sledeće funkcionalnosti:

1. Kreiranje mesta
2. Kreiranje filijale u mestu
3. Kreiranje komitenta
4. Promena sedišta za zadatog komitenta
-  5. Otvaranje računa
6. Zatvaranje računa
7. Kreiranje transakcije koja je prenos sume sa jednog računa na drugi račun
-  8. Kreiranje transakcije koja je uplata novca na račun
9. Kreiranje transakcije koja je isplata novca sa računa
10. Dohvatanje svih mesta
11. Dohvatanje svih filijala
12. Dohvatanje svih komitenata
13. Dohvatanje svih računa za komitenta
14. Dohvatanje svih transakcija za račun
15. Dohvatanje svih podataka iz rezervne kopije
16. Dohvatanje razlike u podacima u originalnim podacima i u rezervnoj kopiji

Centralni server zahteve 1, 2, 3, 4, 10, 11, 12 kroz JMS prosleđuje podsistemu 1, dok zahteve 5, 6, 7, 8, 9, 13, 14 kroz JMS prosleđuje podsistemu 2. Zahtev 15 kroz JMS prosleđuje podsistemu 3. U obrati zahteva 16 učestvuju svi podsistemi.

Zahtevi

Potrebno je realizovati sledeće zahteve:

1. Kreirati navedene baze podataka na MySQL serveru i popuniti ih podacima (radovi koji nemaju bazu podataka popunjenu podacima se neće razmatrati). Dostaviti dump fajlove unutar foldera „baze“
2. Implementirati tražene delove sistema. Dostaviti ih unutar foldera „java aplikacije“
3. Za korisnički zahtev obrade transakcije od strane podsistema 2 kreirati UML dijagrame sekvence. Dostaviti ih unutar foldera „uml/sekvenca“.
4. Za svaku aplikaciju dostaviti UML dijagram klase. Dostavi ih unutar foldera „uml/klase“

Domaći zadatak predati kao zip arhivu pod nazivom „ggggbbbbb.zip“, gde su „gggg“ godina, „bbbb“ broj indeksa. Zip arhivu predati putem linka:

https://rti.etf.bg.ac.rs/domaci/index.php?servis=IS1_domaci_21_22.

Napomene

Domaći zadatak se radi samostalno i vredi 20 poena. O tačnim datumima predaje i odbrane domaćeg zadatka bićete blagovremeno obavešteni.

Ukoliko neke stvari u tekstu zadatka nisu dovoljno precizno definisane, postaviti svoje pretpostavke i na osnovu njih nastaviti sa radom. Na ovaj način je omogućen određeni stepen slobode pri izradi.

Za izradu je potrebno koristiti:

- MySQL Server
- MySQL Workbench ili neki drugi MySQL klijentsku aplikaciju
- Netbeans ili neko drugo okruženje

Zahteve pod rednim brojevima 3. i 4. možete realizovati u alatu po izboru i dijagram eksportovati kao sliku ili PDF fajl ili nacrtati na papiru i dostaviti sliku toga.

Na usmenoj odbrani kandidat je potrebno je da samostalno podesi projekte i instalira sve dodatne programe neophodne za ispravan rad priloženog rešenja, ukoliko oni već ne postoje u laboratoriji. Kandidat mora da poseduje potreban nivo znanja o zadatku, mora da bude svestan nedostataka priloženog rešenja i mogućnosti da te nedostatke reši. Kandidat mora tačno da odgovori i na određen broj pitanja koja se bave tematikom domaćeg zadatka i da uradi modifikaciju koja se radi za vreme odbrane ukoliko se to od kandidata bude tražilo.