

Oblikovanje programske potpore

Ak. god. 2019./2020.

Internet bankarstvo

Dokumentacija, Rev. 1

Grupa: *BugBusters*

Voditelj: *Matija Bačić*

Datum predaje: 15. 11. 2019.

Nastavnik: *Hrvoje Nuić*

Sadržaj

1 Dnevnik promjena dokumentacije	3
2 Opis projektnog zadatka	5
3 Specifikacija programske potpore	10
3.1 Funkcionalni zahtjevi	10
3.1.1 Obrasci uporabe	14
3.1.2 Sekvencijski dijagrami	35
3.2 Ostali zahtjevi	43
4 Arhitektura i dizajn sustava	44
4.1 Baza podataka	46
4.1.1 Opis tablica	47
4.1.2 Dijagram baze podataka	53
4.2 Dijagram razreda	54
4.3 Dijagram stanja	63
4.4 Dijagram aktivnosti	64
4.5 Dijagram komponenti	65
5 Implementacija i korisničko sučelje	66
5.1 Korištene tehnologije i alati	66
5.2 Ispitivanje programskog rješenja	68
5.2.1 Ispitivanje komponenti (JUnit testiranje)	68
5.2.2 Ispitivanje sustava	71
5.3 Dijagram razmještaja	72
5.4 Upute za puštanje u pogon	73
6 Zaključak i budući rad	75
Popis literature	76
Indeks slika i dijagrama	78

Dodatak: Prikaz aktivnosti grupe

79

1. Dnevnik promjena dokumentacije

Rev.	Opis promjene/dodatka	Autori	Datum
0.1	Napravljen predložak.	Bačić	15.10.2019.
0.2	Dodani funkcionalni zahtjevi.	Milošević	17.10.2019.
0.3	Dodani obrasci uporabe.	Bastalić Gudelj	20.10.2019.
0.4	Dodani ostali zahtjevi.	Vučemilo	21.10.2019.
0.5	Dodan opis projektnog zadatka.	Bačić	22.10.2019.
0.6	Nadopuna i ispravak funkcionalnih zahtjeva.	Milošević	22.10.2019.
0.7	Dopuna i ispravak obrazaca uporabe.	Bastalić Gudelj	24.10.2019.
0.7	Dopuna i ispravak obrazaca uporabe.	Bačić	24.10.2019.
0.8	Dodani sekvencijski dijagrami.	Milički	26.10.2019.
0.9	Dodani dijagrami obrazaca uporabe za bankara.	Bačić	27.10.2019.
0.10	Dodani ostali dijagrami obrazaca uporabe.	Anušić	27.10.2019.
0.11	Popravljeni sekvencijski dijagrami.	Milički	6.11.2019.
0.12	Dodan opis baze podataka.	Milički Bastalić	7.11.2019.
0.13	Završen opis arhitekture sustava.	Bačić	14.11.2019.
0.13	Dodani dijagrami razreda.	Bačić	15.11.2019.
1.0	Verzija samo s bitnim dijelovima za 1. ciklus	Bačić	15.11.2019.
1.01	Ispravak grešaka u prvoj verziji	Bačić	3.1.2020.
1.1	Popunjeno poglavlje testiranje	Milički	6.1.2020.
1.2	Dodani dijagrami stanja	Bastalić	11.1.2020.
1.3	Dodani dijagrami aktivnosti	Bastalić	11.1.2020.
1.4	Dodani dijagrami komponenti	Milošević	12.1.2020.
1.5	Dodani dijagrami razmještaja	Milošević	13.1.2020.
1.6	Korištene tehnologije i alati	Bastalić	14.1.2020.
1.7	Upute za puštanje u pogon	Bastalić Milošević	14.1.2020.

Rev.	Opis promjene/dodatka	Autori	Datum
1.8	Zaključak	Bastalić	15.1.2020.
1.9	JUnit testiranje	Bačić	16.1.2020.
2.0	Verzija samo s bitnim dijelovima za 2. ciklus	Bačić	16.1.2020.

2. Opis projektnog zadatka

Potreba za kvalitetnom digitalizacijom bankarskih procesa i uvid u stanje pojedinih računa iz udobnosti doma od velikog su značaja kako za klijente, tako i za djelatnike banke. Cilj je ovog projekta razviti bankarski sustav: bazu podataka i sučelja prema bankarima, administratorima te krajnjim korisnicima banke, odnosno građanima. Korisnike dijelimo u nekoliko skupina :

- **administratori,**
- **bankari,**
- **službenici za odobravanje kreditnih zahtjeva građana i**
- **korisnici u užem smislu (građani).**

Svaka vrsta korisnika sustava ima svoj jedinstveni pogled u sustav putem web sučelja i Android aplikacije.

Neregistrirani korisnik, za registraciju u sustav upisuje svoj OIB i jedinstveni ključ koji je dobio od bankara. Nakon uspješnog unosa tih podataka, odabire korisničko ime i lozinku koji će mu u buduću služiti za prijavu u sustav. Ostale podatke o pojedinom korisniku kao što su: ime, prezime, adresa prebivališta, datum rođenja, e-mail adresa, sliku korisnika i dr., u sustav unosi bankarski službenik prilikom ugovaranja bilo kakve usluge banke.

Administrator u sustav ulazi unosom svoje korisničkog imena i lozinke koji mu daju pristup sustavu. On vidi popis svih korisnika sustava te može uređivati podatke za bilo koju vrstu korisnika, kao i dodavati te brisati same korisnike sustava. Jedina je osoba koja može kreirati račune ostalim djelatnicima banke.

Bankar svoj korisnički račun dobivaju od administratora. Ime, prezime, adresa prebivališta, OIB, datum rođenja, e-mail adresa i slika profila unaprijed su poznati podaci koje u sustav unosi administrator za svakog novog djelatnika. Pošto postanu dio kolektiva banke te nakon što administrator unese njihove podatke u sustav, u web aplikaciju unose korisničko ime i privremenu lozinku dobivenu od administratora sustava. Nakon prijave privremenom lozinkom, obavezni su odmah stvoriti novu lozinku. U sustavu imaju uvid u sve svoje osobne podatke, ali nemaju mogućnost izmjene tih podataka. Bankar nema uvid u popis korisnika sustava, niti u popis klijenata banke (građana). On može otvoriti, obrisati ili mijenjati

profile klijenata. Kako bi pronašao profil nekog građana i uređivao ga ili brisao, dužan je unijeti OIB osobe za koju radi izmjenu u sustavu nakon čega mu se otvara profil te osobe (pretraga profila radi se po OIB-u građana). Također mu je dozvoljeno otvarati i zatvarati tekuće, žiro, štedne račune, ugovarati kredite, kreditne i debitne kartice za korisnike banke, aktivacija i deaktivacija kartica te obavljanje transakcija po računima klijenata.

Svaki tekući račun ima dozvoljeno prekoračenje, a svaka kreditna kartica ima ugovoren limit.

Klijent dobiva dozvolu pristupa sustavu od bankara. Klijent banke ima uvid u sve podatke o svom profilu (ime, prezime, OIB, prebivalište, datum rođenja, slika profila i drugo), ali ih ne može mijenjati. Također, klijent može vidjeti sve svoje transakcijske račune, štedne račune, ugovorene kredite i kartice (debitne i kreditne). Klijent ne može mijenjati osobne podatke, podatke o računima, kreditima ili karticama. Klijent može vršiti prijenos sredstava između svojih transakcijskih računa (tekućeg, žiro ili štednog) ili na račune drugih klijenata banke. Također, klijent može ručno izvršiti uplatu kredit ili na kreditnu karticu kako bi umanjio iznos dugovanja ili otplatiti dugovanje ukupno. Klijent podnosi zahtjev za kreditom ili kreditnom karticom putem aplikacije ili u poslovnici banke gdje taj zahtjev unosi bankarski službenik. Za izradu kreditnog zahtjeva, u sustav se unosi iznos kredita, namjena kredita i rok otplate, dok kamatna stopa ovisi isključivo o namjeni kredita. Za izradu zahtjeva za kreditnom karticom, u sustav je potrebno unijeti samo vrstu kartice koja se traži (Mastercard, Visa, American Express, Diners, Discover i dr.).

Svi korisnici moraju imati sliku profila (standardna slika za osobnu iskaznicu).

Jednom mjesečno, u web aplikaciji, korisniku dolazi izvod po svim transakcijskim računima i kreditnim karticama u PDF i XLS obliku.

Službenici za odobravanje kreditnih zahtjeva građana imaju uvid u svoje osobne podatke koje ne mogu mijenjati. Također, vidljivi su im svi neobrađeni kreditni zahtjevi svih klijenata banke. Službenik može preuzeti bilo koji zahtjev na obradu te tada ima uvid u podatke o zahtjevu i sve podatke o klijentu koje banka posjeduje. Podaci o zahtjevu sadrže iznos zatraženog kredita, namjena kredita, kamatna stopa i vrijeme otplate ako se radi o zahtjevu za kredit, odnosno vrsta zatražene kreditne kartice ako se radi o zahtjevu za kreditnom karticom. Jedini podatak od navedenih koji se ne može mijenjati jest kamatna stopa jer ona ovisi o namjeni kredita. Službenik tada odobrava ili odbija zahtjev za kreditom, odnosno odobrava ili

odbija kreditnu karticu kojoj ručno stvara određeni limit i kamatnu stopu. Nakon donesene odluke, korisnik tu odluku vidi u aplikaciji.

Rate za kredite clijenata skidaju se automatski svaki mjesec na datum kojeg odredi administrator sustava i on je jedinstven za sve. U slučaju da na tekućem računu korisnika ne postoji dovoljno sredstava, ta se rata odgađa za jedan mjesec i bit će naplaćena zajedno s novom ratom idući mjesec.

Dugovanje po kreditnoj kartici korisnik mora ručno otplatiti. Ako se u jednom mjesecu do unaprijed određenog datuma ne otplati račun u cijelosti, na preostali iznos zaračunava se kamata.

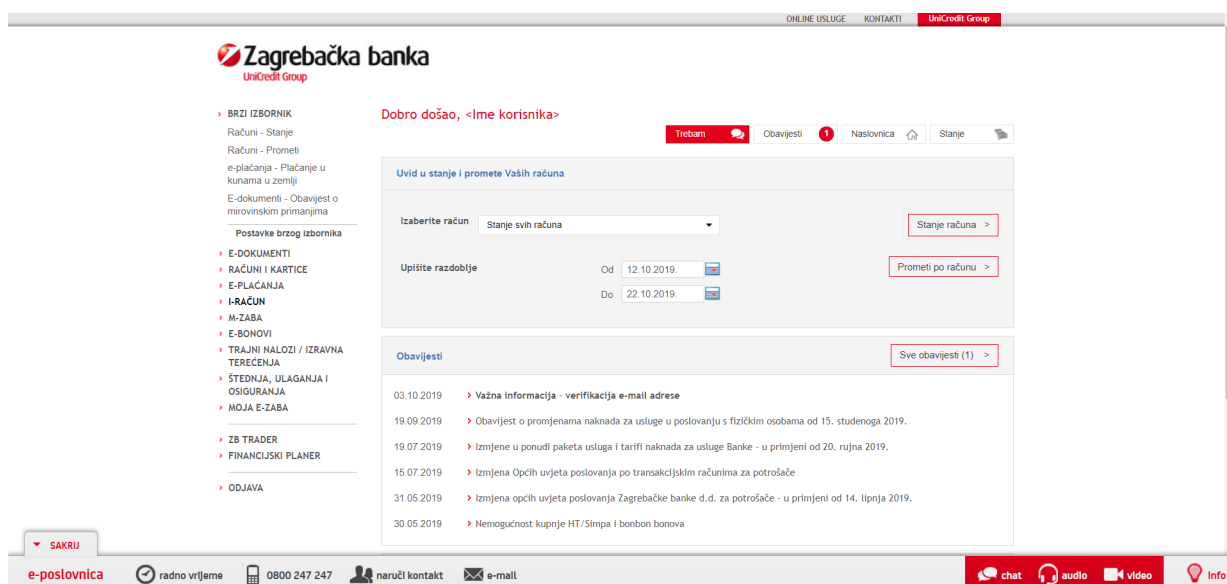
Debitne kartice u sustavu prikazuju se odvojeno od tekućeg ili žiro računa kojim pripadaju. Može postojati račun bez debitne kartice, ali ne i debitna kartica bez računa. Odnosno, svaka debitna kartica mora biti vezana uz jedan račun.

Važno je napomenuti da odvajamo pojam **profila korisnika sustava** od **korisničkog računa**.

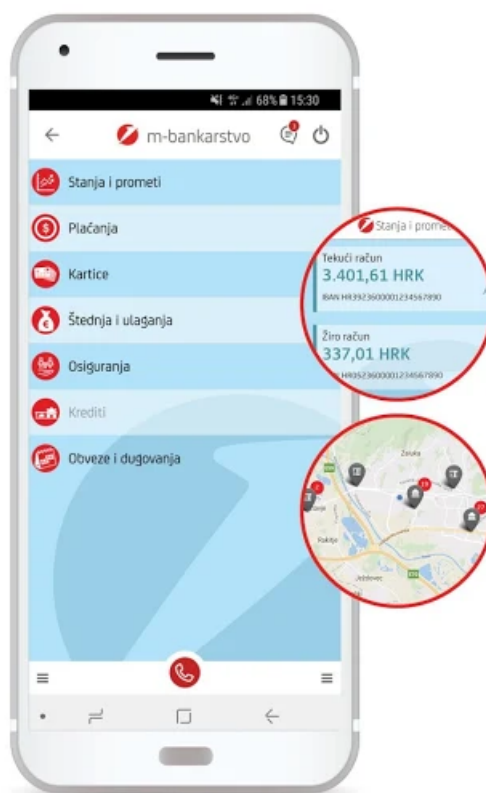
Profil korisnika sustava može postojati i bez korisničkog računa. On se sastoji od svih osobnih podataka korisnika (ime, prezime, OIB, adresa prebivališta, datum rođenja, e-mail adresa i slika profila), razine pristupa sustavu (administrator, bankar, službenik za odobravanje kredita ili klijent banke) te, ukoliko se radi o klijentu banke, svih njegovih ugovorenih usluga (transakcijskih računa, štednih računa, kredita, kreditnih kartica i debitnih kartica).

Pod pojmom **korisničkog računa** podrazumijevamo način na koji korisnik pristupa podacima u sustavu i aktivnostima koji su mu dozvoljeni nad sustavom. Korisnički račun jest način kako korisnik sustava dobiva ovlast pregledavati sve podatke u sustavu koji su mu dozvoljeni. Njega definira korisničko ime i lozinka te svaki od vrste korisnika na drugačiji način (ranije definiran) dobiva korisnički račun. Svaki korisnički račun veže se uz jedan od korisničkih profila, odnosno korisnički račun ne može postojati bez profila korisnika.

Primjeri ovakvog sustava mogu se pronaći u svim modernim, vodećim bankama u svijetu i u Hrvatskoj. Neki od njih prikazani su u nastavku.



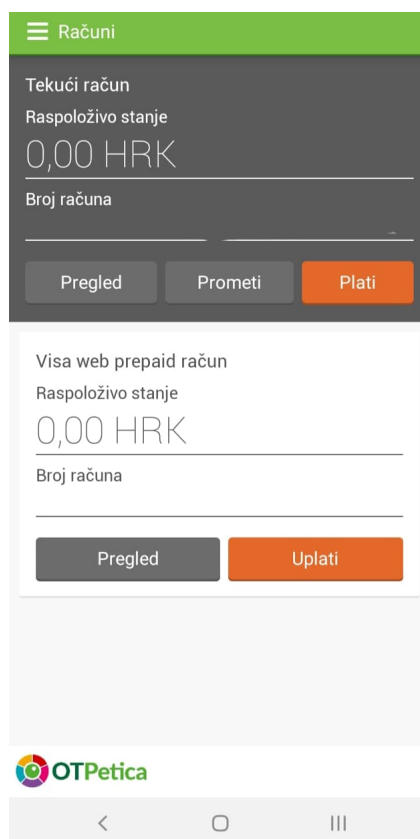
Slika 2.1: Usluga internet bankarstva Zagrebačke banke



Slika 2.2: Usluga mobilnog bankarstva Zagrebačke banke



Slika 2.3: Usluga internet bankarstva OTP banke



Slika 2.4: Usluga mobilnog bankarstva OTP banke

3. Specifikacija programske potpore

3.1 Funkcionalni zahtjevi

Dionici:

1. Banka
2. Administrator sustava
3. Službenici banke
 - (a) Bankari
 - (b) Službenici za odobravanje kredita građana
4. Klijenti banke

Aktori i njihovi funkcionalni zahtjevi:

1. Administrator - inicijator
 - (a) Pristup sustavu pomoću korisničkog imena i lozinke
 - (b) Pregled popisa svih profila i uz njih vezanih korisničkih računa
 - (c) Dodavanje novih profila u sustav
 - i. Unosi ime, prezime, adresu prebivališta, OIB, datum rođenja, e-mail adresu, sliku profila, vrsta korisnika
 - (d) Izmjena podataka profila i mijenjanje razine pristupa korisnicima aplikacije
 - (e) Brisanje profila korisnika
 - (f) Dodavanje novog korisničkog računa
 - i. Odabir korisničkog imena, sustav generira privremenu lozinku koju je potrebno promijeniti prilikom prve prijave
 - (g) Brisanje korisničkog računa pojedinih profila

2. Baza podataka - sudionik

- (a) Sadrži podatke o profilima korisnika sustava
 - i. Ime, prezime, adresa, OIB, datum rođenja, e-mail adresa, slika profila, vrsta korisnika
 - ii. Sve korisničke račune vezane za jedan profil
- (b) Sadrži podatke o računima klijenata
 - i. Broj i stanje tekućeg računa
 - ii. Broj i stanje žiro računa
 - iii. Broj i stanje štednog računa
- (c) Sadrži podatke o karticama klijenata
 - i. Broj i vrsta kartice
 - A. Debitna kartica - vezana uz račun
 - B. Kreditna kartica - nije vezana uz račun
 - ii. Ukoliko je kartica kreditna, sadrži ukupno dugovanje i ukupni odobreni limit po kartici
- (d) Sadrži podatke o transakcijama među računima klijenata banke
- (e) Sadrži podatke o transakcijama po debitnim i kreditnim karticama
- (f) Sadrži podatke o kreditima
 - i. Iznos, namjena, rok otplate kredita u mjesecima i kamatna stopa
 - ii. Ukupno preostalo dugovanje (u početku jest iznos uvećan za kamatnu stopu, smanjuje se s otplatom)

3. Bankar - inicijator

- (a) Inicijalnu prijavu u sustav vrši pomoću korisničkog imena i privremene lozinke dobivene od administratora
 - i. Nakon inicijalne prijave obvezan je odmah stvoriti novu lozinku
- (b) Mogućnost uvida u vlastiti profil
 - i. Ime, prezime, adresa, OIB, datum rođenja, e-mail adresa, slika profila
- (c) Izrada korisničkih profila klijenata banke
 - i. Unosi ime, prezime, adresu, OIB, datum rođenja, e-mail adresu, sliku profila klijenta
- (d) Brisanje korisničkih profila klijenata
- (e) Izmjena podataka u korisničkim profilima klijenata

- (f) Pretraga profila klijenata pomoću OIB-a klijenta
- (g) Otvaranje i zatvaranje tekućih, žiro i štednih računa
- (h) Obavljanje transakcija po računima klijenata
- (i) Ugovaranje, aktivacija i deaktivacija debitnih i kreditnih kartica
- (j) Ugovaranje kredita
- (k) Otvaranje korisničkih računa za profile klijenata
 - i. Otvaranjem korisničkih računa dobiva prikaz privremenog ključa kojeg korisnik unosi uz svoj OIB kako bi odabrao korisničko ime i lozinku
- (l) Brisanje korisničkih računa za profile klijenata

4. Službenik za odobravanje kreditnih zahtjeva - sudionik

- (a) Inicijalnu prijavu u sustav vrši pomoću korisničkog imena i privremene lozinke dobivene od administratora
 - i. Nakon inicijalne prijave obvezan je odmah stvoriti novu lozinku
- (b) Mogućnost uvida u vlastiti profil
 - i. Ime, prezime, adresa, OIB, datum rođenja, e-mail adresa, slika profila
- (c) Odobrenje i blokiranje kreditnih zahtjeva
- (d) Uvid u sve podatke o profilima klijentima čiji zahtjevi za kreditom nisu riješeni

5. Klijent banke - inicijator

- (a) Pregledavanje podataka isključivo o vlastitom profilu
 - i. Ime, prezime, OIB, adresa, datum rođenja, e-mail adresa, slika profila
 - ii. Sve ugovorene usluge banke (transakcijski računi, krediti, debitne i kreditne kartice, štedni računi i dr.)
- (b) Pregledavanje podataka o transakcijama
- (c) Prijenos sredstava između vlastitih računa
- (d) Prijenos sredstava na račun drugih klijenata
- (e) Može ručno izvršiti uplatu kredita ili uplatu na kreditnu karticu kako bi umanjio iznos duga ili kako bi otplate sveukupno dugovanje
- (f) Podnošenje zahtjeva za kreditom
 - i. Unosi iznos i namjenu kredita, te rok otplate

(g) Podnošenje zahtjeva za kreditnom karticom

i. Unos vrste kartice (Mastercard, Visa, American Express, Diners, Discover)

6. Neregistrirani korisnik - inicijator

(a) Upisuje OIB i jedinstveni ključ dobiven od bankara kako bi se registrirao u sustav

(b) Nakon registracije odabire korisničko ime i lozinku koji će mu služiti za buduće prijave u sustav

3.1.1 Obrasci uporabe

Opis obrazaca uporabe

UC1-Prijava u sustav

- **Glavni sudionik:** Svi korisnici sustava
- **Cilj:** Dobiti pristup sustavu
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik ima korisnički račun
- **Opis osnovnog tijeka:**
 1. Korisnik sustava unosi korisničko ime i lozinku
 2. Potvrda o ispravnosti korisničkih podataka
 3. Korisnik sustava dobiva pristup sustavu
- **Opis mogućih odstupanja:**
 - 2.a Unos neispravnih podataka
 1. Ispis odgovarajuće poruke upozorenja

UC2-Pregled profila svih korisnika sustava

- **Glavni sudionik:** Administrator
- **Cilj:** Pregledati registrirane korisnike i njihove osobne podatke
- **Sudionici:** Baza podataka
- **Preduvjet:** : Korisnik je registriran i dodijeljena su mu prava administratora
- **Opis osnovnog tijeka:**
 1. Administrator odabire opciju pregledavanja korisnika
 2. Prikazuje se lista svih profila korisnika

UC3-Dodavanje novih profila u sustav

- **Glavni sudionik:** Administrator
- **Cilj:** Dodati profil u sustav
- **Sudionici:** Baza podataka
- **Preduvjet:** Profil se ne nalazi u bazi podataka
- **Opis osnovnog tijeka:**
 1. Administrator unosi potrebne osobne podatke
 - (a) Ime, prezime, adresa, OIB, datum rođenja, e-mail adresa, slika profila, vrsta korisnika (razina pristupa sustavu)
 2. Podaci se unose u bazu podataka

3. Stvoren je novi profil

- **Opis mogućih odstupanja:**

1.a Unos neispravnih podataka (krivi OIB, profil s tim OIB-om ime već postoji i dr.)

1. Ispis odgovarajuće poruke upozorenja

UC4-Izmjena podataka profila korisnika

- **Glavni sudionik:** Administrator

- **Cilj:** Urediti i izmijeniti podatke profila

- **Sudionici:** Baza podataka

- **Preduvjet:** Korisnik je prijavljen u sustav kao administrator

- **Opis osnovnog tijeka:**

1. Administrator otvara željeni profil iz popisa svih korisnika

2. Administrator odabire opciju uređivanja profila

3. Administrator izmjenjuje podatke korisnika i sprema ih u bazu podataka

UC5-Brisanje profila korisnika

- **Glavni sudionik:** Administrator

- **Cilj:** Obrisati korisnika

- **Sudionici:** Baza podataka

- **Preduvjet:** Korisnik je prijavljen u sustav kao administrator

- **Opis osnovnog tijeka:**

1. Administrator otvara željeni profil iz popisa svih korisnika

2. Administrator odabire opciju brisanja profila

3. Uklanja se korisnik i njegovi podaci iz baze podataka

UC6 - Promjena razine pristupa profila

- **Glavni sudionik:** Administrator

- **Cilj:** Promijeniti razinu pristupa korisnika sustavu (administrator, bankar, službenik za odobravanje kreditnih zahtjeva klijenata, klijent banke)

- **Sudionici:** Baza podataka

- **Preduvjet:** Korisnik je prijavljen u sustav kao administrator

- **Opis osnovnog tijeka:**

1. Administrator odabire željenog korisnika s popisa profila

2. Izmijeni razinu pristupa korisnika

UC7 - Dodavanje novog korisničkog računa

- **Glavni sudionik:** Administrator
- **Cilj:** Dozvoliti pristup sustavu nekom korisniku
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen u sustav kao administrator, postoji profil korisnika kome želimo dodati korisnički račun
- **Opis osnovnog tijeka:**
 1. Administrator otvara željeni profil iz popisa svih korisnika
 2. Administrator odabire opciju dodavanja korisničkog računa
 3. Administrator odabire korisničko ime i privremenu lozinku koja se mora promijeniti prilikom prve prijave s tim vjerodajnicama
- **Opis mogućih odstupanja:**
 - 3.1 Unos već postojećeg korisničkog ime
 1. Ispis odgovarajuće poruke upozorenja
 2. Korisnički račun ne dodaje se u sustav

UC8 - Brisanje korisničkog računa

- **Glavni sudionik:** Administrator
- **Cilj:** Onemogućiti pristup sustavu nekom korisniku
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen u sustav kao administrator, postoji profil korisnika koji ima korisnički račun vezan za sebe
- **Opis osnovnog tijeka:**
 1. Administrator otvara željeni profil iz popisa svih korisnika
 2. U popisu korisničkih računa vezanih za odabrani profil, odabere onog kojeg želi obrisati
 3. Administrator potvrđuje opciju uklanjanja korisničkog računa

UC9 - Inicijalna prijava bankara

- **Glavni sudionik:** Bankar
- **Cilj:** Aktivirati korisnički račun za pristup sustavu
- **Sudionici:** Baza podataka
- **Preduvjet:** Dobivena privremena lozinka i korisničko ime od administratora
- **Opis osnovnog tijeka:**
 1. Bankar odabire opciju za prijavu

2. Bankar unosi korisničko ime i privremenu lozinku dobivene od administratora
 3. Bankar mijenja privremenu lozinku
 4. U bazu podataka se pohrani promjena
- **Opis mogućih odstupanja:**
 - 2.a Upisana kriva kombinacija korisničkog imena i privremen lozinke
 - (a) Sustav obavještava korisnika o neuspjelom upisu
 - 3.a Korisnikova nova lozinka identična je privremenoj
 - (a) Sustav obavještava korisnika o obaveznom mijenjanju privremene lozinke

UC10 - Uvid u vlastite osobne podatke

- **Glavni sudionik:** Bankar
- **Cilj:** Pregledati osobne podatke
- **Sudionici:** Baza podatak
- **Preduvjet:** Korisnik je prijavljen u sustav i dodijeljena su mu prava bankara
- **Opis osnovnog tijeka:**
 1. Korisnik odabire opciju "Osobni podatci"
 2. Aplikacija prikazuje osobne podatke : ime, prezime, adresu, OIB, datum rođenja, e-mail adresu i sliku profila korisnika

UC11 - Izrada novog korisničkog profila klijenta banke

- **Glavni sudionik:** Bankar
- **Cilj:** Dodati novi korisnički profil
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen u sustav i dodijeljena su mu prava bankara te klijent nema korisnički profil
- **Opis osnovnog tijeka:**
 1. Bankar odabire opciju izrade novog korisničkog profila klijenta
 2. Otvara se prozor za upis osobnih podataka klijenta
 3. Bankar upiše ime, prezime, adresu, OIB, datum rođenja, e-mail adresu, sliku profila i potvrdi upis
 4. U bazu podataka se pohrani promjena

UC12 - Brisanje korisničkog profila klijenta

- **Glavni sudionik:** Bankar

- **Cilj:** Obrisati korisnički profil
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen u sustav i dodijeljena su mu prava bankara te klijent ima korisnički profil
- **Opis osnovnog tijeka:**
 1. Bankar u tražilicu upisuje OIB klijenta
 2. Otvara se profil traženog klijenta sa svim njegovim podacima i opcijom "obriši profil"
 3. Bankar odabere opciju "obriši profil" i potvrđuje odabir
 4. U bazu podataka se pohrani promjena (briše se i profil i svi korisnički računi klijenta vezani uz profil)
- **Opis mogućih odstupanja:**
 - 1.a Upisan nepostojeći OIB
 - (a) Sustav obavještava bankara o neuspjelom upisu OIB-a

UC13 - Ažuriranje korisničkog profila klijenta

- **Glavni sudionik:** Bankar
- **Cilj:** Izmijeniti korisnički profil klijenta
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen u sustav i dodijeljena su mu prava bankara te klijent ima korisnički profil
- **Opis osnovnog tijeka:**
 1. Bankar u tražilicu upisuje OIB klijenta
 2. Bankar pronalazi željenog korisnika
 3. Bankar mijenja podatke u profilu klijenta
 4. Bankar odabire opciju "Spremi"
 5. U bazu podataka se pohrani promjena
 6. Korisnik vidi promjene korisničkog profila u aplikaciji
- **Opis mogućih odstupanja:**
 - 1.a Upisan nepostojeći OIB
 1. Sustav obavještava bankara o neuspjelom upisu OIB-a
 - 3.a Bankar unosi novi nevažeći OIB
 1. Sustav obavještava korisnika da nije spremio podatke

UC14 - Otvaranje računa

- **Glavni sudionik:** Bankar

- **Cilj:** Otvoriti račun klijentu
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen u sustav i dodijeljena su mu prava bankara te klijent ima korisnički profil
- **Opis osnovnog tijeka:**
 1. Bankar u tražilicu upisuje OIB klijenta
 2. Bankar pronalazi željenog korisnika
 3. Bankar odabire opciju "Izrada računa"
 4. Odabire iz padajućeg izbornika tekući, žiro ili štedni
 5. Odabire opciju "Otvori"
 6. Promjene spremljene u bazu podataka
 7. Klijentu se omogućuje prikaz računa u aplikaciji - broj računa i iznos sredstava na računu

UC15 - Zatvaranje računa

- **Glavni sudionik:** Bankar
- **Cilj:** Zatvoriti račun klijentu
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen u sustav i dodijeljena su mu prava bankara te klijent ima korisnički profil i otvoren transakcijski račun
- **Opis osnovnog tijeka:**
 1. Bankar u tražilicu upisuje OIB klijenta
 2. Bankar pronalazi željenog korisnika
 3. Bankar pored željenog računa odabire opciju "Zatvaranja računa"
 4. Odabire opciju "Zatvori" (potvrđuje odabrano)
 5. U bazu podataka se pohrani promjena

UC16 - Obavljanje transakcija po računima klijenata

- **Glavni sudionik:** Bankar
- **Cilj:** Obaviti transakciju po računu klijenata
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen u sustav i dodijeljena su mu prava bankara te klijent ima korisnički profil i otvoren transakcijski račun
- **Opis osnovnog tijeka:**
 1. Bankar u tražilicu upisuje OIB klijenta
 2. Bankar pronalazi željenog korisnika

3. Bankar odabire opciju "Transfer"
 4. Otvara se novi prozor u kojem je za račun terećenja padajući izbornik - mogućnost odabira samo računa trenutnog klijenta
 5. Otvara se polje za upis računa odobrenja i bankar upisuje bilo čiji broj računa i iznos transfera te poziv na broj transakcije
 6. U bazu podataka se pohrani promjena
- **Opis mogućih odstupanja:**
 - 3.a Upisan nepostojeći račun odobrenja ili ako je iznos transfera veći nego što klijent ima na računu terećenja
 1. Sustav obavještava bankara o neuspjelom upisu

UC17 -Ugovaranje debitne kartice

- **Glavni sudionik:** Bankar
- **Cilj:** Ugovoriti debitnu karticu klijentu
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen u sustav i dodijeljena su mu prava bankara
- **Opis osnovnog tijeka:**
 1. Bankar u tražilicu upisuje OIB klijenta
 2. Bankar pronalazi željenog korisnika
 3. Bankar odabire opciju "Dodaj karticu" pored željenog transakcijskog računa za račun koji nema debitnu karticu
 4. Promjene se spremaju u bazu podataka
 5. Dodaje se mogućnost uvida u broj kartice i račun uz koji je vezana preko korisničkog profila klijenta
- **Opis mogućih odstupanja:**
 - 1.a Upisan nepostojeći OIB klijenta
 1. Sustav obavještava bankara o neuspjelom upisu OIB-a

UC18 -Ugovaranje kreditnih kartica

- **Glavni sudionik:** Bankar
- **Cilj:** Ugovoriti kreditnu karticu klijentu
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen u sustav i dodijeljena su mu prava bankara
- **Opis osnovnog tijeka:**
 1. Bankar u tražilicu upisuje OIB klijenta
 2. Bankar pronalazi željenog korisnika

3. Bankar odabire opciju "Ugovaranje kreditne kartice"
 4. Bankar bira vrstu kreditne kartice (Visa, MasterCard, Diners, American Express, Discover i dr.)
 5. Promjene se spremaju u bazu podataka
 6. Zahtjev odlazi službeniku za odobravanje kreditnih zahtjeva na odobravanje
- **Opis mogućih odstupanja:**
 - 1.a Upisan nepostojeći OIB klijenta
 1. Sustav obavještava bankara o neuspjelom upisu OIB-a

UC19 - Aktiviranje kartica klijenta

- **Glavni sudionik:** Bankar
- **Cilj:** Aktivirati karticu klijentu
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen u sustav i dodijeljena su mu prava bankara te je klijent zatražio aktivaciju kartice
- **Opis osnovnog tijeka:**
 1. Bankar u tražilicu upisuje OIB klijenta
 2. Bankar pronalazi željenog korisnika
 3. Bankar u popisu kartica odabire opciju "Aktivacija kartice" uz željenu karticu
 4. Bankar odabire opciju "Potvrdi"
 5. U bazu podataka se pohrani promjena
 6. Omogućeno je provođenje transakcija karticom
- **Opis mogućih odstupanja:**
 - 1.a Upisan nepostojeći OIB klijenta
 1. Sustav obavještava bankara o neuspjelom upisu OIB-a

UC20 - Deaktiviranje kartica klijenta

- **Glavni sudionik:** Bankar
- **Cilj:** Deaktivirati karticu klijentu
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen u sustav i dodijeljena su mu prava bankara te je klijent zatražio deaktivaciju kartice
- **Opis osnovnog tijeka:**
 1. Bankar u tražilicu upisuje OIB klijenta

2. Bankar pronalazi željenog korisnika
 3. Bankar u popisu kartica odabire opciju "Dekativacija kartice" uz željenu karticu
 4. Bankar odabire opciju "Potvrdi"
 5. U bazu podataka se pohrani promjena
 6. Onemogućeno je provođenje transakcija karticom
- **Opis mogućih odstupanja:**
 - 1.a Upisan nepostojeći OIB klijenta
 1. Sustav obavještava bankara o neuspjelom upisu OIB-a

UC21 - Ugovaranje kredita

- **Glavni sudionik:** Bankar
- **Cilj:** ugovoriti kredit klijentu
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen u sustav i dodijeljena su mu prava bankara te je klijent zatražio ugovaranje kredita
- **Opis osnovnog tijeka:**
 1. Bankar u tražilicu upisuje OIB klijenta
 2. Bankar pronalazi željenog korisnika
 3. Bankar odabire opciju "Ugovaranje kredita"
 4. Unos odgovarajućih podataka
 - (a) Iznos, namjena kredita, rok otplate kredita u mjesecima
 - (b) Kamatna stopa određena je namjenom kredita
 5. U bazu podataka se pohrani promjena
 6. Klijent se dodaje na listu koja se šalje službeniku za odobravanje kredita
- **Opis mogućih odstupanja:**
 - 1.a Upisan nepostojeći OIB klijenta
 1. Sustav obavještava bankara o neuspjelom upisu OIB-a

UC22 - Otvaranje korisničkih računa za profile klijenata

- **Glavni sudionik:** Bankar
- **Cilj:** Omogućiti klijentu pristup sustavu
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen u sustav i dodijeljena su mu prava bankara te je klijent zatražio otvaranje korisničkog računa
- **Opis osnovnog tijeka:**

1. Bankar u tražilicu upisuje OIB klijenta
 2. Bankar pronalazi željenog korisnika
 3. Bankar odabire opciju "Dodaj korisnički račun"
 4. Bankar dobiva privremeni ključ za klijenta
- **Opis mogućih odstupanja:**
 - 1.a Upisan nepostojeći OIB klijenta
 1. Sustav obavještava bankara o neuspjelom upisu OIB-a

UC23 - Zatvaranje korisničkih računa za profile klijenata

- **Glavni sudionik:** Bankar
- **Cilj:** Onemogućiti klijentu pristup sustavu
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je prijavljen u sustav i dodijeljena su mu prava bankara te je klijent zatražio zatvaranje korisničkog računa
- **Opis osnovnog tijeka:**
 1. Bankar u tražilicu upisuje OIB klijenta
 2. Bankar pronalazi željenog korisnika
 3. Odabire opciju "obriši korisnički račun" pored željenog računa s popisa svih vezanih računa
 4. U bazu podataka se pohrani promjena
- **Opis mogućih odstupanja:**
 - 1.a Upisan nepostojeći OIB klijenta
 1. Sustav obavještava bankara o neuspjelom upisu OIB-a

UC24 - Odobrenje kreditnih zahtjeva

- **Glavni sudionik:** Službenik za odobravanje kredita
- **Cilj:** Odobriti zahtjev za kredit klijentu
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je registriran i dodijeljena su mu prava službenika za odobravanje kreditnih zahtjeva
- **Opis osnovnog tijeka:**
 1. Preuzimanje liste zahtjeva za kredit
 2. Odabir jednog od zahtjeva
 - (a) Ako je zahtjev za kreditnom karticom, službenik dodatno unosi odobreni limit po kartici te kamatnu stopu na iskorišteni dio limita
 3. Odabir opcije "odobri"

4. Spremanje promjena u bazu podataka
5. Odluka o odobrenju postaje vidljiva korisniku unutar aplikacije

UC25 - Blokiranje kreditnih zahtjeva

- **Glavni sudionik:** Službenik za odobravanje kredita
- **Cilj:** Blokirati zahtjev za kredit klijentu
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je registriran i dodijeljena su mu prava službenika za odobravanje kreditnih zahtjeva
- **Opis osnovnog tijeka:**
 1. Preuzimanje liste zahtjeva za kredit
 2. Odabir jednog od zahtjeva
 3. Odabir opcije "blokiraj"
 4. Spremanje promjena u bazu podataka
 5. Odluka o blokiranju postaje vidljiva korisniku unutar aplikacije

UC26 - Uvid u podatke o klijentima

- **Glavni sudionik:** Službenik za odobravanje kredita
- **Cilj:** Uvid u profil klijenta za kojeg djelatnik obrađuje zahtjev
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je registriran i dodijeljena su mu prava službenika za odobravanje kreditnih zahtjeva te je klijent podnio zahtjev za kredit
- **Opis osnovnog tijeka:**
 1. S liste zahtjeva za kredit, službenik bira klijenta čije osobne podatke želi pregledati u svrhu odobravanja kredita
 2. Prikazuju se svi dostupni podaci u profilu klijenta: ime, prezime, adresa, OIB, datum rođenja, e-mail adresa, slika profila, podaci o računima: broj i stanje računa, podaci o kreditnim karticama, podaci o ugovorenim kreditima: iznos, trajanje otplate, podaci o kreditnim karticama i ostalo

UC27 - Pregled podataka o vlastitom profilu

- **Glavni sudionik:** Službenik za odobravanje kredita
- **Cilj:** Pregledati podatke na vlastitom profilu
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik sustava je prijavljen
- **Opis osnovnog tijeka:**

1. Korisnik sustava odabire opciju pregleda osobnih podataka
2. Aplikacija prikazuje osobne podatke korisnika

UC28 - Pregled podataka o vlastitom profilu

- **Glavni sudionik:** Klijent
- **Cilj:** Pregledati podatke na vlastitom profilu
- **Sudionici:** Baza podataka
- **Preduvjet:** Klijent je prijavljen
- **Opis osnovnog tijeka:**
 1. Korisnik odabire opciju pregleda osobnih podataka
 2. Aplikacija prikazuje sve podatke korisnika
 - (a) Osobne podatke (ime, prezime, adresa prebivališta, OIB, datum rođenja, e-mail adresa, slika profila)
 - (b) Podatke o računima (tekućim, žiro i štednim) : brojevi računa i stanje na računu
 - (c) Podatke o karticama (debitnim i kreditnim) : broj kartice, broj računa za koji je kartica vezana (debitna), odobren i iskorišten limit kartice (kreditna)
 - (d) Podatke o kreditima (Iznos kredita, kamatna stopa, rok otplate, datum ugovaranja, preostalo dugovanje)

UC29 - Pregled podataka o transakcijama

- **Glavni sudionik:** Klijent
- **Cilj:** Pregledati podatke o transakcijama
- **Sudionici:** Baza podataka
- **Preduvjet:** Klijent je prijavljen
- **Opis osnovnog tijeka:**
 1. Korisnik odabire opciju pregleda podataka o transakcijama pored računa u kojeg želi izvršiti uvid
 2. Aplikacija prikazuje podatke o transakcijama željenog računa

UC30 - Prijenos sredstava s računa

- **Glavni sudionik:** Klijent
- **Cilj:** Prijenos novčanih sredstava s računa
- **Sudionici:** Baza podataka
- **Preduvjet:** Klijent je prijavljen

- **Opis osnovnog tijeka:**
 1. Korisnik odabire opciju prijenosa sredstava
 2. Korisnik odabire račun terećenja preko padajućeg izbornika (jedan od svojih transakcijskih računa)
 3. Korisnik upisuje broj računa odobrenja, broj računa za otplatu kredita ili broj računa za otplatu kreditne kartice
 4. Korisnik upisuje iznos plaćanja i pritisne plati
 5. Aplikacija uspješno obavlja transakciju
- **Opis mogućih odstupanja:**
 - 4.a Unos iznosa plaćanja većeg od stanja na računu terećenja
 1. Ispis odgovarajuće poruke upozorenja

UC31 - Podnošenje zahtjeva za kreditom

- **Glavni sudionik:** Klijent
- **Cilj:** Podnijeti zahtjev za kreditom
- **Sudionici:** Baza podataka
- **Preduvjet:** Klijent je prijavljen
- **Opis osnovnog tijeka:**
 1. Korisnik odabire opciju zahtjeva za kredit
 2. Korisnik unosi iznos, namjenu kredita i rok otplate
 3. Sustav prosljeđuje zahtjev službeniku za odobravanje kreditnih zahtjeva građana

UC32 - Podnošenje zahtjeva za kreditnom karticom

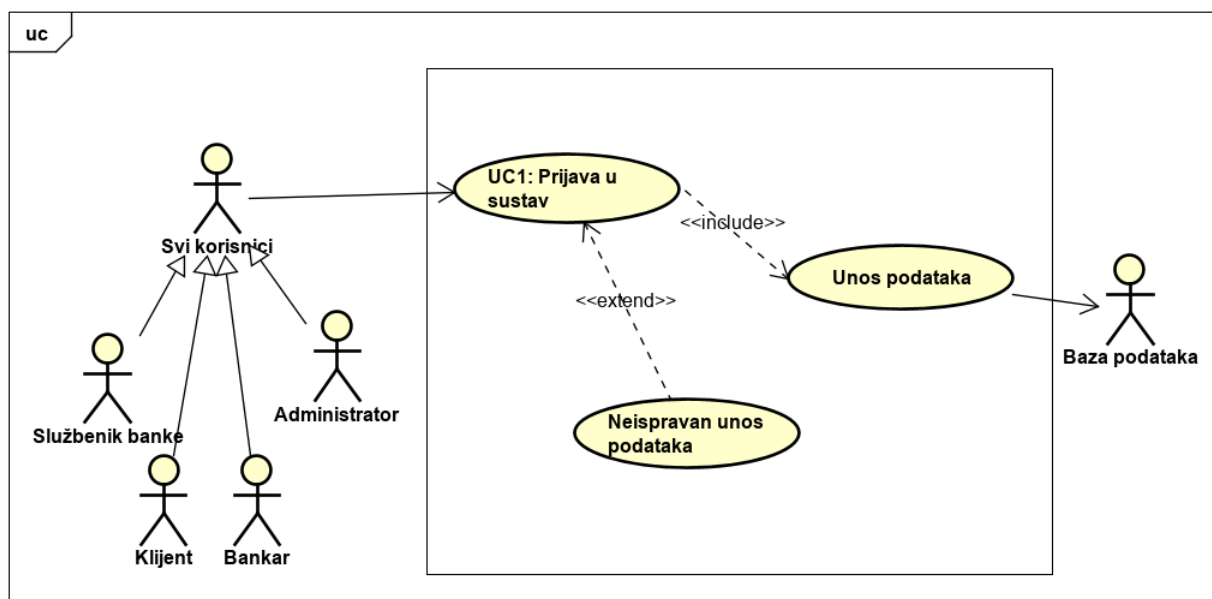
- **Glavni sudionik:** Klijent
- **Cilj:** Podnijeti zahtjev za kreditnom karticom
- **Sudionici:** Baza podataka
- **Preduvjet:** Klijent je prijavljen
- **Opis osnovnog tijeka:**
 1. Korisnik odabire opciju zahtjeva za kreditnom karticom
 2. Korisnik odabire vrstu kreditne kartice
 3. Sustav prosljeđuje zahtjev službeniku za odobravanje kreditnih zahtjeva građana

UC33 - Registracija korisnika

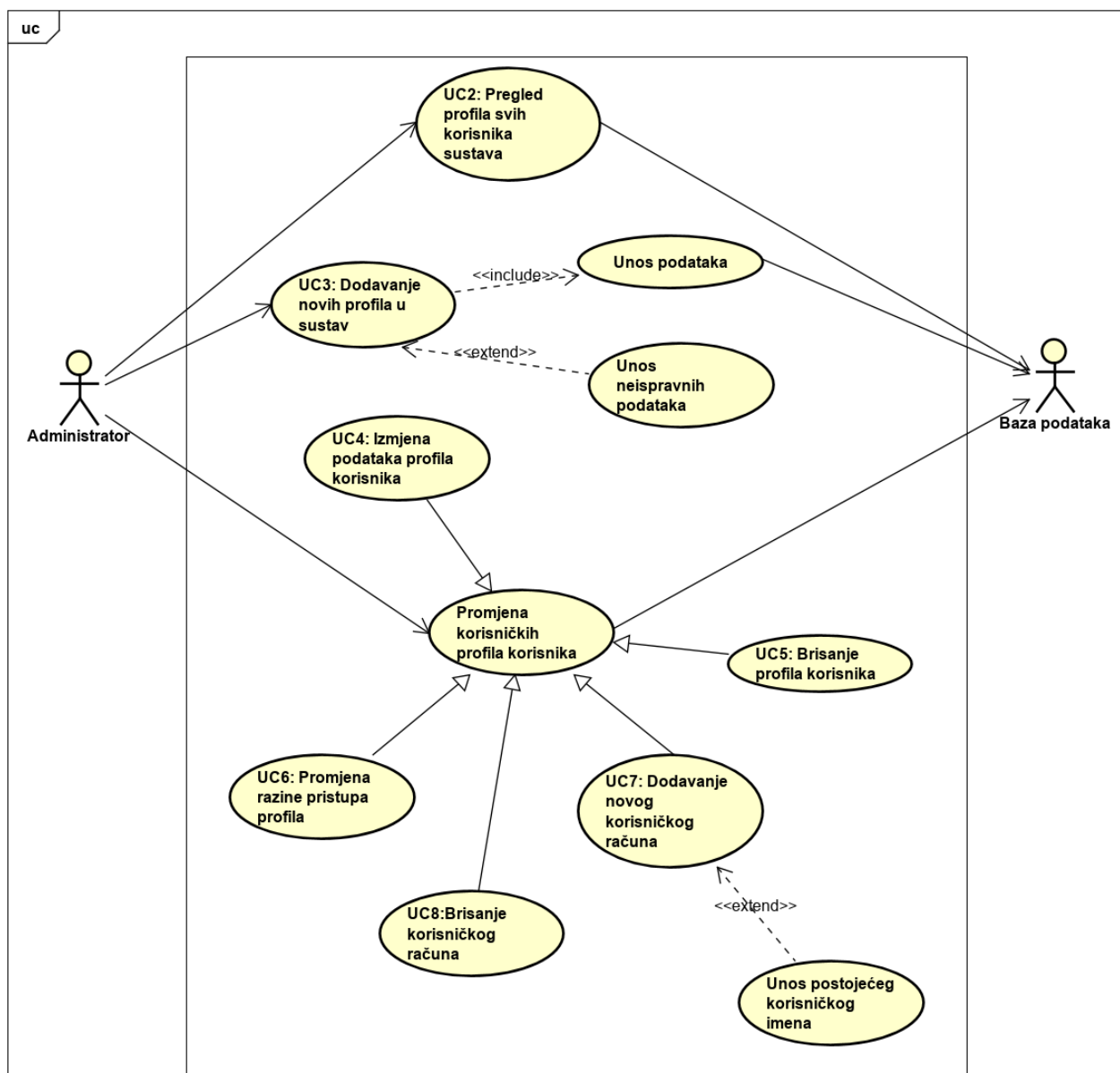
- **Glavni sudionik:** Neregistrirani klijent

- **Cilj:** Otvoriti uslugu internet bankarstva
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik ima otvoren profil u banci
- **Opis osnovnog tijeka:**
 1. Korisnik upisuje OIB i privremeni ključ dobiven od bankara u aplikaciju
 2. Korisnik unosi novo korisničko ime i novu lozinku
 3. Aplikacija javlja uspješnost otvaranja usluge internet bankarstva
- **Opis mogućih odstupanja:**
 - 1.a Unos neispravnih podataka
 1. Ispis odgovarajuće poruke upozorenja
 - 2.a Korisnik unosi korisničko ime koje već postoji
 1. Ispis odgovarajuće poruke upozorenja

Dijagrami obrazaca uporabe



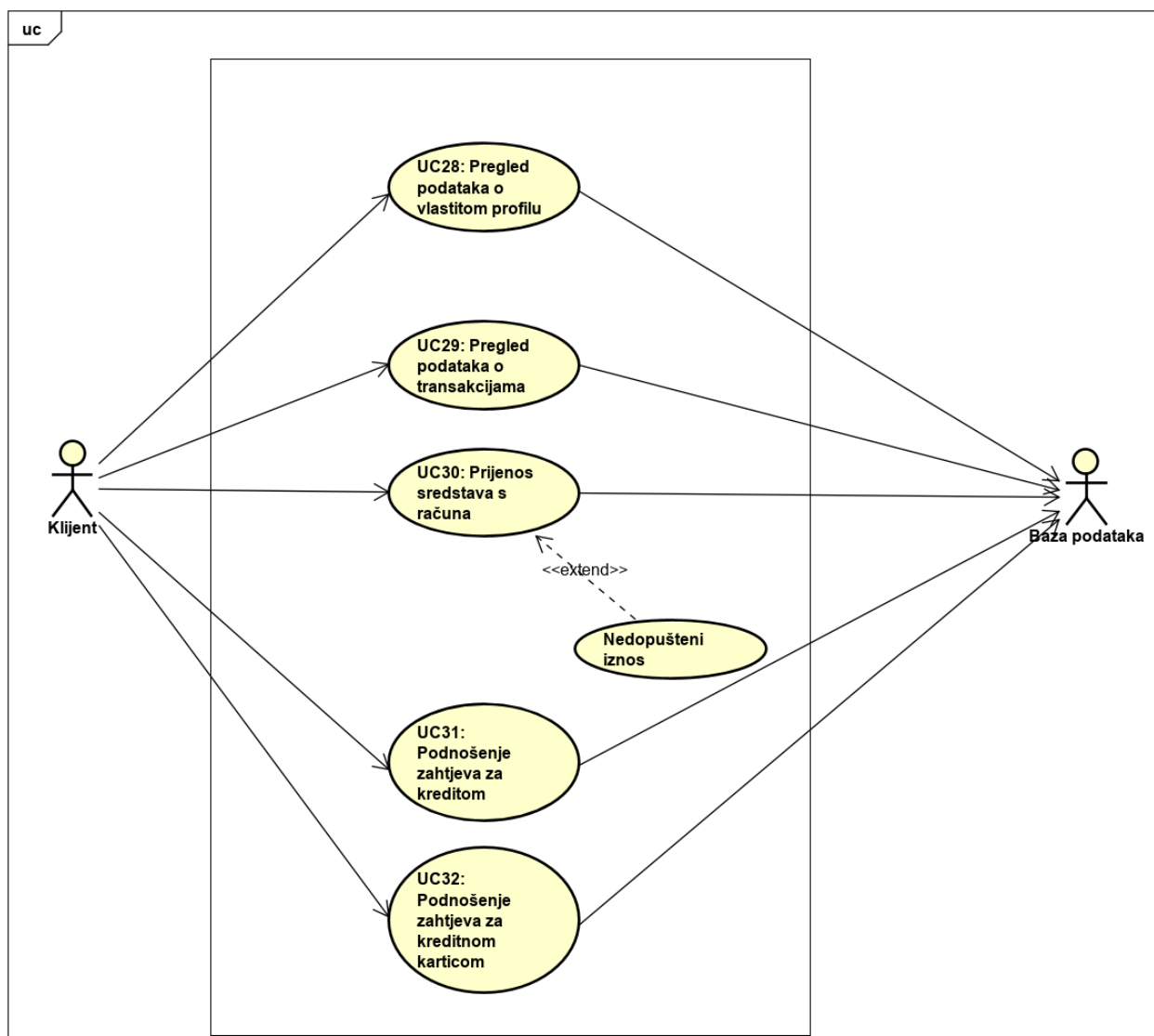
Slika 3.1: Dijagram obrasca uporabe, ponašanje svih korisnika



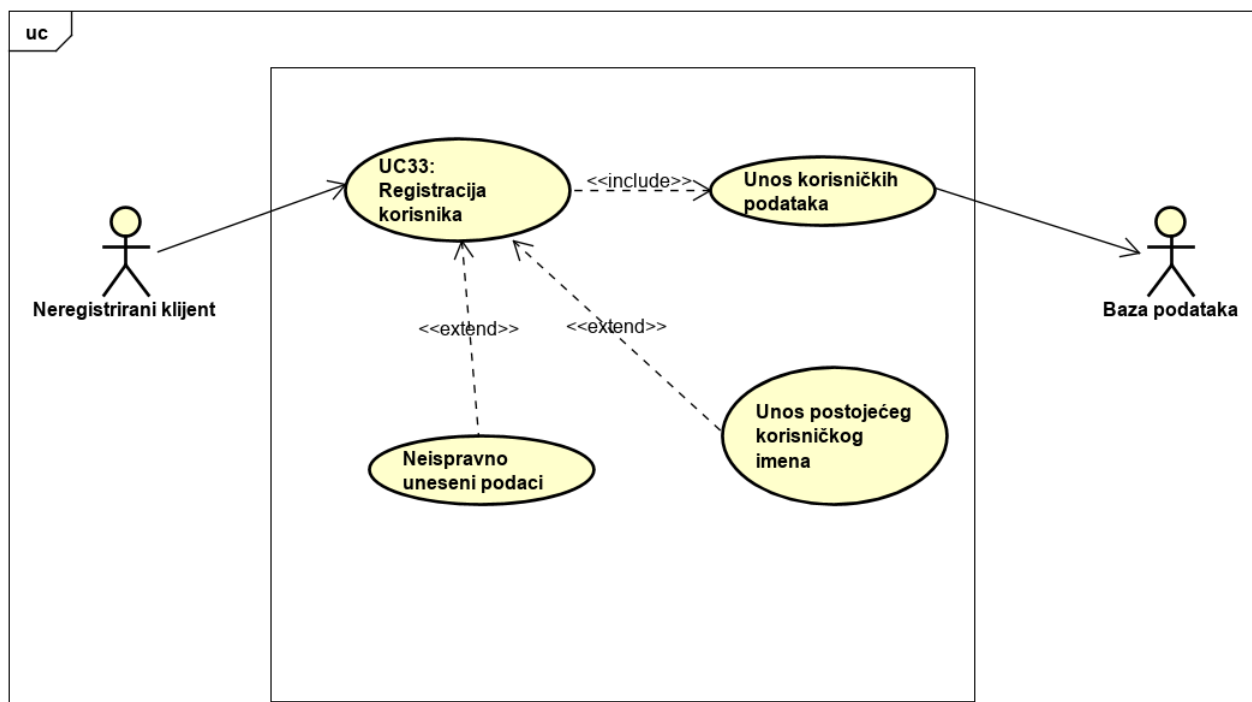
Slika 3.2: Dijagram obrasca uporabe, ponašanje administratora



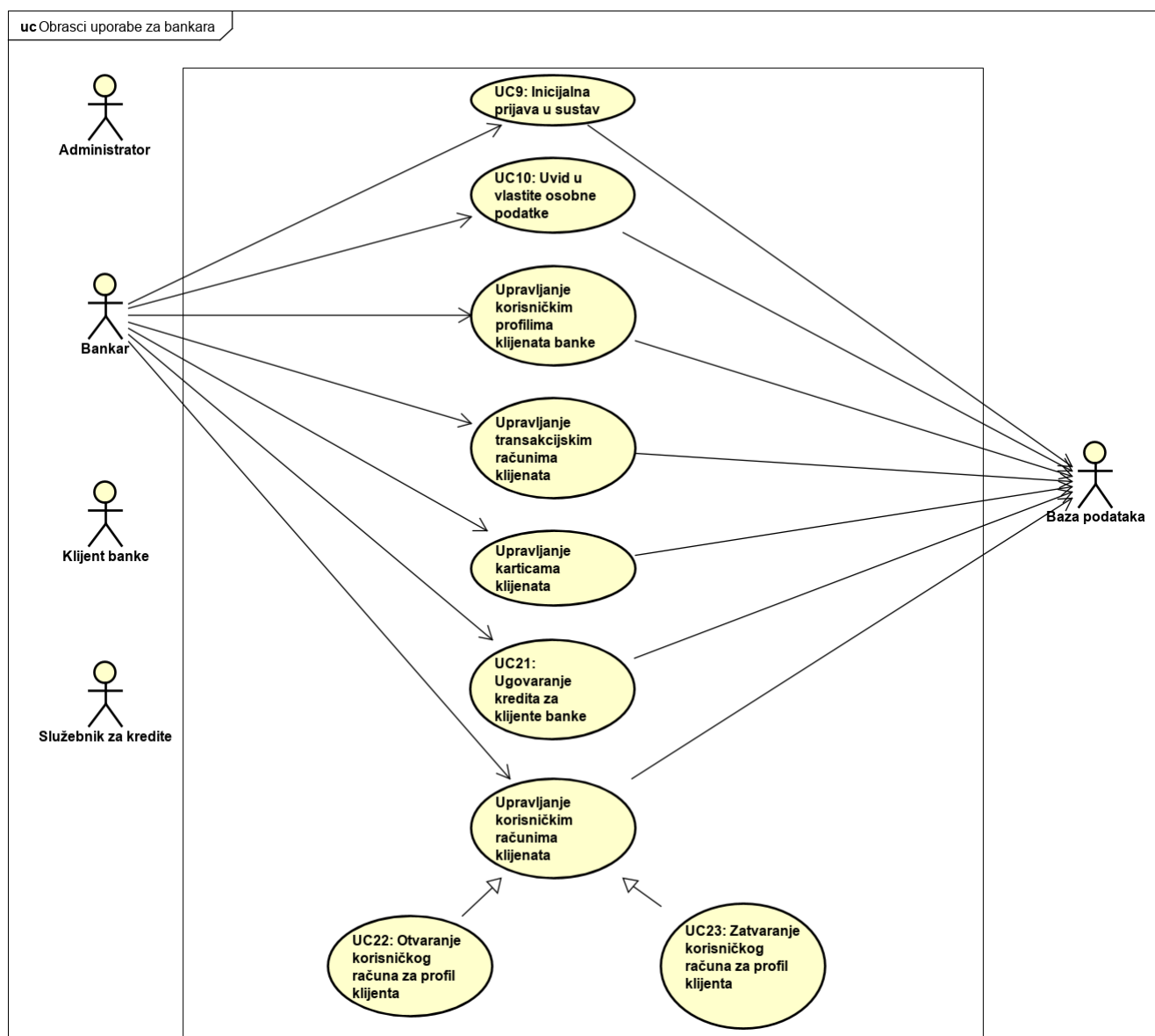
Slika 3.3: Dijagram obrasca uporabe, ponašanje službenika za odobravanje kredita



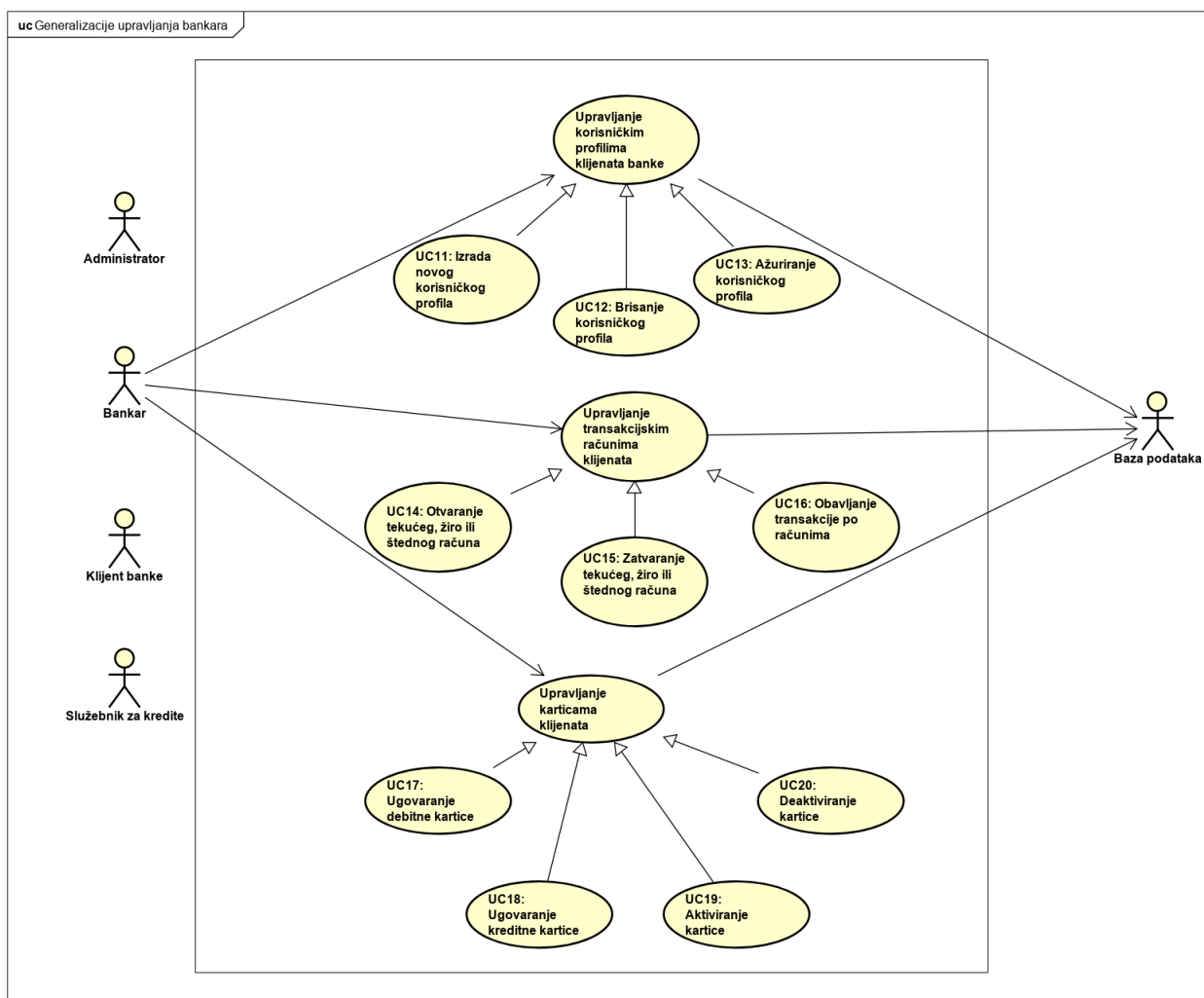
Slika 3.4: Dijagram obrasca uporabe, ponašanje klijenta



Slika 3.5: Dijagram obrasca uporabe, ponašanje neregistriranog klijenta



Slika 3.6: Dijagram obrasca uporabe, funkcionalnost bankara

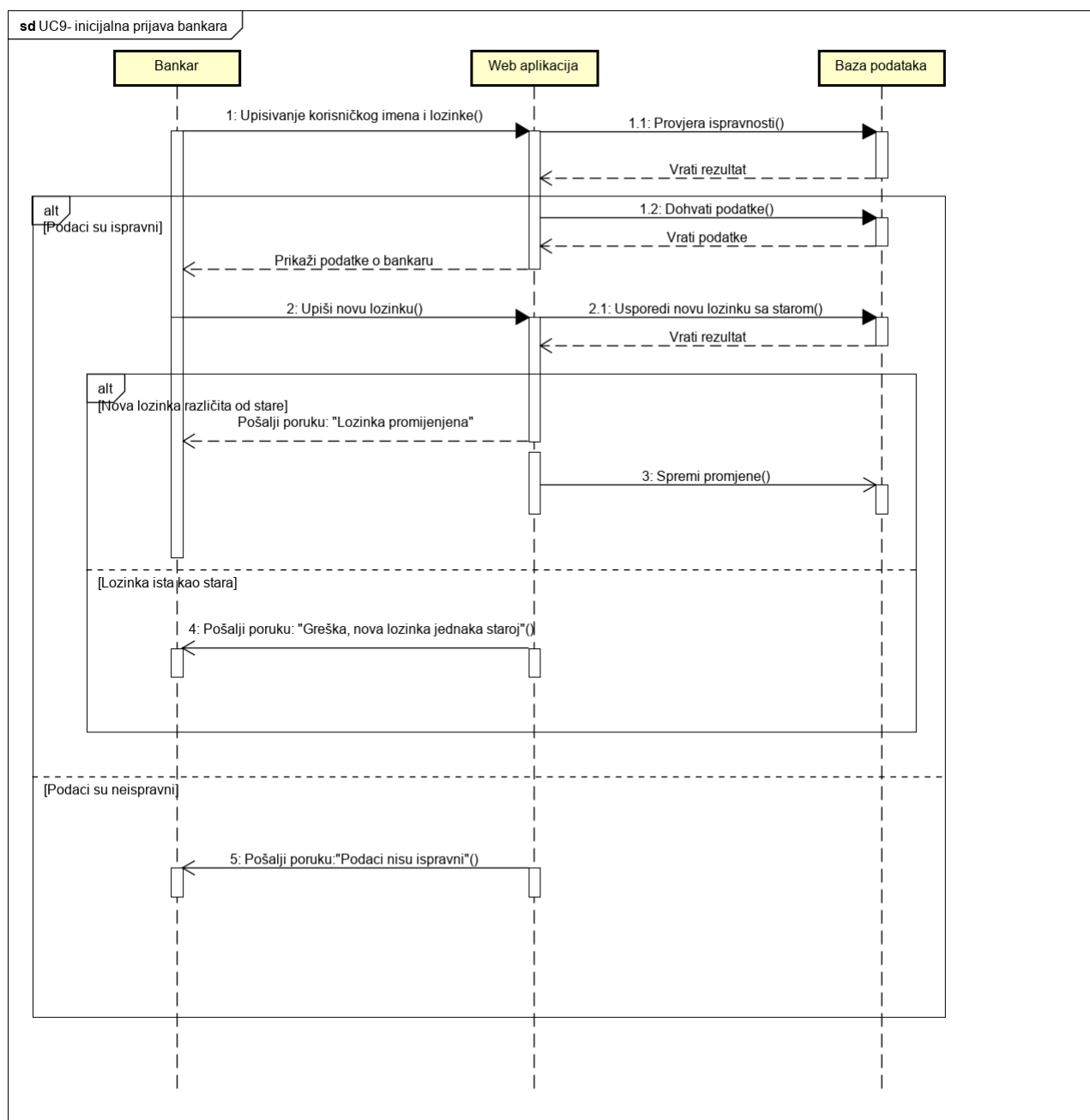


Slika 3.7: Dijagram generalizacija bankara

3.1.2 Sekvencijski dijagrami

UC9 - Inicijalna prijava bankara

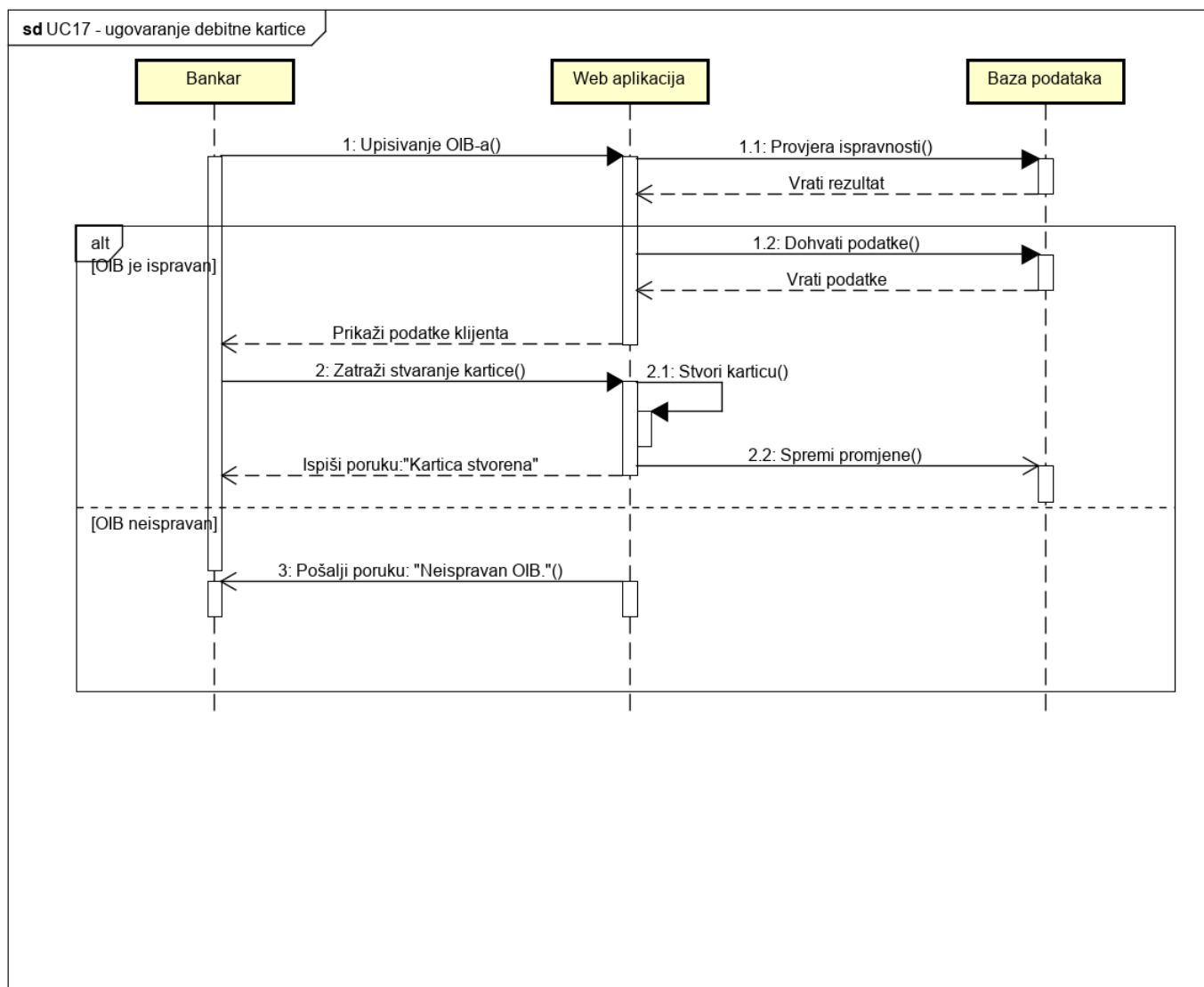
Bankar se prijavljuje u sustav pomoću korisničkog računa i privremene lozinke koju je prethodno dobio od administratora. Sustav provjerava upisane podatke, i ako su podaci točni, dohvaća podatke o bankaru iz baze podataka. Ukoliko su upisani podaci neispravni, sustav šalje poruku kojoj javlja bankaru da su uneseni podaci netočni. Bankar upisuje novu lozinku. Sustav provjerava jesu li nova i stara lozinka različite i sprema promjene u bazu podataka, a bankar je time uspješno promijenio lozinku i može početi s radom u banci. Ukoliko su nova i stara lozinka iste, sustav šalje poruku bankaru da lozinka nije promijenjena i da su te dvije lozinke iste.



Slika 3.8: Sekvencijski dijagram za UC9

UC17 - Ugovranje debitne kartice

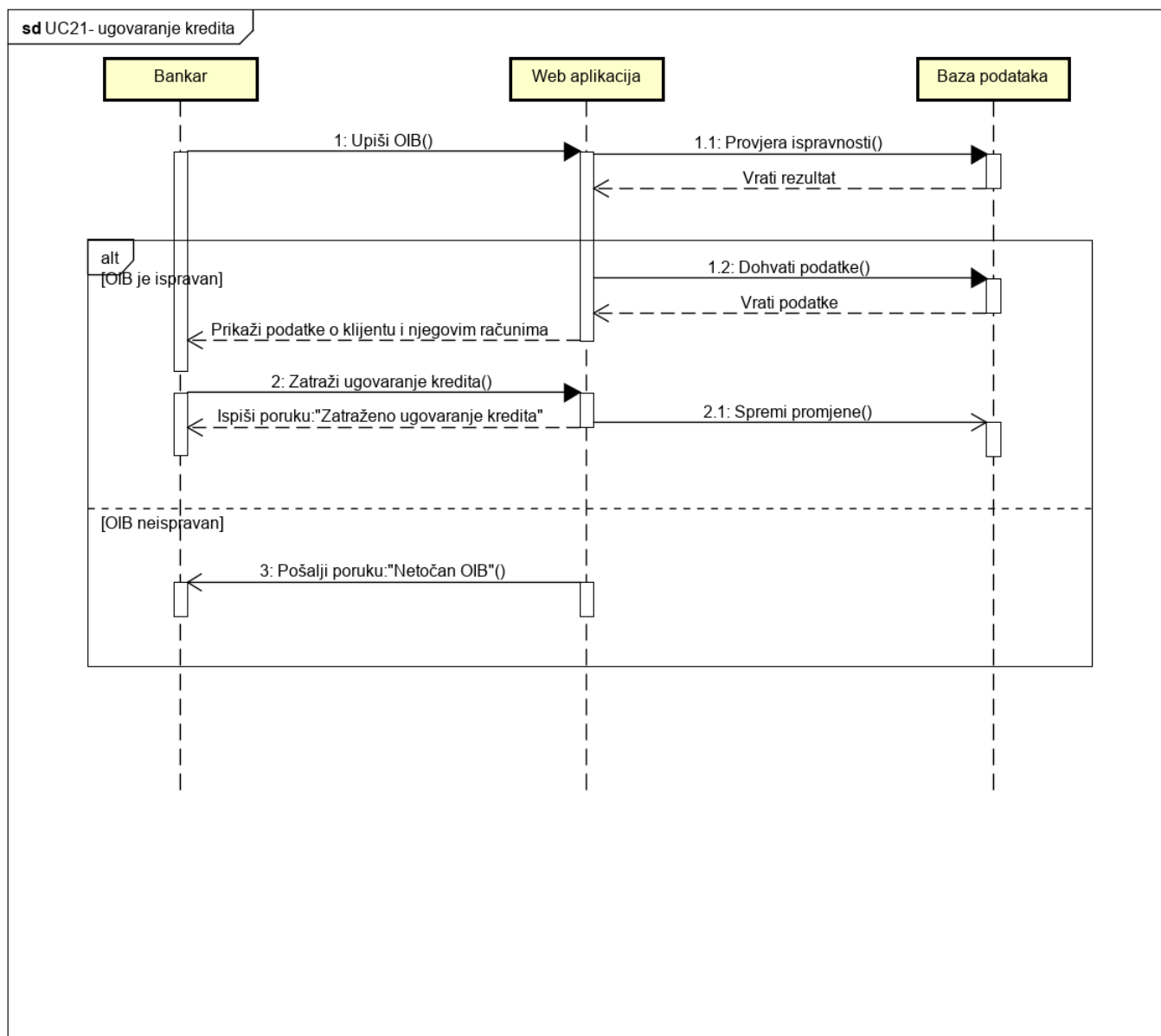
Bankar upisuje klijentov OIB. Sustav provjerava ispravnost OIB-a u bazi podataka. Baza podataka vraća rezultate provjere. Ako je upisan OIB ispravan i postoji u bazi podataka, sustav dohvaća podatke o klijentu iz baze podataka i vraća ih bankaru na uvid. Bankar zatraži stvaranje kartice, sustav stvara karticu i promjene sprema u bazu podataka. Sustav šalje poruku bankaru da je kartica uspješno stvorena. Ukoliko je upisan OIB na početku bio neispravan i baza podataka je vratila negativne rezultate, sustav šalje poruku bankaru kako je upisan OIB neispravan.



Slika 3.9: Sekvencijski dijagram za UC17

UC21 - Ugovaranje kredita

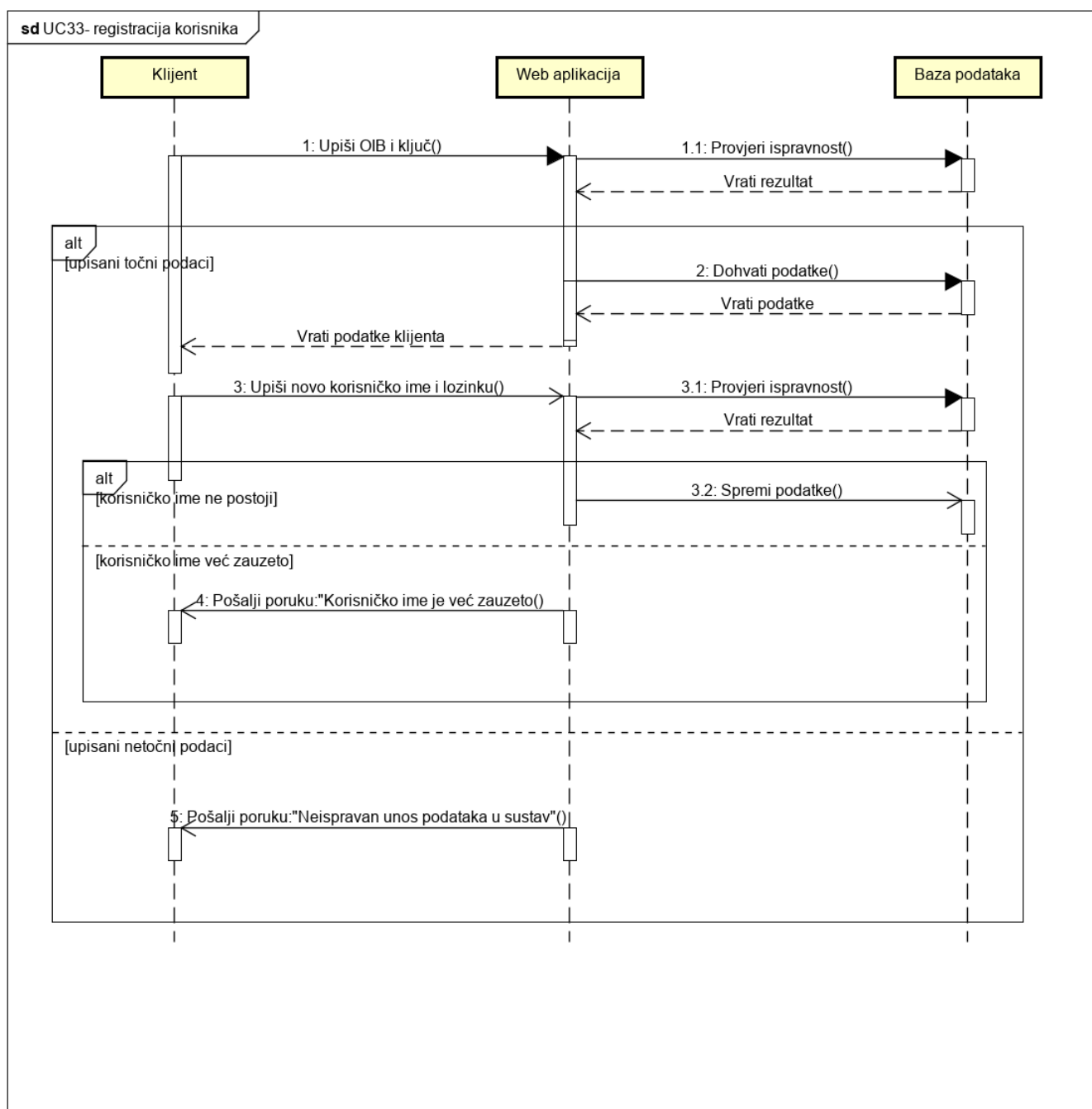
Bankar upisuje OIB u sustav. Web aplikacija provjerava ispravnost OIB-a s bazom podataka te baza podataka vraća rezultat. Ukoliko je OIB ispravan, sustav traži od baze podataka dohvaćanje klijentovih podataka. Baza podataka vraća podatke i sustav prikazuje bankaru podatke o klijentu. Bankar zatraži ugovaranje kredita, promjene se spremaju u bazu podataka i sustav šalje poruku da je zatraženo ugovaranje kredita. Ukoliko je OIB bio neispravan, bankar dobiva poruku od sustava da je OIB pogrešan.



Slika 3.10: Sekvencijski dijagram za UC21

UC33 - Registracija korisnika

Kako bi se neregistrirani korisnik mogao registrirati, u sustav upisuje OIB i ključ koji je prethodno dobio od bankara. Sustav provjerava upisane podatke u bazi podataka te baza vraća rezulta. Ako su upisani podaci točni, sustav dohvaća podatke iz baze podataka i prikazuje iz klijentu. Klijent upisuje novo korisničko ime i lozinku te sustav opet provjerava zauzetost korisničkog imena u bazi podataka. Ukoliko je baza podataka vratila pozitivan rezultat, sustav sprema promjene u bazu podataka, a u slučaju negativnog rezultata baze podataka, sustav ispisuje poruku kojoj govori da je korisničko ime već zauzeto.



Slika 3.11: Sekvencijski dijagram za UC33

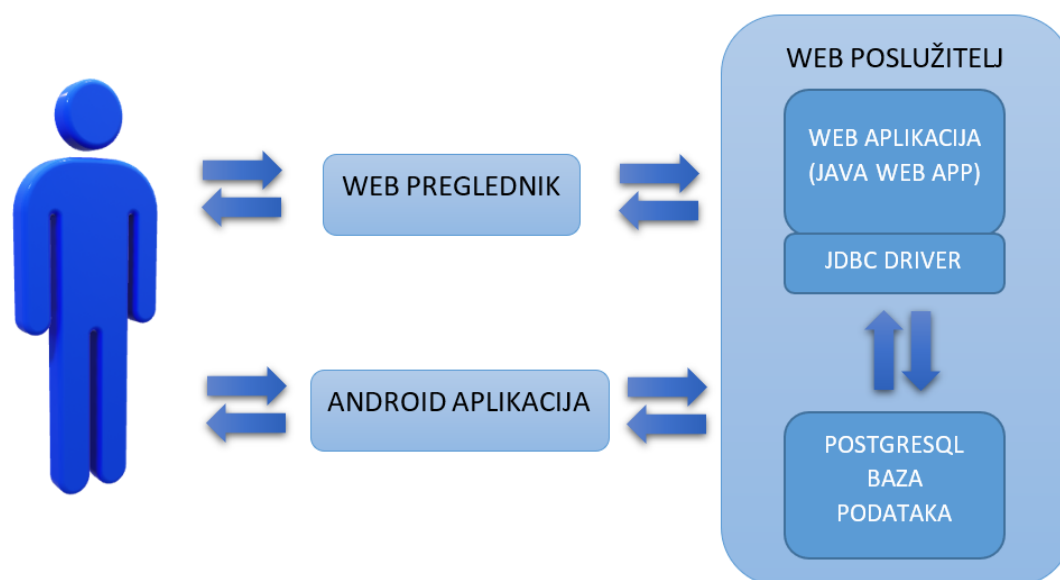
3.2 Ostali zahtjevi

- Sustav mora podržavati rad više korisnika u stvarnom vremenu
- Aplikacija mora podržavati hrvatsku abecedu (dijakritičke znakove)
- Izvršavanje dijela programa u kojem se pristupa bazi podataka ne smije trajati duže od nekoliko sekundi
- Veza s bazom podataka mora biti kvalitetno zaštićena, brza i otporna na vanjske greške
- Nepravilno korištenje korisničkog sučelja ne smije narušiti funkcionalnost i rad sustava
- Sustav treba biti jednostavan za upotrebu kako bi ga korisnici mogli koristiti bez opširnih uputa
- Nadogradnja sustava ne smije narušavati postojeće funkcionalnosti sustava
- Korisnički podaci jednog klijenta ne smiju biti vidljivi ostalim klijentima
- Klijent ne može mijenjati osobne podatke svog profila niti podatke o računima i karticama
- Svi računi moraju imati sliku profila
- Kao valuta se koristi HRK
- Jednom mjesečno korisnik treba dobiti izvod po svim transakcijskim računima i kreditnim karticama u PDF i XLS obliku
- Sustav automatski jednom mjesečno skida klijentima ratu za kredit s tekućeg računa te zaračunava unaprijed određenu kamatu na neotplaćeni dio duga po kreditnoj kartici

4. Arhitektura i dizajn sustava

Arhitektura aplikacije može se podijeliti na četiri podsustava:

- Web poslužitelj
- Web aplikacija
- Baza podataka
- Mobilna aplikacija



Slika 4.1: Arhitektura sustava s prikazom podsustava i toka podataka

Web poslužitelj osnova je web aplikacije. To je računalo koje preko internetske mreže uspostavlja komunikaciju s korisnicima sustava. U našem slučaju, to je Windows Server virtualna mašina podignuta preko servisa Microsoft Azure. Na njemu su instalirani svi servisi koji su potpora radu web aplikacije i baze podataka.

Web aplikacija mozak je našeg sustava. U njoj je opisana sva logika aplikacije. Razgovara sa korisnikom sustava putem HTTP (engl. *Hyper Text Transfer Protocol*) protokola, obrađuje zahtjeve, spaja se s bazom podataka te šalje odgovor korisniku na web preglednik ili Android aplikaciju. Odgovor je oblika HTML dokumenta

u slučaju korištenja web preglednika ili oblika JSON (JavaScript Object Notation) dokumenta u slučaju spajanja Android aplikacije na RESTful API sustava.

Za backend sustava odabran je programski jezik Java te tehnologije Java Servleti i JSP (Java Server Pages). Backend je pisan kao Maven projekt radi lakšeg upravljanja ovisnostima naše aplikacije. Kao razvojno okruženje koristi se IDE Eclipse, a Apache Tomcat kao Servlet container.

Oblikovnim obrascem MVC (Model-View-Controller) ostvarili smo razdvajanje frontend i backend dijela sustava te njihovo nezavisno i paralelno razvijanje. Posljedica toga jest jednostavno testiranje i utvrđivanje grešaka sustava te jednostavna buduća nadogradnja.

Obrazac MVC sastoji se od tri komponente:

- **Model** - strukture podataka koje služe preslikavanju podataka iz baze u objektno orijentiran jezik. Omogućuju lakši dohvat i lakše upravljanje podacima spremljenim u bazu.
- **View** - komponenta koja služi prikazu podataka iz modela na korisniku razumljiv i jednostavan način. U slučaju naše web aplikacije, za to je zadužena tehnologija JSP.
- **Controller** - komponenta koja služi upravljanju (sprema i dohvaća podataka iz baze, izvršava operacija nad podacima i cijelim sustavom, priprema podatke za prikaz i dr.)

Odabrali smo relacijski model *baze podataka* zbog njegove raširenosti, jednostavnosti i skalabilnosti. Bazu podataka opisali smo jezikom SQL i pokrenuli ju na sustavu PostgreSQL.

Mobilna aplikacija još jedan je pogled korisnika u naš sustav. Ona je izgrađena u programskom jeziku Java koristeći Android studio kao IDE. Mobilna aplikacija spaja se na RESTful dio web poslužitelja preko kojeg vrši prijavu u sustav i dohvaćanje podataka. Za izradu dizajna Android aplikacije koristi se sustav Adobe XD i Google Material Design kao vizualni jezik.

4.1 Baza podataka

Za potrebe našeg sustava koristit ćemo SQL relacijsku bazu podataka. Relacijska baza podataka najviše nam odgovara zbog olakšanog modeliranja događaja i entiteta iz stvarnog svijeta. Osnovna jedinka baze podataka je relacija, to jest tablica koja ima svoj naziv i potreban skup atributa. Zadaća baze podataka je brza i jednostavna pohrana, izmjena i dohvat podataka koje potom treba obraditi. Sve su relacije u bazi svedene na treću normalnu formu kako baza ne bi sadržavala redundantne podatke. Prilikom izrade baze podataka poslužili smo se PostgreSQL-om. Baza podataka ove aplikacije sastoji se od sljedećih entiteta:

- Profil
- Korisnički račun
- Registracija klijenta
- Razina ovlasti
- Kredit
- Vrsta kredita
- Zahtjev kredit
- Transakcija
- Račun
- Vrsta računa
- Kartica
- Vrsta kartice
- Zahtjev kartica
- Mjesto
- Županija

4.1.1 Opis tablica

Profil Ovaj entitet sadrži sve važne informacije za pristup web aplikaciji. Sadrži attribute: ime, prezime, OIB, adresa prebivališta, poštanski broj, datum rođenja, e-mail adresa i slika profila. Ovaj entitet u vezi je *Zero-to-Many* s entitetom Korisnički račun preko OIB-a i u vezi je *Many-to-One* sa entitetom Mjesto preko poštanskog broja. Također je i u vezi sa *One-to-One* sa entitetom Registracija klijenta preko OIB-a.

Profil		
Ime	VARCHAR	ime korisnika
Prezime	VARCHAR	prezime korisnika
OIB	VARCHAR	OIB korisnika
Adresa prebivališta	VARCHAR	adresa korisnika
Poštanski broj	INT	poštanski broj (mjesto.poštanskiBroj)
Datum rođenja	DATE	datum rođenja korisnika
Email	VARCHAR	e-mail adresa korisnika
Slika	VARCHAR	slika korisnika

Korisnički Račun Ovaj entitet sadrži sve važne informacije o korisničkom računu. Sadrži attribute: OIB, korisničko ime, lozinka, razina ovlasti i promjena lozinke. Ovaj entitet je u vezi *Many-to-Zero* s entitetom Profil preko OIB-a i s entitetom *One-to-One* Razina ovlasti preko razine ovlasti.

Korisnički račun		
Korisničko ime	VARCHAR	jedinstveni identifikator korisnika
Lozinka	VARCHAR	hash lozinke
OIB	VARCHAR	OIB korisnika (profil.OIB)
Razina ovlasti	INT	broj razine ovlasti (razinaOvlasti.razinaOvlasti)
Promjena lozinske	BOOLEAN	treba li promijeniti lozinku prilikom prijave

Registracija klijenta Ovaj entitet sadrži sve važne informacije o registraciji klijenta. Sadrži attribute: OIB i privremeni ključ. Ovaj entitet je u vezi *One-to-One* s entitetom Profil preko OIB-a.

Registracija klijenta		
OIB	VARCHAR	OIB korisnika (profil.OIB)
Privremeni ključ	VARCHAR	privremeni ključ prilikom otvaranja računa

Razina ovlasti Ovaj entitet sadrži informacije vezane za razinu ovlasti. Sadrži attribute: razina ovlasti i naziv. Ovaj je entitet u vezi *One-to-One* s entitetom Korisnički račun preko razine ovlasti.

Razina ovlasti		
Razina ovlasti	INT	Broj razine ovlasti
Naziv	VARCHAR	Naziv ovlasti

Mjesto Ovaj entitet sadrži sve važne informacije o mjestu. Sadrži attribute: poštanski broj, naziv mjesta i šifra županije. U vezi je *One-to-Many* s entitetom Profil preko poštanskog broja i u vezi *Many-to-One* s entitetom Županija preko šifre županije.

Mjesto		
Poštanski broj	INT	poštanski broj mjesta
Naziv mjesta	VARCHAR	naziv mjesta
Šifra županije	INT	šifra županije(županija.šifraŽupanije)

Županija Ovaj entitet sadrži sve važne informacije o županiji. Sadrži attribute: šifra županije i naziv županije. U vezi u vezi *One-to-Many* s entitetom Mjesto preko šifre županije.

Županija		
Šifra županije	INT	šifra županije
Naziv županije	VARCHAR	naziv županije

Kredit Ovaj entitet sadrži sve važne informacije o kreditu koji klijent zatraži. Sadrži attribute: broj kredita, OIB, iznos, vrsta kredita, datum ugovaranja, vremenski period otplate kredita, datum rate i preostalo dugovanje. U vezi je *One-to-One* s entitetom Vrsta Kredita preko vrste kredita i u vezi *Many-to-One* s entitetom Profil preko OIB-a.

Kredit		
Broj kredita	INT	broj kredita
OIB	VARCHAR	OIB korisnika (profil.OIB)
Iznos	DECIMAL (10, 2)	iznos kredita
Vrsta	INT	vrsta kredita (vrstaKredita.tipKredita)
Datum ugovaranja	DATE	datum ugovaranja kredita
Vremenski period otplate	INT	duljina otplate kredita u godinama
Datum rate	INT	datum plaćanja mjesečne rate
Preostalo dugovanje	DECIMAL(10,2)	preostali iznos dugovanja

Vrsta kredita Ovaj entitet sadrži sve važne informacije o vrsti kredita. Sadrži attribute: vrsta kredita, naziv vrste kredita i kamatnu stopu. U vezi je *One-to-One* sa entitetom Kredit preko vrste kredita i u vezi *One-to-Many* sa entitetom Zahtjev kredita preko vrste kredita.

Vrsta kredita		
Vrsta kredita	INT	broj vrste kredita
Naziv vrste kredita	VARCHAR	vrsta kredita
Kamatna stopa	DECIMAL (3,2)	iznos kamatne stope

Zahtjev kredita Ovaj entitet sadrži sve važne informacije o zahtjevu kredita. Sadrži attribute: šifra zahtjeva, OIB, iznos, šifra vrste kredita, period otplate i odobrenje. U vezi je *One-to-One* s entitetom Vrsta Kredita preko šifre vrste kredita i u vezi *Many-to-One* s entitetom Profil preko OIB-a.

Zahtjev kredita		
Šifra zahtjeva	INT	šifra zahtjeva kredita
<i>OIB</i>	VARCHAR	OIB korisnika (profil.OIB)
Iznos	DECIMAL (10, 2)	iznos kredita
<i>Šifra vrste kredita</i>	INT	vrsta kredita (šifraVrsteKredita.tipKredita)
Period otplate	INT	duljina otplate kredita u godinama
Odobrenje	BOOLEAN	status zahtjeva kredita

Transakcija Ovaj entitet sadrži sve važne informacije o transakcijama koje klijent želi provesti. Sadrži attribute: broj transakcije, broj računa terećenja, račun odobrenja, iznos i datum transakcije. Entitet Transakcija u vezi je *Many-to-One* s entitetom Račun preko broja računa terećenja i broja računa odobrenja.

Transakcija		
Broj transakcije	INT	broj transakcije
<i>Račun terećenja</i>	VARCHAR	broj računa terećenja(racun.brojRačuna)
<i>Račun odobrenja</i>	VARCHAR	broj računa odobrenja(racun.brojRačuna)
Iznos	DECIMAL (10,2)	iznos uplate
Datum transakcije	DATE	datum transakcije

Račun Ovaj entitet sadrži sve važne informacije o računu koji ima klijent. Sadrži attribute: broj računa, OIB klijenta, datum otvaranja računa, stanje računa, vrsta računa, prekoračenje, kamatna stopa i datum zatvaranja. U vezi je *Many-to-One* s entitetom Profil preko OIB-a, sa entitetom Vrsta Računa *One-to-One* preko vrste računa. Također je u vezi *One-to-Many* s entitetom Transakcija preko broja računa.

Račun		
Broj računa	VARCHAR	broj računa
<i>OIB</i>	VARCHAR	OIB korisnika (profil.OIB)

Račun		
Datum otvaranja	DATE	datum otvaranja računa
Stanje	DECIMAL (10, 2)	trenutno stanje računa
Vrsta računa	INT	broj vrste računa (vrstaRačuna.vrstaRačuna)
Prekoračenje	DECIMAL(10,2)	iznos prekoračenja računa
Kamatna stopa	DECIMAL(3, 2)	iznos kamatne stope
Datum zatvaranja	DATE	datum zatvaranja računa

Vrsta računa Ovaj entitet sadrži sve važne informacije o vrsti računa koje neki klijent posjeduje. Sadrži attribute: broj vrste računa i naziv računa. U vezi je *One-to-One* s entitetom Račun preko vrste računa.

Vrsta računa		
Vrsta računa	INT	Broj vrste računa
Naziv računa	VARCHAR	Naziv vrste računa

Kartica Ovaj entitet sadrži sve važne informacije o karticama koje ima klijent. Sadrži attribute: broj kartice, broj računa, OIB, broj vrste kartice, stanje, valjanost, limit, kamatna stopa i datum rate. U vezi je *Many-to-One* s entitetom Račun preko broja računa i sa entitetom Vrsta kartice *One-to-One* preko vrste kartice.

Kartica		
Broj kartice	VARCHAR	broj kartice
Broj računa	VARCHAR	broj računa za koji je kartica vezana (račun.brojRačuna)
OIB	VARCHAR	OIB klijenta (profil.OIB)
Vrsta kartice	INT	broj tipa kartice (vrstaKartice.tipKartice)
Stanje	DECIMAL(10,2)	stanje kartice
Valjanost	DATE	datum isteka valjanosti kartice
Limit	DECIMAL (10,2)	odobren limit
Kamatna stopa	DECIMAL(3, 2)	iznos kamatne stope
Datum rate	INT	datum otplate dugovanja kartice

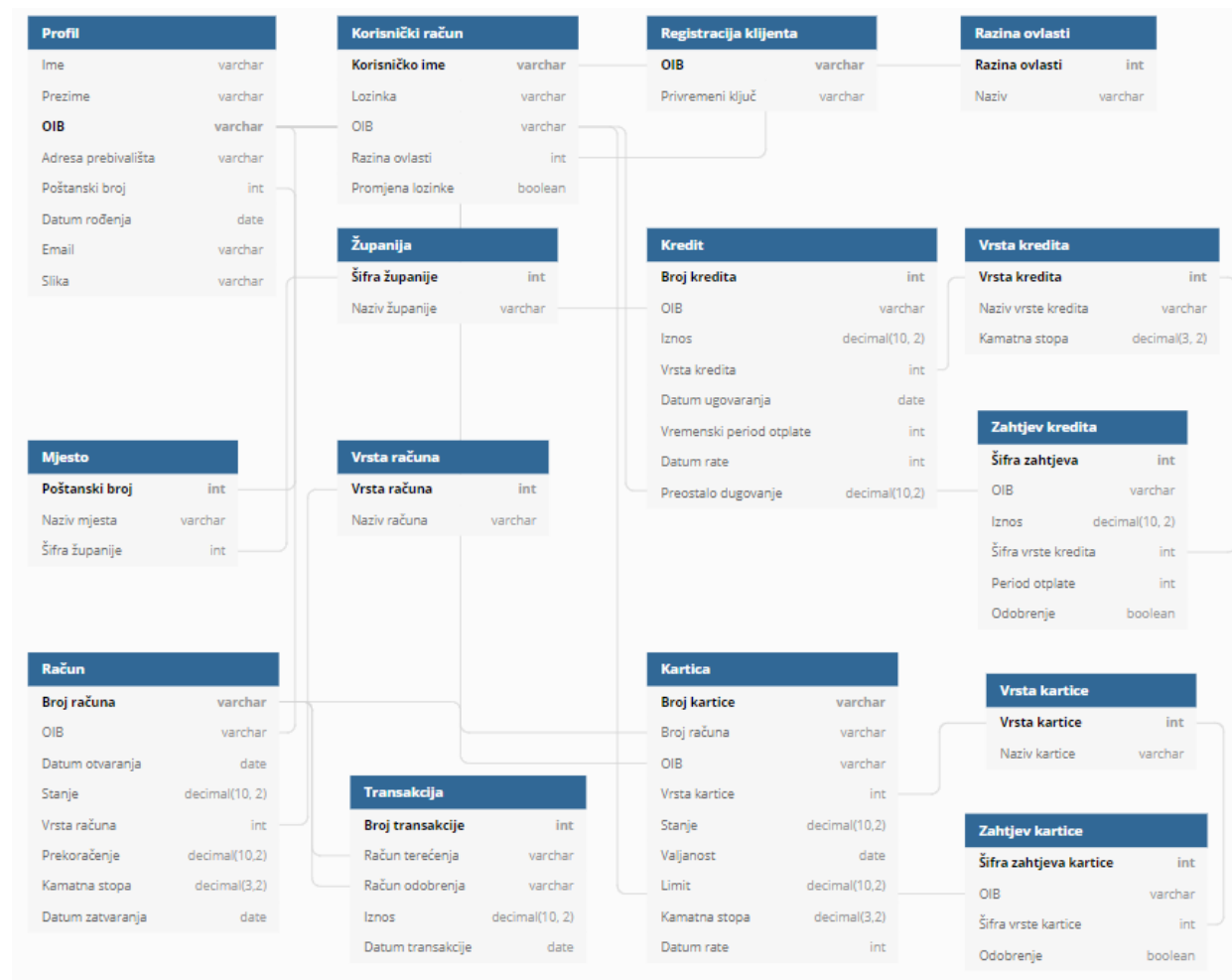
Vrsta kartice Ovaj entitet sadrži sve važne informacije o vrstama kartice koje ima klijent. Sadrži attribute: broj vrste kartice i naziv kartice. U vezi je *One-to-One* s entitetom Kartica preko vrste kartice i u vezi sa entitetom Zahtjev kartice *One-to-One* preko vrste kartice.

Vrsta kartice		
Vrsta kartice	INT	broj vrste kartice
Naziv kartice	VARCHAR	naziv kartice

Zahtjev kartice Ovaj entitet sadrži sve važne informacije o zahtjevu kartice. Sadrži attribute: šifra zahtjeva, OIB, šifra vrste kartice i odobrenje. U vezi je *Many-to-One* s entitetom Profil preko OIB-a i sa entitetom Vrsta kartice *One-to-One* preko šifre vrste kartice.

Zahtjev kartice		
Šifra zahtjeva	INT	šifra zahtjeva kartice
<i>OIB</i>	VARCHAR	OIB klijenta (profil.OIB)
<i>Šifra vrste kartice</i>	INT	broj tipa kartice (vrstaKartice.tipKartice)
Odobrenje	BOOLEAN	status zahtjeva kartice

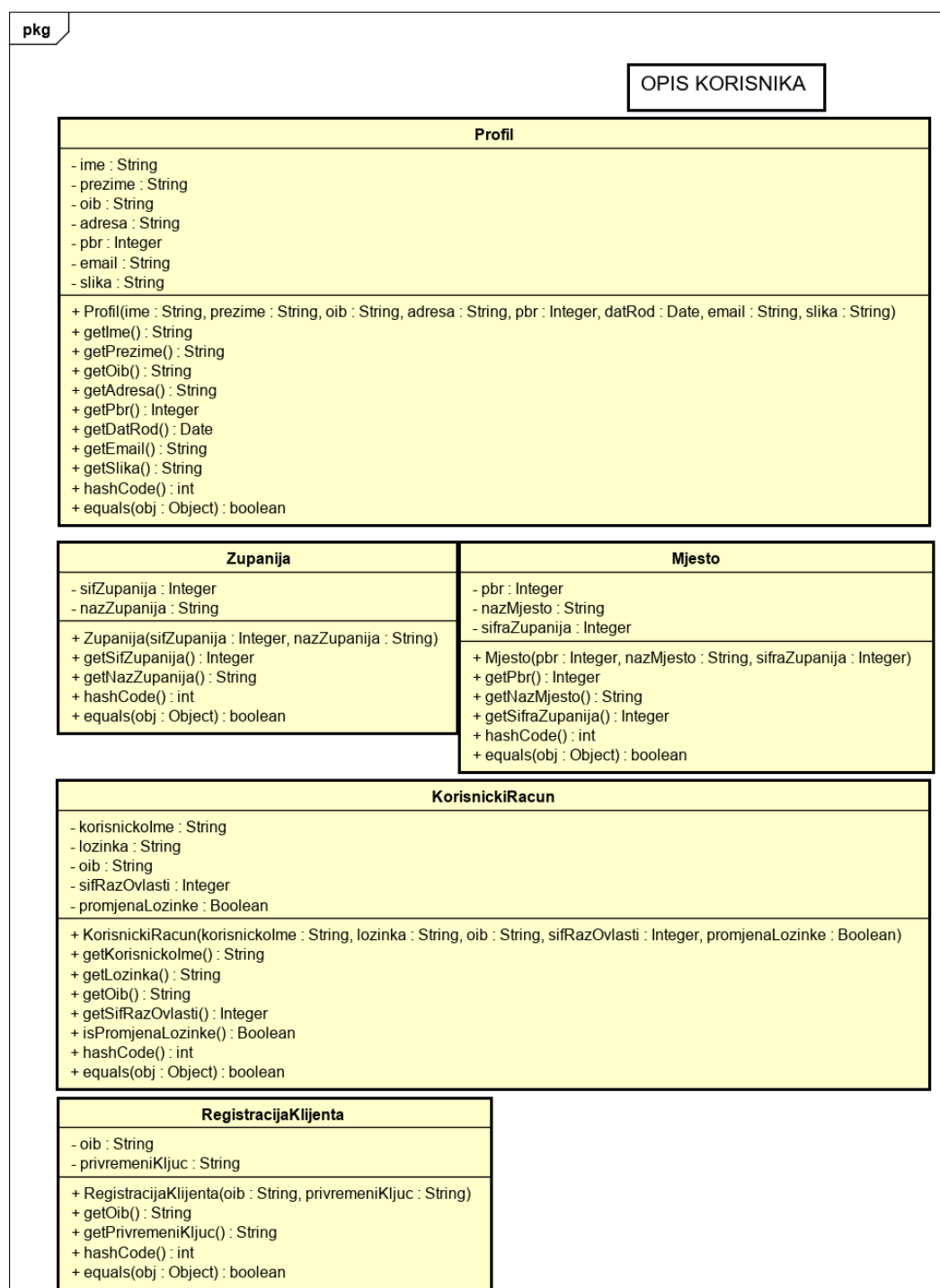
4.1.2 Dijagram baze podataka



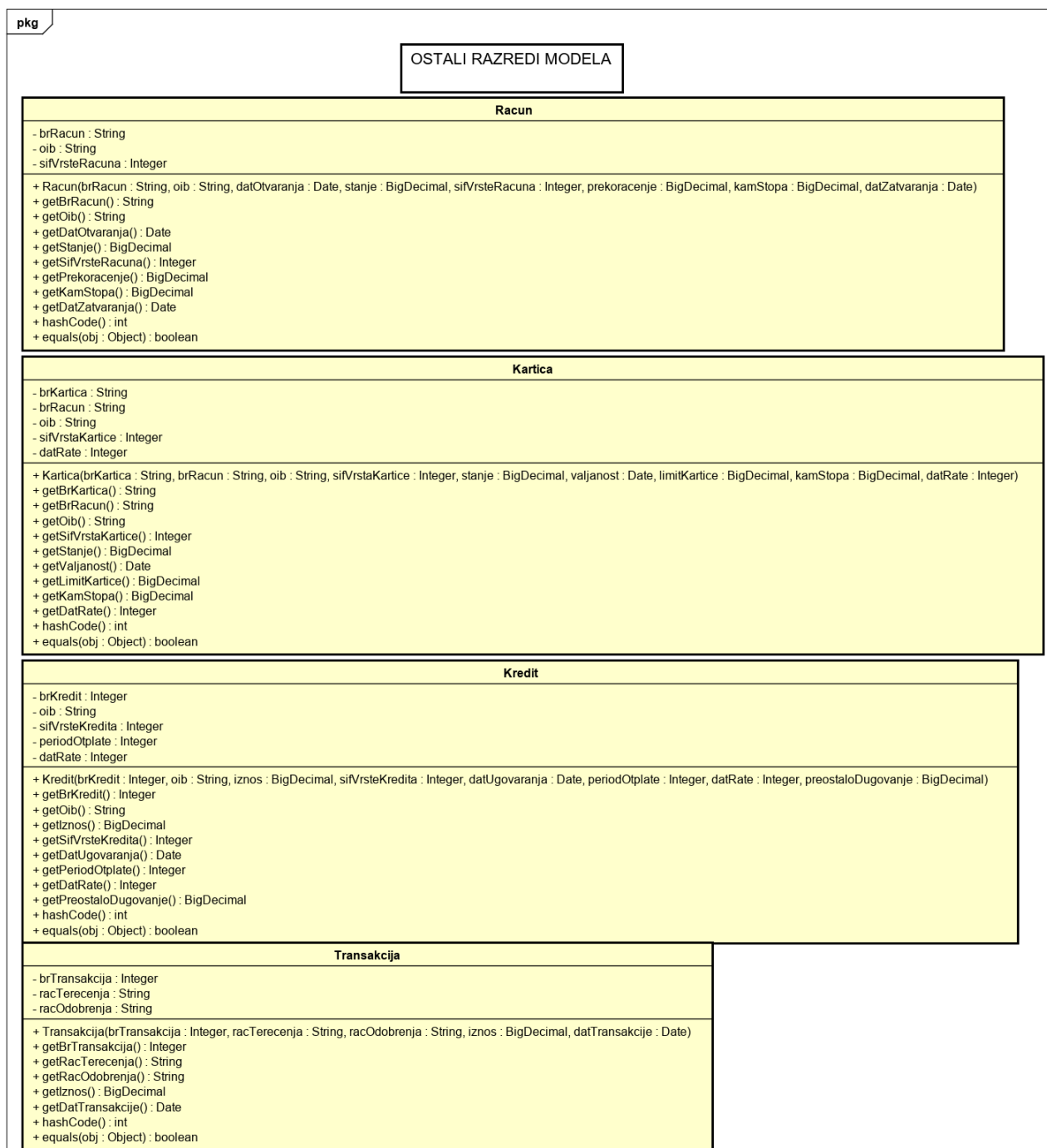
Slika 4.2: E-R dijagram baze podataka

4.2 Dijagram razreda

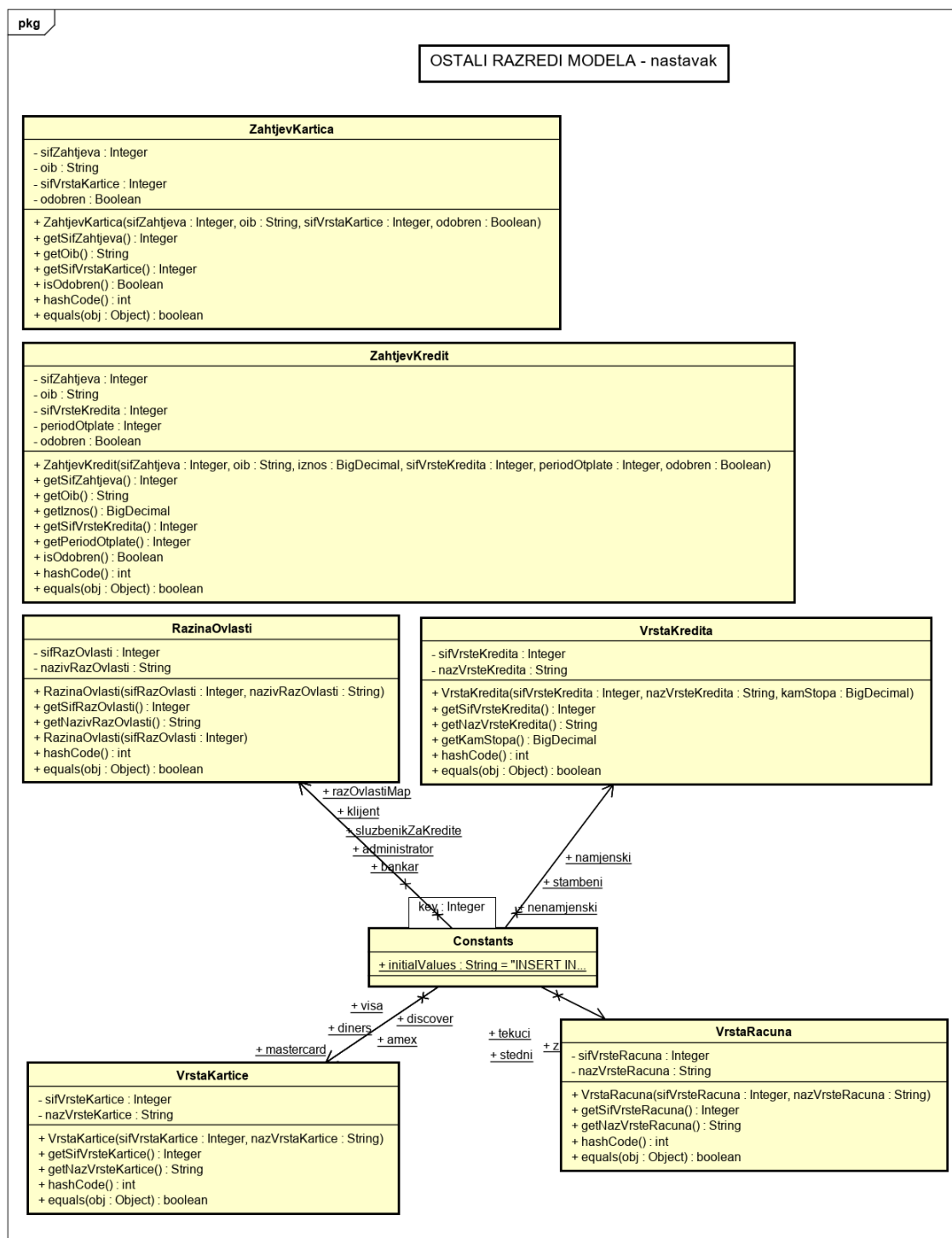
Na slikama 4.3 - 4.14 prikazani su razredi koji pripadaju backend dijelu MVC arhitekture.



Slika 4.3: Razredi modela opisa korisnika

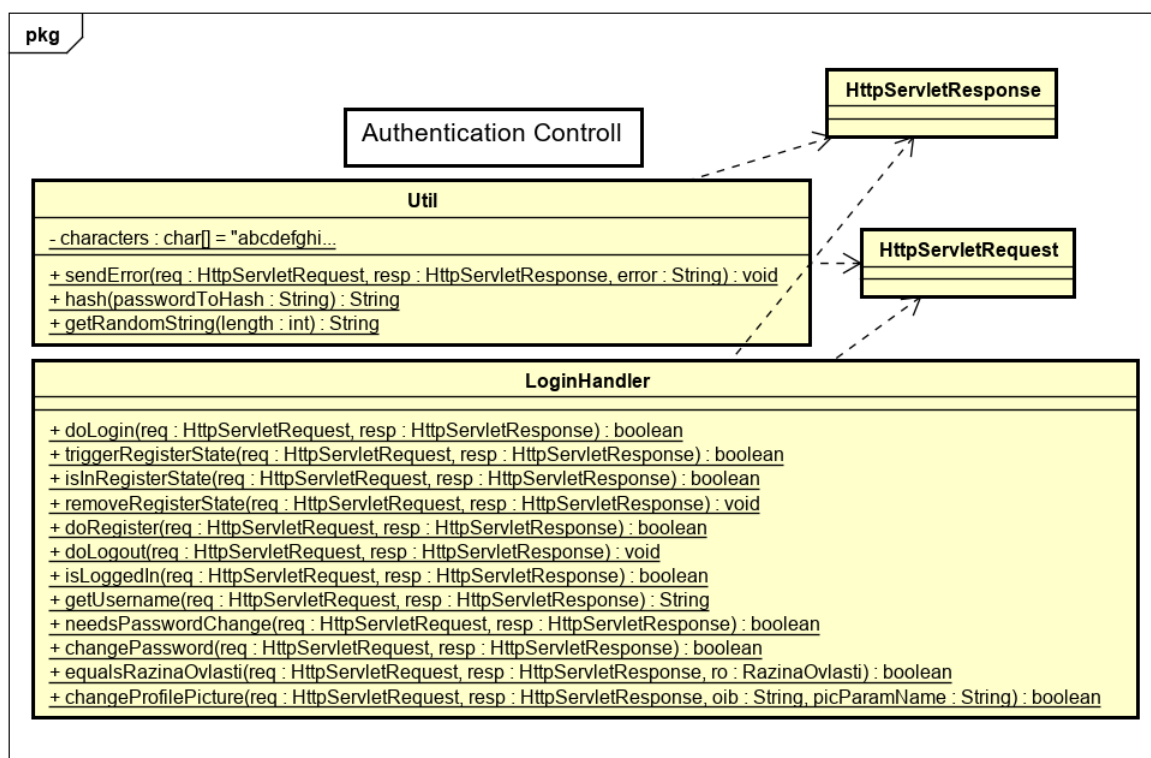


Slika 4.4: Ostali razredi modela



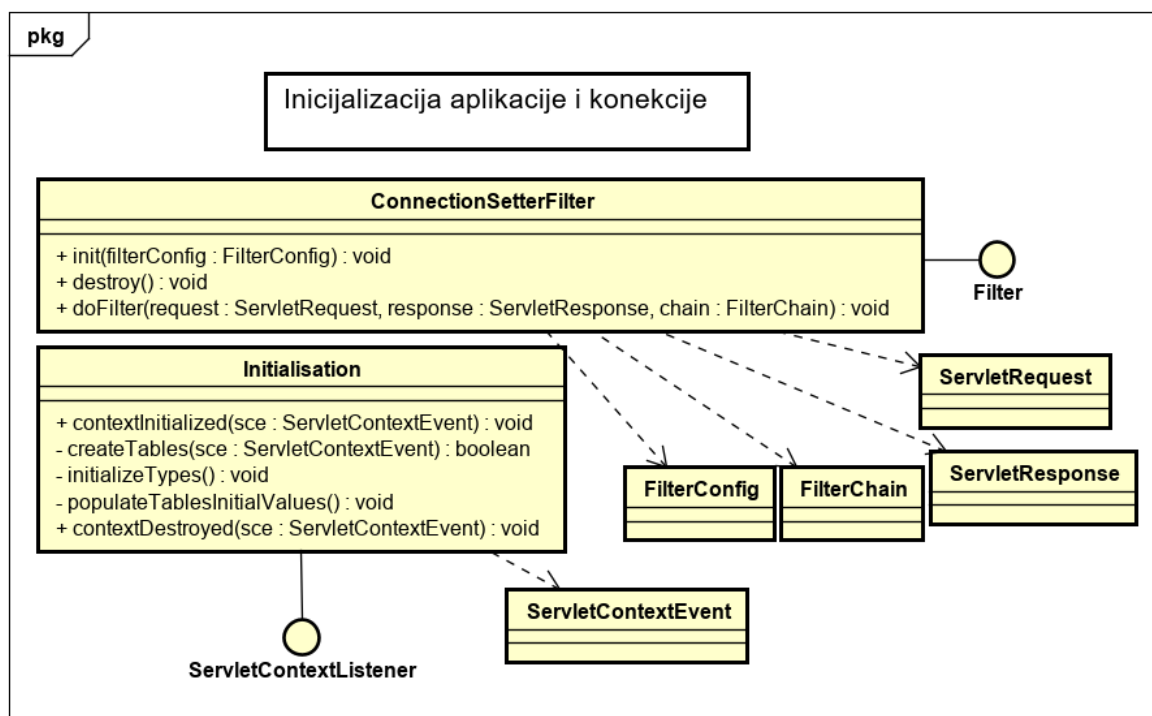
Slika 4.5: Ostali razredi modela - nastavak

Slike 4.3, 4.4 i 4.5 prikazuju strukture podataka pomoću kojih su oblikovani i prikazani podaci dohvaćeni iz baze podataka. Predstavljaju jedan-na-jedan presliku relacija baze podataka.



Slika 4.6: Razredi zaduženi za kontrolu autentifikacije

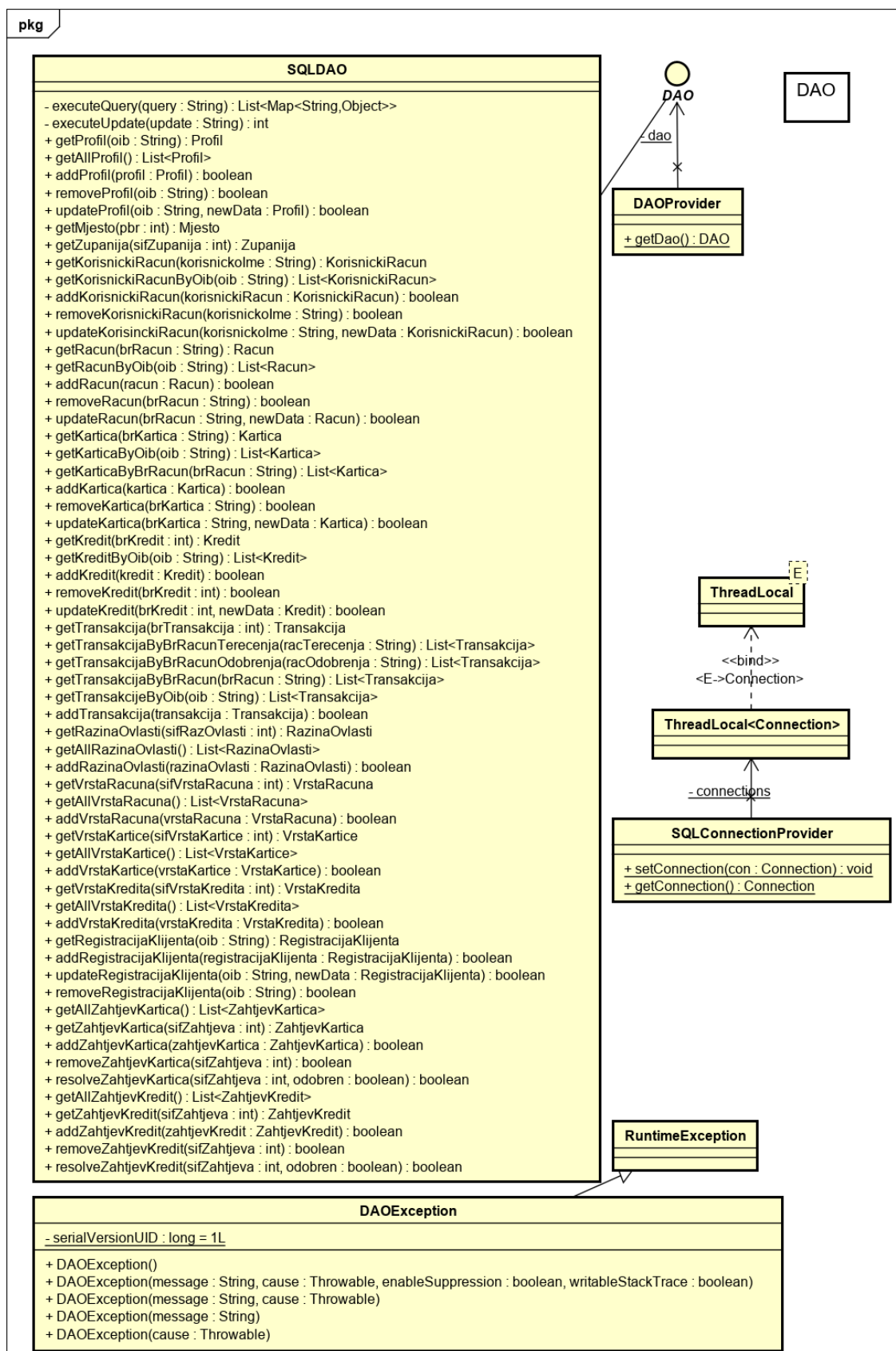
Slika 4.6 prikazuje razrede zadužene za kontrolu autentifikacije korisnika sustava. Razred LoginHandler preko DAO sloja komunicira s bazom podataka i uspoređuje vjerodajnice za prijavu i registraciju s onima spremljenim u bazu. Tu su i funkcionalnosti koje se koriste na više mjesta kroz aplikaciju objedinjene u jedan razred - Util.



Slika 4.7: Razredi zaduženi za inicijalizaciju aplikacije i konekcije

Metode razreda `ConnectionSetterFilter` izvide se prilikom svakog HTTP zahtjeva upućenog od strane korisnika sustava. Služi za postavljanje konekcije prema bazi podataka trenutnoj dretvi te za vraćanje konekcije u bazen konekcija po završetku obrade zahtjeva.

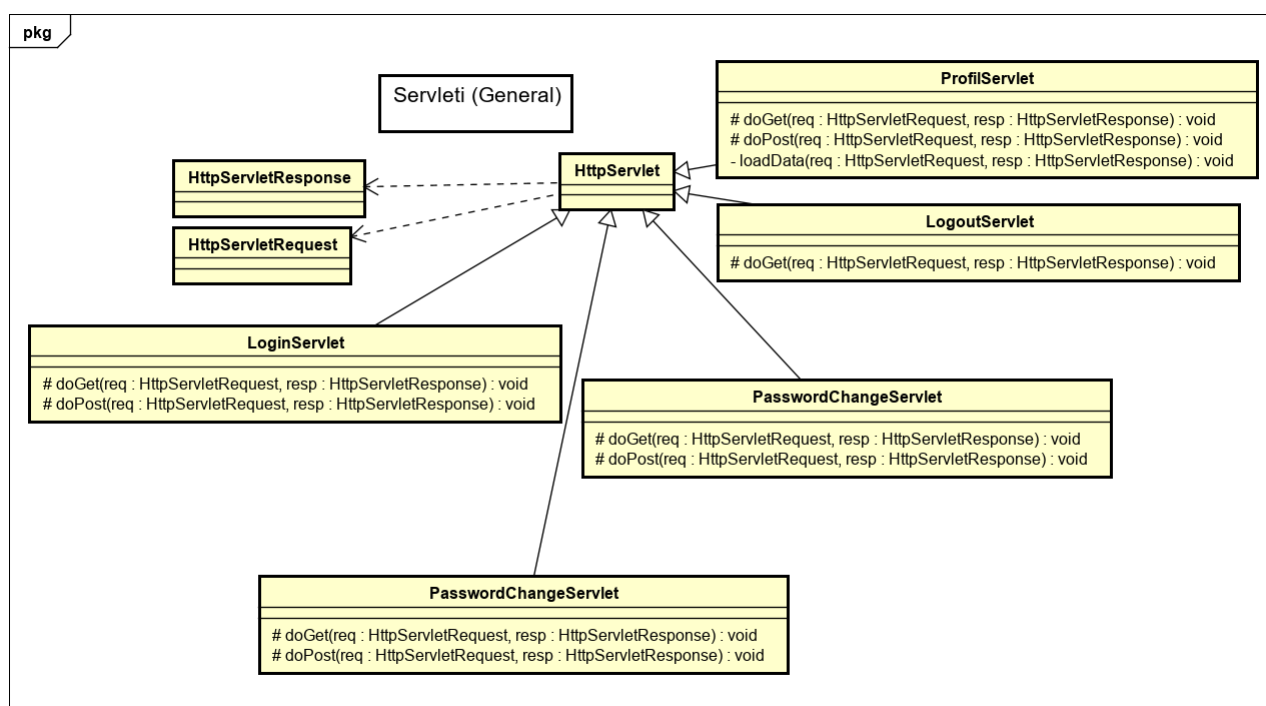
Metode razreda `Initialisation` izvide prilikom pokretanja web aplikacije. One inicijaliziraju bazen konekcija prema bazi podataka te (ukoliko je potrebno) pune relacije baze inicijalnim podacima.



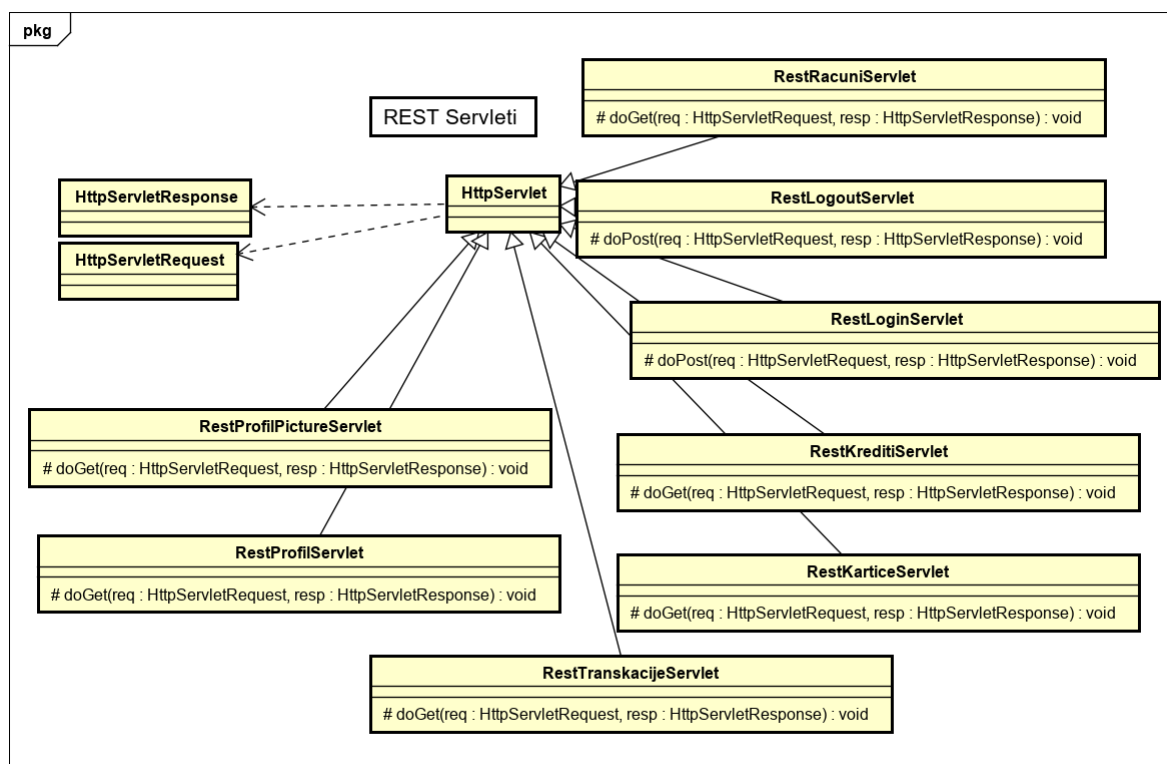
Slika 4.8: Razredi DAO sučelja

Razredi DAO sučelja oblikuju sučelje za perzistenciju podataka. Također, tu je i konkretna realizacija DAO sučelja koja podatke sprema i čita u relacijsku bazu podataka - SQLDAO.

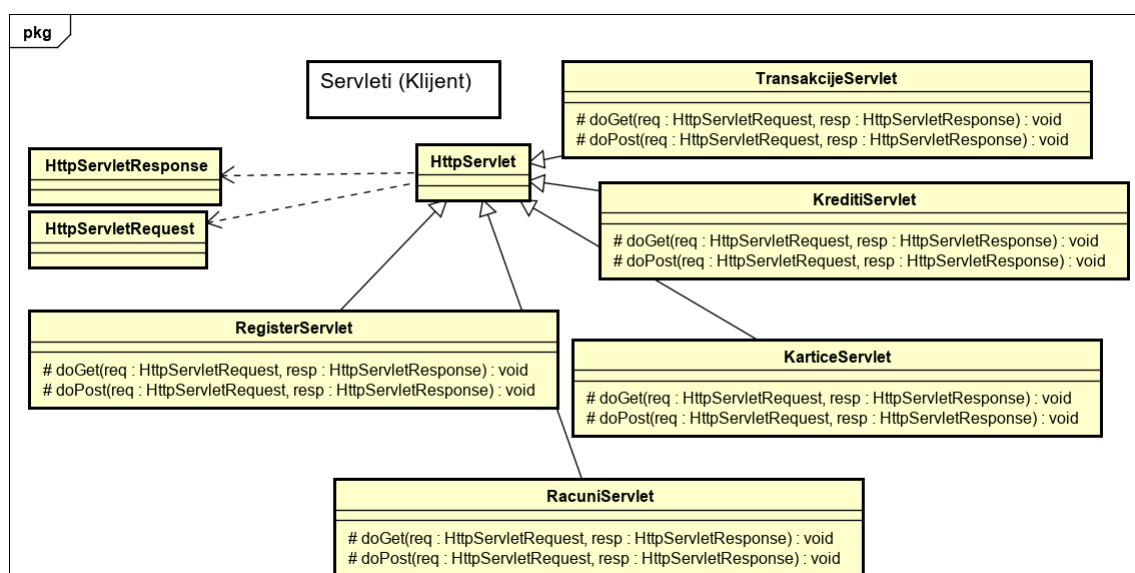
Servleti služe za obradu HTTP zahtjeva. Pomoću nekoliko slika koje slijede opisani su servleti za sve tipove korisnika te oni kojima je oblikovan REST API za spajanje Android aplikacije.



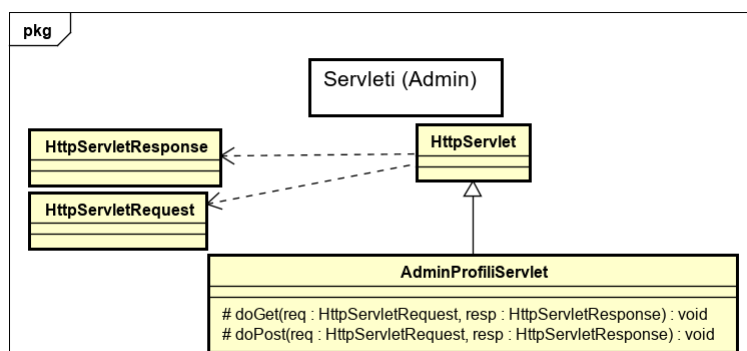
Slika 4.9: Razredi Servleti - općenite operacije dostupne svim korisnicima



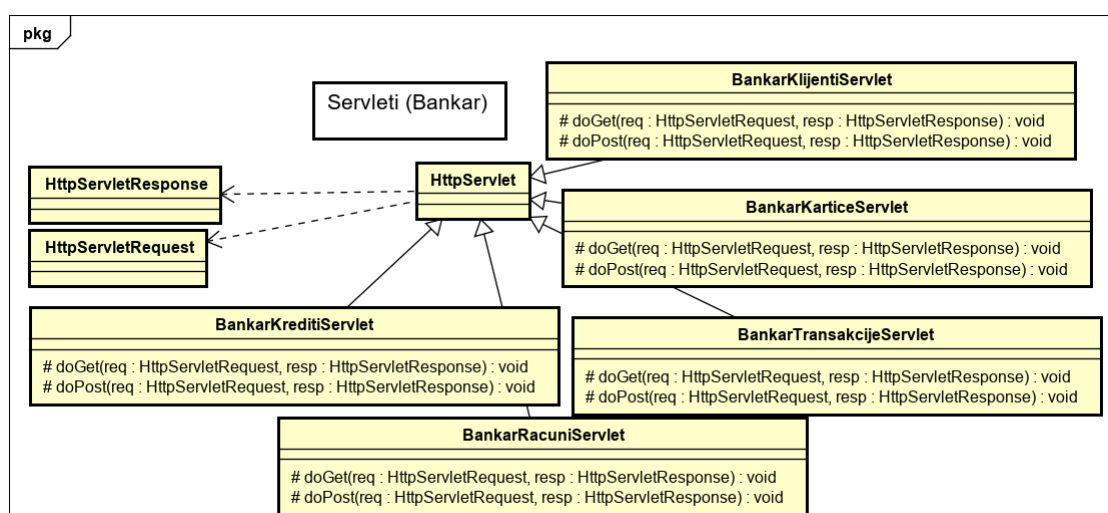
Slika 4.10: Razredi Servleti - funkcije RESTful API-a



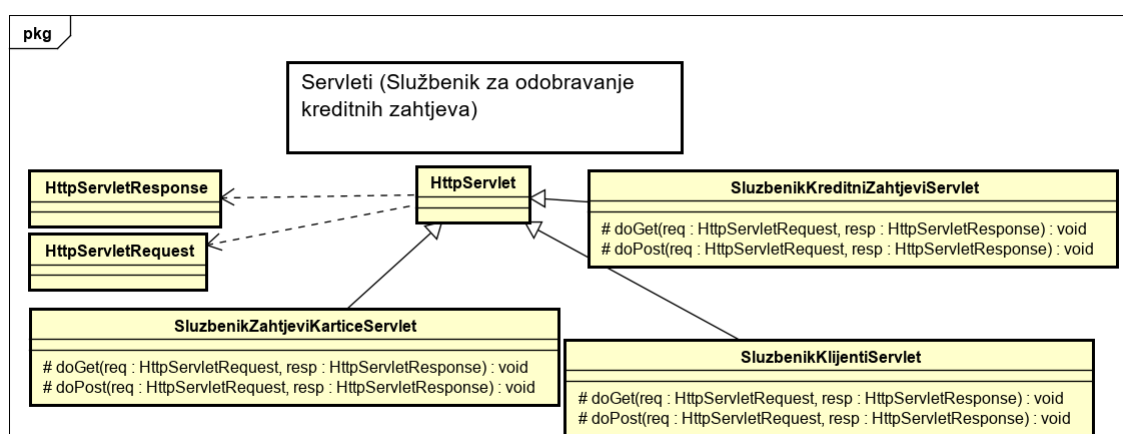
Slika 4.11: Razredi Servleti - klijent



Slika 4.12: Razredi Servleti - administrator



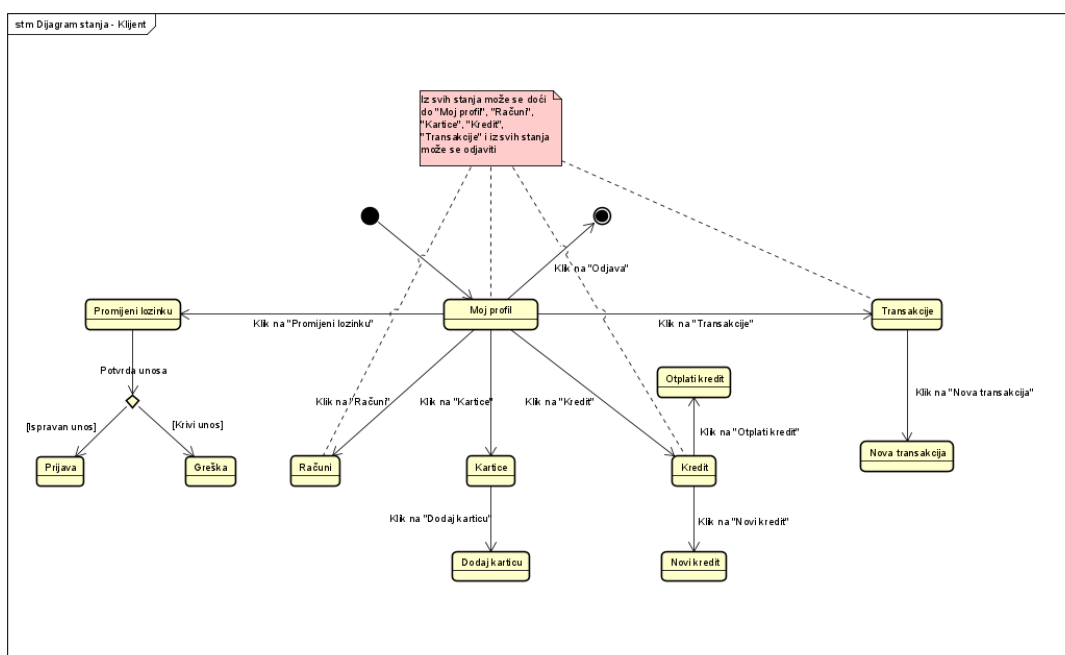
Slika 4.13: Razredi Servleti - bankar



Slika 4.14: Razredi Servleti - službenik za odobravanje kreditnih zahtjeva

4.3 Dijagram stanja

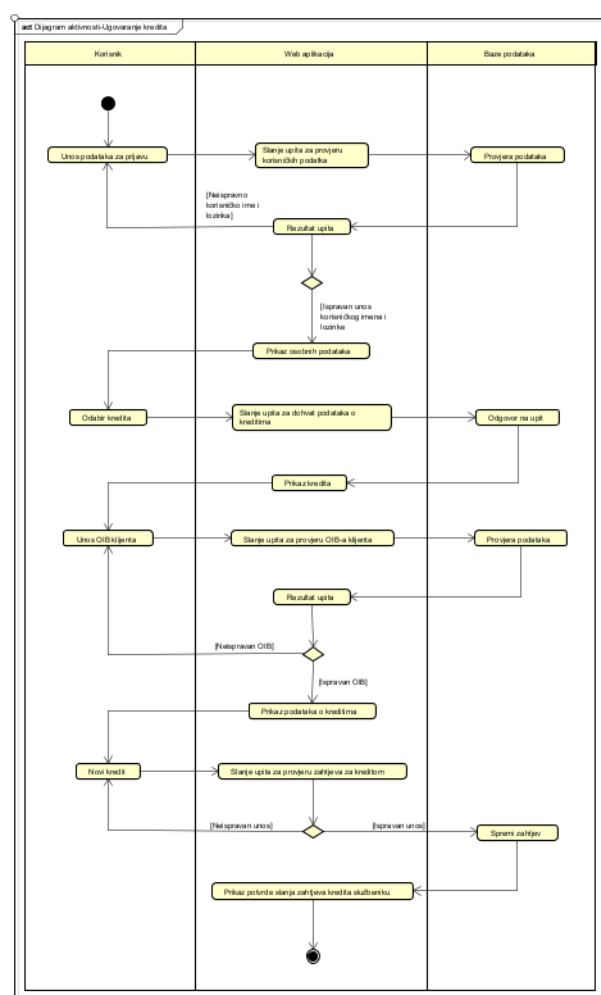
Dijagram stanja spada pod ponašajne i dinamičke dijagrame te prikazuje stanja objekta i prijelaze iz jednog stanja u drugo temeljene na događajima. Na slici 4.9 prikazan je dijagram stanja za registriranog korisnika. Nakon prijave, klijentu se prikazuje početna stranica na kojoj može pregledati svoj profil odnosno osobne podatke. Klikom na "Promijena lozinke" klijent mijenja svoju lozinku. Odabirom "Računa" klijent ima uvid u stanje svojih računa. Klikom na "Kartice" prikazuju mu se podatci o karticama te odabirom "Dodaj karticu" klijent može zatražiti novu kreditnu karticu koju odobrava službenik. Klijent svoje kredite može pregledavati klikom na "Krediti", a odabirom "Novi kredit" poslati zahtjev za novim kreditom službeniku za odobravanje kredita. Klijent odabirom "Otplati kredit" može prijevremeno otplatiti kredit. Pregled transakcija dostupan je klikom na "Transakcije", a klijent može obaviti prijenos sredstava s računa odabirom "Nova transakcija".



Slika 4.15: Dijagram stanja

4.4 Dijagram aktivnosti

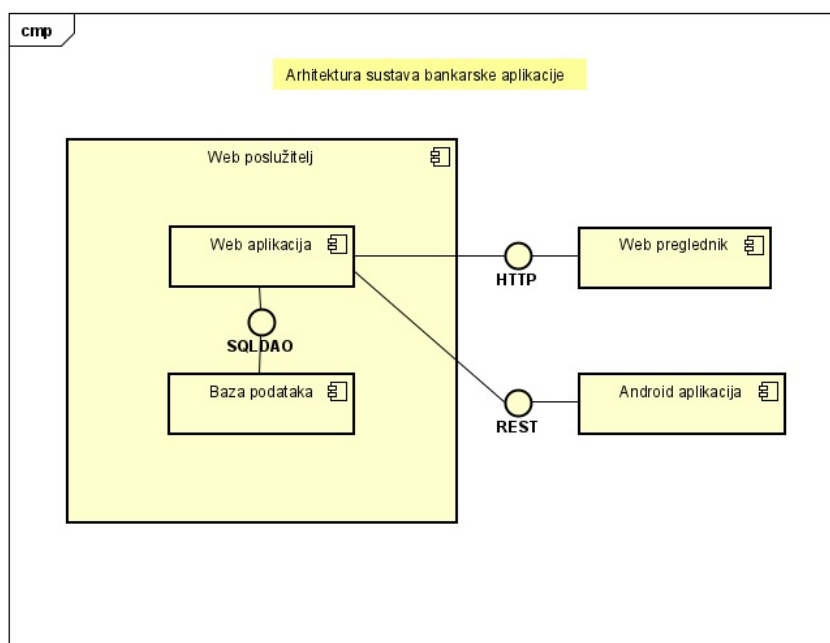
Dijagram aktivnosti je ponašajni dijagram i dinamički dijagram. Dijagrami aktivnosti vrlo su slični dijagramima stanja, ali s drugačijom semantikom elemenata s kojima se modelira. Primjenjuje se za opis modela toka upravljanja ili toka podataka. U modeliranju toka upravljanja svaki novi korak poduzima se nakon završenog prethodnog. Na dijagramu aktivnosti 4.10 prikazan je proces ugovaranja kredita. Korisnik se prijavi u sustav, klikne na "Kredit", i upisuje OIB klijenta koji je zatražio kredit. Korisniku se ispisuju krediti klijenta te nakon toga odabirom "Novi kredit" te unosi odgovarajuće podatke u polja. Nakon što je ispunio ispravno sva polja za unos, korisnik odabire "Provedi zahtjev" te se zahtjev za kredit sprema u bazu podataka.



Slika 4.16: Dijagram aktivnosti

4.5 Dijagram komponenti

Dijagram komponenti prikazuje najvažnije komponente aplikacije i njihov međusobni odnos. Na web poslužitelju nalazi se web aplikacija i baza podataka. Web aplikacija komunicira s bazom podataka preko SQL upita koji se ostvaruju uporabom klase SQLDAO koja implementira DAO sučelje. Pomoću web preglednika korisnik pristupa web aplikaciji. Web preglednik i aplikacija komuniciraju pomoću HTTP protokola. Android aplikacija komunikaciju s web poslužiteljem ostvaruje spajanjem na RESTful dio web poslužitelja.



Slika 4.17: Dijagram komponenti

5. Implementacija i korisničko sučelje

5.1 Korištene tehnologije i alati

Za izradu web aplikacije korištene su brojne tehnologije koje su pomogle u izradi backend i frontend dijela aplikacije. Razvojna okolina koja se koristila za backend dio je IDE Eclipse¹. Eclipse IDE je programska razvojna okolina pisana u Javi razvijena od tvrtke Eclipse Foundation, a može se koristiti za razvoj aplikacija u raznim programskim jezicima (Java, C, C++, Perl, Python,...). Za izradu backenda koristio se programski jezik Java² te tehnologije Java Servleti³ i JSP⁴ (Java Servlet Pages). Za smještaj servleta korišten je Apache Tomcat⁵. Apache Tomcat je open source Web poslužitelj koji implementira nekoliko Java EE specifikacija kao što su Java Servlet, JSP (Java Servlet Pages), Java EL (Java Expression Language). Može se koristiti kao samostalan web server ili kao poslužitelj za servlete i Java Server Pages (JSP) integriran s nekim drugim web poslužiteljem. Kao web poslužitelj koristi se Windows Server⁶ virtualna mašina podignuta preko servisa Microsoft Azure⁷ na kojemu su instalirani svi servisi koji podupiru rad web aplikacije i baze podataka. Baza podataka pokrenuta je pomoću sustava PostgreSQL⁸.

Mobilna aplikacija je izgrađena u programskom jeziku Java koristeći razvojnu okolinu Android studio⁹. Mobilna aplikacija spaja se na RESTful dio web poslužitelja preko kojeg vrši prijavu u sustav i dohvaćanje podataka. Dizajna Android aplikacije postignut je korištenjem sustava Adobe XD¹⁰ i Google Material Design¹¹ kao vizualnim jezikom.

Pomoću alata Astah Proffesional¹² nacrtani su UML dijagrami, a kao sustav za

¹<https://www.eclipse.org/>

²<https://www.java.com/>

³<https://www.oracle.com/technetwork/java/javaee/servlet/index.html>

⁴<https://www.oracle.com/technetwork/java/>

⁵<https://tomcat.apache.org/>

⁶<https://www.microsoft.com/en-us/cloud-platform/windows-server>

⁷<https://portal.azure.com/>

⁸<https://www.postgresql.org/>

⁹<https://developer.android.com/studio/index.html>

¹⁰<https://www.adobe.com/products/xd.html>

¹¹<https://www.material.io/>

¹²<https://http://astah.net/editions/professional>

upravljanje izvornim kodom korišten je Git¹³. Udaljeni repozitorij projekta je dostupan na web platformi GitLab¹⁴. Članovi tima komunicirali su korištenjem aplikacije Slack¹⁵.

¹³<https://git-scm.com/>

¹⁴<https://www.gitlab.com/>

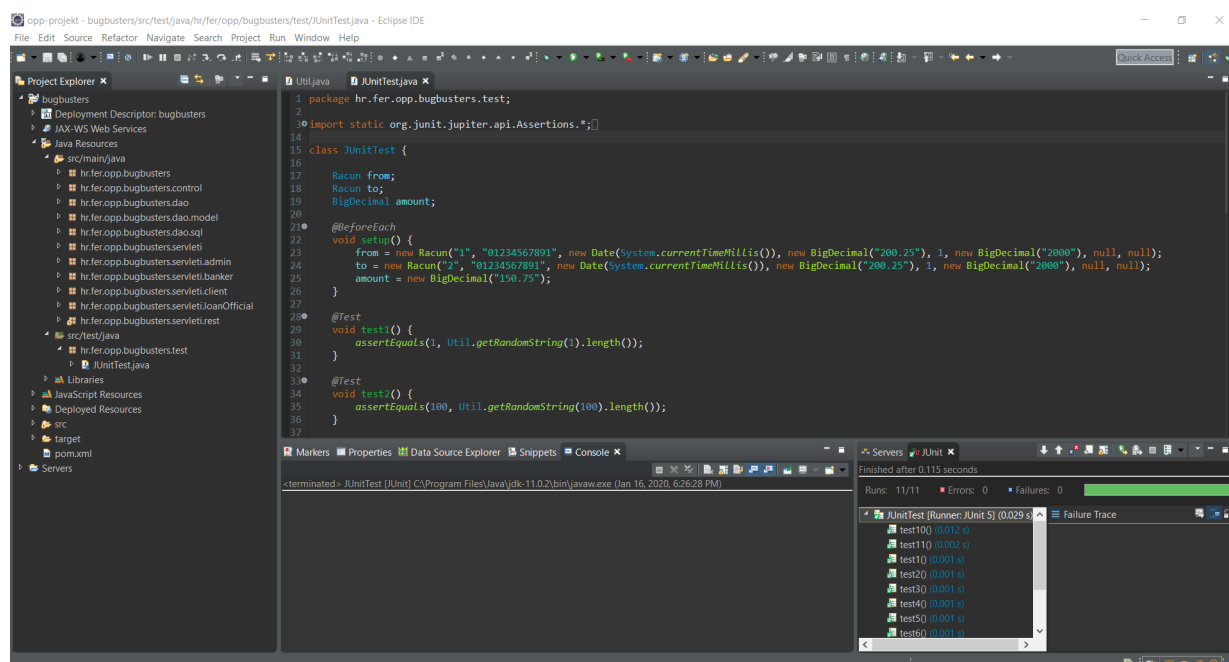
¹⁵<https://www.slack.com/>

5.2 Ispitivanje programskog rješenja

dio 2. revizije

U ovom poglavlju je potrebno opisati provedbu ispitivanja implementiranih funkcionalnosti na razini komponenti i na razini cijelog sustava s prikazom odabranih ispitnih slučajeva. Studenti trebaju ispitati temeljnu funkcionalnost i rubne uvjete..

5.2.1 Ispitivanje komponenti (JUnit testiranje)



Slika 5.1: JUnit testovi

```
package hr.fer.opp.bugbusters.test;
```

```
import static org.junit.jupiter.api.Assertions.*;
```

```
import java.math.BigDecimal;
```

```
import java.sql.Date;
```

```
import org.junit.jupiter.api.BeforeEach;
```

```
import org.junit.jupiter.api.Test;
```

```
import hr.fer.opp.bugbusters.control.Util;
import hr.fer.opp.bugbusters.control.Util.TransakcijaCollection;
import hr.fer.opp.bugbusters.dao.model.Racun;

class JUnitTest {

    Racun from;
    Racun to;
    BigDecimal amount;

    @BeforeEach
    void setup() {
        from = new Racun("1", "01234567891", new
            Date(System.currentTimeMillis()), new BigDecimal("200.25"), 1, new
            BigDecimal("2000"), null, null);
        to = new Racun("2", "01234567891", new
            Date(System.currentTimeMillis()), new BigDecimal("200.25"), 1, new
            BigDecimal("2000"), null, null);
        amount = new BigDecimal("150.75");
    }

    @Test
    void test1() {
        assertEquals(1, Util.getRandomString(1).length());
    }

    @Test
    void test2() {
        assertEquals(100, Util.getRandomString(100).length());
    }

    @Test
    void test3() {
        assertThrows(IllegalArgumentException.class, () ->
            Util.getRandomString(0));
    }
}
```

```
@Test
void test4() {
    assertThrows(IllegalArgumentException.class, () ->
        Util.getRandomString(-10));
}

@Test
void test5() {
    assertThrows(NullPointerException.class, () -> Util.doTransaction(null,
        to, amount));
}

@Test
void test6() {
    assertThrows(NullPointerException.class, () -> Util.doTransaction(from,
        null, amount));
}

@Test
void test7() {
    assertThrows(NullPointerException.class, () -> Util.doTransaction(from,
        to, null));
}

@Test
void test8() {
    assertThrows(IllegalArgumentException.class, () ->
        Util.doTransaction(from, to, new BigDecimal("2500")));
}

@Test
void test9() {
    assertThrows(IllegalArgumentException.class, () ->
        Util.doTransaction(from, to, new BigDecimal("0")));
}

@Test
void test10() {
```

```
    assertThrows(IllegalArgumentException.class, () ->
        Util.doTransaction(from, to, new BigDecimal("-1")));
}

@Test
void test11() {
    TransakcijaCollection tc = Util.doTransaction(from, to, amount);
    assertEquals(new BigDecimal("49.50"), tc.from.getStanje());
    assertEquals(new BigDecimal("351.00"), tc.to.getStanje());
}

}
```

5.2.2 Ispitivanje sustava

Potrebno je provesti i opisati ispitivanje sustava koristeći radni okvir Selenium¹⁶. Razraditi **minimalno 4 ispitna slučaja** u kojima će se ispitati redovni slučajevi, rubni uvjeti te poziv funkcionalnosti koja nije implementirana/izaziva pogrešku kako bi se vidjelo na koji način sustav reagira kada nešto nije u potpunosti ostvareno. Ispitni slučaj se treba sastojati od ulaza (npr. korisničko ime i lozinka), očekivanog izlaza ili rezultata, koraka ispitivanja i dobivenog izlaza ili rezultata.

Izradu ispitnih slučajeva pomoću radnog okvira Selenium moguće je provesti pomoću jednog od sljedeća dva alata:

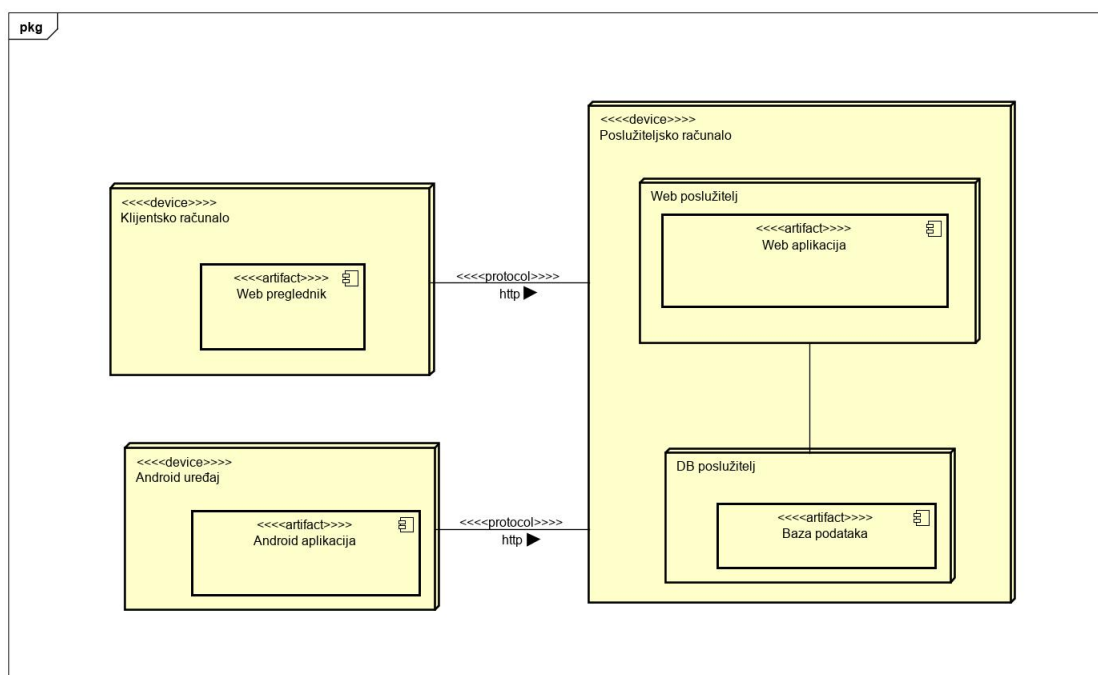
- dodatak za preglednik **Selenium IDE** - snimanje korisnikovih akcija radi automatskog ponavljanja ispita
- **Selenium WebDriver** - podrška za pisanje ispita u jezicima Java, C#, PHP koristeći posebno programsko sučelje.

Detalji o korištenju alata Selenium bit će prikazani na posebnom predavanju tijekom semestra.

¹⁶<https://www.seleniumhq.org/>

5.3 Dijagram razmještaja

Slika prikazuje dijagram razmještaja u sustavu Internet bankarstva. Klijentsko računalo povezuje se s poslužiteljem putem Web preglednika putem http protokola. Android uređaji koriste Android aplikaciju za komunikaciju s poslužiteljem koja se također odvija http protokolom. Poslužiteljsko računalo sastoji se od Web poslužitelja na kojemu se nalazi Web aplikacija i od DB poslužitelja na kojemu se nalazi baza podataka.



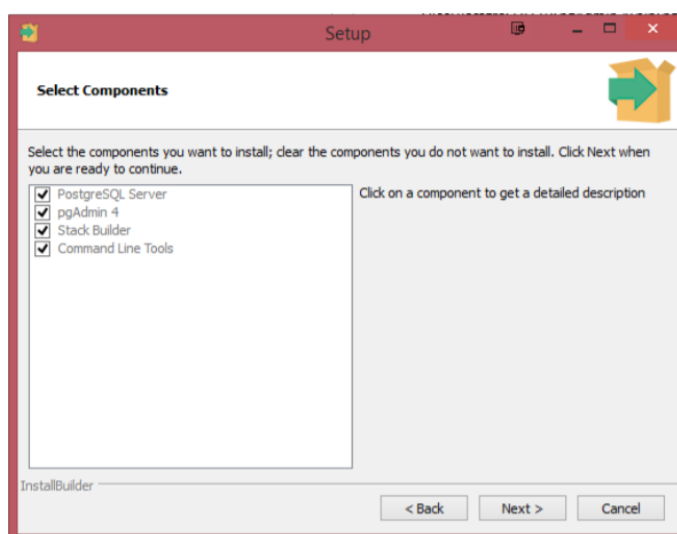
Slika 5.2: Dijagram razmještaja

5.4 Upute za puštanje u pogon

Web aplikacija Bugbusters banka puštena je u pogon pomoću servisa Microsoft Azure, gdje je i postavljen kod same aplikacije (backend i frontend). Aplikacija se pokreće pomoću web preglednika (npr. Google Chrome, Mozilla Firefox...) upisom adrese <http://104.45.11.92/>. Ovu adresu dodjelio je odabrani servis na kojem je aplikacija puštena u pogon. Upisom navedene adrese u web preglednik otvara se stranica za prijavu (engl. login) korisnika u aplikaciju. Za daljnji rad s aplikacijom potrebno je unijeti valjano korisničko ime i lozinku, a u slučaju prve prijave u sustav potrebno je unijeti OIB i korisnički ključ, te zatim promijeniti lozinku. Baza podataka pokrenuta je pomoću sustava PostgreSQL. Svi entiteti baze podataka opisani su java klasama čiji se kod nalazi unutar direktorija izvorniKod, točnije u poddirektoriju dao/model. Veza sa bazom podataka i mogućnost postavljanja upita nad bazom podataka ostvarena je pomoću klasa SQLConnectionProvider i SQLDAO. Ove klase služe kako bi se izradila funkcionalna baza podataka koju aplikacija može koristiti za svoj rad, tj. dohvaćanje, brisanje i izmjena podataka unutar baze podataka.

Instalacija i konfiguracija poslužitelja baze podataka

Potrebno je preuzeti PostgreSQL (<https://www.postgresql.org/download/>). Nakon pokretanja instalacije, kod odabira komponenti koje će se instalirati, obavezno odabrati PostgreSQL Server i PgAdmin4 (preporučeno je odabrati i ostale komponente).



Slika 5.3: Odabir komponenti za instalaciju

Zatim se unosi lozinka za administratora i odabire se port za pristup bazi podataka (defaultni port 5432). Kliknuti na "Next" kako bi započela instalacija, kada je proces instalacije završen kliknuti na "Finish".

Nakon instalacije baze podataka potrebno ju je osposobiti za BugBusters aplikaciju. Potrebno je kreirati novu praznu bazu podataka imena "bugbusters" te novog korisnika SUBP-a korisničkog imena "bugbustersUser" te lozinke "bbpassword1234" i dati mu ovlasti nad kreiranom bazom.

Pokretanje web aplikacije

Za pokretanje web aplikacije potrebno je instalirati servlet container Tomcat 9¹⁷ inačice core. Nakon preuzimanja zip arhive s navedenog linka, raspakirajte ju. U direktorij webapps, unutar raspakiranog direktorija, potrebno je smjestiti .war datoteku objavljenu na GitLab sustavu u direktoriju "izvrsniKod". Zadnji korak je pokretanje samog servlet containera. Pozicionirajte se pomoću naredbenog retka u raspakirani direktorij u poddirektorij imena "bin". Nakon toga je potrebno izvesti naredbu "startup.bat" ukoliko se koristi Windows radno okruženje ili "startup.sh" ukoliko se koristi Unix-based radno okruženje.

Pokretanje android aplikacije Mobilna android aplikacija objavljena je na F-Droid trgovini, gdje ju je moguće preuzeti i zatim instalirati na vlastiti mobilni uređaj. Spajanje na RESTful dio web poslužitelja omogućuje mobilnoj aplikaciji komunikaciju s bazom podataka i prijavu u sustav.

¹⁷<https://tomcat.apache.org/download-90.cgi>

6. Zaključak i budući rad

Zadatak našeg tima bio je razvoj web i mobilne aplikacije za online bankarstvo. Realizacija projekta podijeljena je u dvije faze.

U prvoj fazi projekta okupljen je tim za razvoj aplikacije. Na prvom sastanku definirana je ideja projekta te izvršena podjela uloga članovima tima. U prvoj fazi definiran je i napravljen temelj cijelog sustava. Značajan dio dokumentacije napisan u ovoj fazi projekta uvelike je olakšao daljnji rad pri realizaciji aplikacije. Pisanjem funkcionalnih i nefunkcionalnih zahtjeva, crtanjem uml dijagrama, izradom modela baze podataka pomoglo se članovima tima zaduženim za implementaciju aplikacije.

U drugoj fazi projekta naglasak se stavlja na implementaciju aplikacije. Članovi tima zaduženi za backend i frontend intenzivno rade na realizaciji aplikacije kako bi uspjeli implementirati sve dogovorene funkcionalnosti aplikacije. Ostatak tima bio je zadužen za crtanje preostalih uml dijagrama, testiranje i dovršavanje dokumentacije. Zahvaljujući dobro definiranim temeljima projekta u prvoj fazi dokumentacije, ubrzali smo fazu implementacije te spriječili moguće pogreške. Članovi tima stekli su znanja o raznim tehnologijama i alatima za koje su pokazali interes i s kojima će se možda susresti u budućnosti.

Nakon gotove aplikacije i dalje postoji prostor za usavršavanje. U budućnosti bilo bi potrebno doraditi mobilnu aplikaciju kako bi sadržavala sve funkcionalnosti kao i web aplikacija.

Zadovoljni smo postignutim rezultatom te stječenim znanjima kroz ovaj projekt. Osim susretanja sa novim tehnologijama i sudjelovanju u procesu implementacije aplikacije, stekli smo osjećaj rada u timu te važnosti dobre organizacije i pravilne raspodjele poslova.

Popis literature

Kontinuirano osvježavanje

Popisati sve reference i literaturu koja je pomogla pri ostvarivanju projekta..

1. Oblikovanje programske potpore, FER ZEMRIS, <http://www.fer.hr/predmet/opp>
2. I. Sommerville, "Software engineering", 8th ed, Addison Wesley, 2007.
3. T.C.Lethbridge, R.Langaniere, "Object-Oriented Software Engineering", 2nd ed. McGraw-Hill, 2005.
4. I. Marsic, Software engineering book“, Department of Electrical and Computer Engineering, Rutgers University, <http://www.ece.rutgers.edu/~marsic/books/SE>
5. The Unified Modeling Language, <https://www.uml-diagrams.org/>
6. Astah Community, <http://astah.net/editions/uml-new>

Indeks slika i dijagrama

2.1	Usluga internet bankarstva Zagrebačke banke	8
2.2	Usluga mobilnog bankarstva Zagrebačke banke	8
2.3	Usluga internet bankarstva OTP banke	9
2.4	Usluga mobilnog bankarstva OTP banke	9
3.1	Dijagram obrasca uporabe, ponašanje svih korisnika	28
3.2	Dijagram obrasca uporabe, ponašanje administratora	29
3.3	Dijagram obrasca uporabe, ponašanje službenika za odobravanje kre- dita	30
3.4	Dijagram obrasca uporabe, ponašanje klijenta	31
3.5	Dijagram obrasca uporabe, ponašanje neregistriranog klijenta	32
3.6	Dijagram obrasca uporabe, funkcionalnost bankara	33
3.7	Dijagram generalizacija bankara	34
3.8	Sekvencijski dijagram za UC9	36
3.9	Sekvencijski dijagram za UC17	38
3.10	Sekvencijski dijagram za UC21	40
3.11	Sekvencijski dijagram za UC33	42
4.1	Arhitektura sustava s prikazom podsustava i toka podataka	44
4.2	E-R dijagram baze podataka	53
4.3	Razredi modela opisa korisnika	54
4.4	Ostali razredi modela	55
4.5	Ostali razredi modela - nastavak	56
4.6	Razredi zaduženi za kontrolu autentifikacije	57
4.7	Razredi zaduženi za inicijalizaciju aplikacije i konekcije	58
4.8	Razredi DAO sučelja	59
4.9	Razredi Servleti - općenite operacije dostupne svim korisnicima	60
4.10	Razredi Servleti - funkcije RESTful API-a	61
4.11	Razredi Servleti - klijent	61
4.12	Razredi Servleti - administrator	62
4.13	Razredi Servleti - bankar	62

4.14	Razredi Servleti - službenik za odobravanje kreditnih zahtjeva . . .	62
4.15	Dijagram stanja	63
4.16	Dijagram aktivnosti	64
4.17	Dijagram komponenti	65
5.1	JUnit testovi	68
5.2	Dijagram razmještaja	72
5.3	Odabir komponenti za instalaciju	73

Dodatak: Prikaz aktivnosti grupe

Dnevnik sastajanja

1. sastanak

- Datum: u ovom formatu: 4. listopada 2019.
- Prisustvovali: M. Bačić, M. Anušić, D. Milošević, M. Milički, K. Gudelj, M. Vučemilo, L. Bastalić
- Teme sastanka:
 - Upoznavanje
 - Razrada prijedloga projektnog zadatka
 - Podjela uloga u projektu

2. sastanak

- Datum: u ovom formatu: 9. listopada 2019.
- Prisustvovali: M. Bačić, M. Anušić, D. Milošević, M. Milički, K. Gudelj, M. Vučemilo, L. Bastalić
- Teme sastanka:
 - Inicijalni sastanak s asistentom
 - Razgovor oko prijedloga zadatka

3. sastanak

- Datum: u ovom formatu: 15. listopada 2019.
- Prisustvovali: M. Bačić, M. Anušić, D. Milošević, M. Milički, K. Gudelj, M. Vučemilo, L. Bastalić
- Teme sastanka:
 - Budući da je prijedlog zadatka odobren, konkretne upute dobivene od asistenta

4. sastanak

- Datum: u ovom formatu: 22. listopada 2019.
- Prisustvovali: M. Bačić, M. Anušić, D. Milošević, K. Gudelj, L. Bastalić
- Teme sastanka:
 - Povratna informacija od asistenta za trenutni napredak

5. sastanak

- Datum: u ovom formatu: 24. listopada 2019.
- Prisustvovali: M. Bačić, M. Anušić, D. Milošević, M. Milički, K. Gudelj, M. Vučemilo, L. Bastalić
- Teme sastanka:
 - Povratna informacija od asistenta za trenutni napredak

6. sastanak

- Datum: u ovom formatu: 5. studenog 2019.
- Prisustvovali: M. Bačić, M. Anušić, D. Milošević, M. Milički, M. Vučemilo, L. Bastalić
- Teme sastanka:
 - Razrada baze podataka
 - Povratna informacija od asistenta za trenutni napredak
 - Dogovor oko početka implementacije generičkih funkcionalnosti

7. sastanak

- Datum: u ovom formatu: 9. siječnja 2020.
- Prisustvovali: M. Bačić, M. Anušić, D. Milošević, M. Milički, M. Vučemilo, L. Bastalić, K. Gudelj
- Teme sastanka:
 - Alfa inačice aplikacije

Tablica aktivnosti

	Matija Bačić	Klara Gudelj	Lorena Bastalić	Magda Milički	Marija Vučemilo	Marko Anušić	Dominik Milošević
Upravljanje projektom	5	1.5	1.5	1.5	1.5	1.5	1.5
Opis projektnog zadatka	2						
Funkcionalni zahtjevi	0.5						2.5
Opis pojedinih obrazaca	2	6	6				
Dijagram obrazaca	3					3	
Sekvencijski dijagrami				4.5			
Opis ostalih zahtjeva					3		
Arhitektura i dizajn sustava	8						
Baza podataka	3		4	2			
Dijagram razreda	4.5						
Dijagram stanja			3				
Dijagram aktivnosti			3				
Dijagram komponenti							
Korištene tehnologije i alati			2				
Ispitivanje programskog rješenja				x			
Dijagram razmještaja						2	
Upute za puštanje u pogon							
Dnevnik sastajanja	0.5						
Zaključak i budući rad			1				
Popis literature							
Izrada web korisničkog sučelja	1					8	12
Izrada android aplikacije		22			22		
Izrada baze podataka			8	8			
Spajanje s bazom podataka	2						
Back end	35						

Dijagrami pregleda promjena

dio 2. revizije

Prenijeti dijagram pregleda promjena nad datotekama projekta. Potrebno je na kraju projekta generirane grafove s gitlaba prenijeti u ovo poglavlje dokumentacije. Dijagrami za vlastiti projekt se mogu preuzeti s gitlab.com stranice, u izborniku Repository, pritiskom na stavku Contributors.