



Projektni Zadatak

Sistem za strimovanje multimedijalnog sadržaja

Predmet: SE201 UVOD U SOFTVERSKO INŽENJERSTVO

Školska godina 2017/18.

Projektni zadatak

Ime i prezime studenta: Nemanja Kuzmanović

Broj indeksa: 2851

Datum izrade: 18.01.2018.

Profesor: Prof. dr Dragan Domazet

Asistent: Lazar Mrkela

U Beogradu 2018.

Sadržaj

1. Uvod.....	5
2. Opšti opis	6
2.1 OPIS	6
2.2 MODEL SOFTVERSKOG PROCESA	6
2.3 AKTIVNOSTI MODELA SOFTVERSKOG PROCESA – OPIS.....	7
2.4 KORISNICI SISTEMA	8
2.5 PERSPEKTIVA PROIZVODA.....	9
2.5.1 Interfejs sistema.....	9
2.5.2 Korisnički interfejs.....	9
2.5.3 Hardverski interfejs.....	9
2.5.4 Softverski interfejs	9
2.5.5 Komunikacioni interfejs	9
2.5.6 Memorijski zahtevi	9
2.6 ZAHTEVI KORISNIKA	9
2.7 OPŠTA OGRANIČENJA	10
2.8 PRETPOSTAVKE I OGRANIČENJA	10
3. Specifični zahtevi	11
3.1 FUNKCIONALNI ZAHTEVI	11
3.2 FUNKCIONALNI ZAHTEVI KORISNIKA	11
3.2.1 INVENTORY LIST – Pregled sadržaja.....	11
3.2.2 ENTERTAINMENT LIST – Lista sadržaja (filmova i serija).....	11
3.2.3 ENTERTAINMENT REQUESTED FROM INVENTORY LIST – Zahtev sadržaja iz liste.....	12
3.2.4 ADDITION TO INVENTORY – Dodavanje sadržaja u „Moj Sadržaj,,.....	12
3.2.5 INVENTORY DATABASE	12
3.2.6 ENTERTAINMENT VIEWER - Pregled	12
3.2.7 BUY – Kupovina sadržaja.....	12
3.2.8 RENT – Iznajmljivanje	12
3.2.9 CHANGE NOTE – Komentar.....	12
3.2.10 INVENTORY SEARCH – Pretraga	12
3.2.11 USER REGISTRATION – Pravljenje naloga.....	13
3.2.12 USER PROFILE – Profil korisnika	13
3.2.13 ADMIN CONTACT – Kontakt administrator.....	13
3.2.14 REMOTE ACCESS – Udaljeni pristup	13
3.3 FUNKCIONALNI ZAHTEVI ADMINISTRATORA	14

3.3.1 USER DATABASE	14
3.3.2 USER VIEW – Pregled korisnika	14
3.3.3 NEW USER – Novi korisnik	14
3.3.4 CHANGE USER – Promena kredencijala korisnika	14
3.3.5 REMOVE USER – Uklanjanje korisnika	14
3.3.6 REMOTE ACCESS	14
3.4 NEFUNKCIONALNI ZAHTEVI	15
3.4.1 Adaptive Design and User-friendly Interface	15
3.4.2 Performances	15
3.4.3 Security	15
3.5 SLUČAJEVI KORIŠĆENJA	16
3.6 APSTRAKTNI DIJAGRAM SEKVENCI (SISTEM SEKVENCIJALNI DIJAGRAM)	22
3.6.1 Dijagram sekvenci za registrovanje korisnika (gosta)	22
3.6.2 Dijagram sekvenci za prijavu na sistem – GOST	23
3.6.3 Dijagram sekvenci za pregled limitiranog (određenog) sadržaja – GOST	24
3.6.4 Dijagram sekvenci za kontaktiranje administratora - GOST	25
3.6.5 Dijagram sekvenci za registraciju – KORISNIK	26
3.6.6 Dijagram sekvenci prijavljivanja na sistem – KORISNIK	27
3.6.7 Dijagram sekvenci za pretragu sadržaja - KORISNIK	28
3.6.8 Dijagram sekvenci za odabir sadržaja – KORISNIK	29
3.6.9 Dijagram sekvenci kupovine sadržaja – KORISNIK	30
3.6.10 Dijagram sekvenci iznajmljivanja sadržaja – KORISNIK	31
3.6.11 Dijagram sekvenci za kontakt administratora – KORISNIK	32
3.6.12 Dijagram sekvenci za registraciju na sistem – Administrator	33
3.6.13 Dijagram sekvenci za prijavljivanje na sistem – Administrator	34
3.6.14 Dijagram sekvenci za pregled korisnika – Administrator	35
3.6.15 Dijagram sekvenci za dodavanje korisnika – Administrator	36
3.6.16 Dijagram sekvenci za ažuriranje korisnika – Administrator	37
3.6.17 Dijagram sekvenci za brisanje korisnika – Administrator	38
3.7 KLASI I OBJEKTI	39
3.8 SEKVENCIJALNI DIJAGRAM(NIVO UPOTREBLJENIH KLASA)	40
3.9 MODEL BAZE PODATAKA	42
4. ARHITEKTURA SISTEMA	44
4.1 MVC arhitektura	44
4.2 INFRASTRUKTURNI DIJAGRAM	45

4.3 DIJAGRAM ARHITEKTURE.....	46
5. TESTIRANJE.....	47
6. ZAKLJUČAK	60
7. LITERATURA	61

1. Uvod

Dokument sadrži opis projekta i softversku dokumentaciju sistema za online strimovanje multimedijalnog sadržaja. U okviru softverske dokumentacije se pored funkcionalnih i nefunkcionalnih zahteva, i ostalih delova koje dokumentacija sadrži, kroz ceo dokument će se sve detaljnije ulaziti u sistem i dobijati kompletna slika i svrha celog sistema, kao i tehnologije kojima je sistem rađen. Pored dokumenta priložena će biti i implementacija projekta (Beta verzija), koja i radi ono što dokument predstavlja za sistem.

Kompletan projekat se još, pored dokumentacije, sastoji od dijagrama rađenih u PowerDesigneru, MySQL baze podataka, servisa rađenim u PHP skripting jeziku, a glavni deo (core) aplikacije je pisan u JavaScript frontend framework-u Angular-u.

Sistem (platforma) se pokreće u pretraživaču i predstavlja web aplikaciju.

2. Opšti opis

2.1 OPIS

Sistem za strimovanje multimedijalnog sadržaja putem interneta. Platforma koja omogućava korisnicima pretragu i odabir filma ili serije, nakog čega korisnici mogu kupiti odabrani sadržaj koji bi nakon kupovine neograničeno stajao u njihovoj biblioteci na platformi a vezan za nalog korisnika, ili iznajmiti sadržaj, koji bi im bio dostupan za reprodukciju u narednom određenom periodu (24 sati). Reprodukcijski dostupnog sadržaja, prethodno kupljenog ili iznajmljenog, je moguća na platformi i bez pretrage nekog filma ili serije, pristupom u biblioteku korisnika u okviru naloga kojim pristupa sistemu.

Sličnih aplikacija ima nekoliko, neke od njih su: Netflix, Hulu Plus..

Aplikacija je prilagođena korisnicima bez ili sa jako malo iskustva sa računarom, te je mogu svi koristiti, jedino što je potrebno je internet pretraživač, i internet konekcija.

2.2 MODEL SOFTVERSKOG PROCESA

Model softverskog procesa koji će se koristiti za izradu sistema je model vodopada (waterfall model). On prikazuje osnovne aktivnosti procesa : specifikacija, razvoj, provera (validacija) i evolucija. Ove aktivnosti su prikazane kao posebne faze procesa, kao što su: specifikacija zahteva, projektovanje (dizajn) softvera, primena (implementacija), testiranje i dr. koje se jedna za drugom realizuju.

2.3 AKTIVNOSTI MODELA SOFTVERSKOG PROCESA – OPIS

1) Specifikacija softvera

Zahtevi ovog sistema za strimovanje multimedijalnog sadržaja putem interneta, koji korisniku jasno definišu potrebe su odabir filma ili serije (putem pretrage liste filmova na sistemu), izbor kupovine filma/serije ili izbor iznajmljivanja na zahtev (video on demand – VOD) u zavisnosti od želje korisnika, reprodukcija odabranog sadržaja.

2) Razvoj softvera

Korisnici pretražuju filmove ili serije do trenutka kada odaberu željeni, u tom trenutku oni imaju mogućnosti kupovine filma/serije ili iznajmljivanja na zahtev (gde dobijaju mogućnost reprodukcije sadržaja u narednih 24 sati), bilo da korisnici izaberu iznajmljivanje ili kupovinu, sadržaj se smešta u deo koji je povezan sa nalogom korisnika (My Library), te iz njega mogu vršiti reprodukciju istog ili u narednom periodu, ili u neograničenom periodu ukoliko su odabrali da kupe sadržaj. Takođe korisnici nisu u obavezi da pretražuju sadržaj koji sistem nudi, nego mogu direktno odabrati da reprodukuju sadržaj koji im stoji na raspolaganju, a ranije su ga ili kupili ili iznajmili.

3) Validacija softvera

Validacijom softvera utvrđujemo da li smo zadovoljili specifikaciju kao i očekivanje samog korisnika sistema. Sistem prolazi kroz niz faza same validacije, ali i testiranja. Kroz alfa testiranja samog tima sistema gde je uključen i korisnik preko beta testiranja gde se većoj ali određenoj grupi korisnika daje sistem na korišćenje, bliže utvrđujemo da li su zahtevi korisnika ispunjeni, i da li korisnik zadovoljan sistemom.

4) Evolucija softvera

Evolucijom softvera tokom njegovog životnog veka treba omogućiti da se softver proširuje i da mu se dodaju funkcionalnosti u zavisnosti od potreba i zahteva korisnika koji se mogu u budućnosti pojaviti, ili zahteva u pogledu izmena već implemetiranog rešenja. Sistem mora biti fleksibilan, te da omogućava buduće proširenje i nakon same izrade. Kako zahtevi rastu, nove funkcionalnosti za korisnika moraju biti implementirane, a svakako i sam sadržaj redovno ažuriran.

2.4 KORISNICI SISTEMA

Sistem poznaje dva tipa korisnika sistema, i to:

1. Administrator
2. Korisnik

Korisnik, ukoliko nema već registrovan nalog na sistemu, prvenstveno treba da napravi novi nalog, gde upisuje osnovne informacije o sebi, poput imena, prezimena, potom upisuje jedinstveno korisničko ime, dakle, na sistemu ne postoje dva korisnika sa istim korisničkim imenom, i upisuje lozinku koja je u opsegu sigurne lozinke, što znači da mora imati 8 ili više karaktera, kao i barem jedan karakter veliko slovo, i jedan broj.

Ukoliko korisnik već ima nalog, ulogovaće se na sistem pomoću svojih kredencijala (korisničkog imena i lozinke). Ovako ulogovan korisnik prvenstveno može ući u deo sa listom svih filmova/serija, gde pored osnovne pretrage pomoću ulaznih uređaja, ima i pretragu (search) za brže i lakše traženje željenog filma/serije. Kada korisnik odabere željeni sadržaj, on onda ima opciju za kupovinu tog sadržaja ili iznajmljivanje. Kupovina omogućava neograničeno gledanje u bilo koje doba tog sadržaja, dok iznajmljivanje, koje je znatno jeftinije, omogućuje korisniku da gleda narednih 24 sati taj sadržaj. Nakon kupovine, korisnik ima mogućnost pregleda svih svojih filmova, odakle ih može gledati, ili uklanjati iz liste. (Filovi koji su iznajmljeni iz bilo kog razloga korisnik može obrisati u bilo kom trenutku, ostali ostaju na nalogu bez mogućnosti brisanja). Takođe, korisnici mogu u bilo kom trenutku kontaktirati administratora putem forme koja se nalazi na sajtu. Administrator ima mogućnosti dodavanja novih korisnika, ažuriranje starih (dodavanje rola admina, blokiranje korisnika...), pregled svih korisnika. Pored toga, administrator može pregledati postojeći sadržaj (filmove/serije). Sve ovo radi pomoću predefinisane forme u kojoj menja željeni tip podataka, ili bira opciju za recimo brisanje.

Korisnici koji nisu ulogovani na sistem pored osnovnog pregleda platforme, imaju mogućnosti umesto pregleda svih filmova, pregled nekoliko nasumičnih filmova iz baze podataka, sa jako ograničenim opisom svakog filma, i bez ikakve opcije da urade sa istim, jedino što mogu, je kontaktiranje administratora putem forme, ili registracija tj. login na sistem.

2.5 PERSPEKTIVA PROIZVODA

Nezavisna aplikacija sa sledećim redosledom:

2.5.1 Interfejs sistema

Korisnik ima mogućnost unosa komentara u bazu podataka koji je vezan za film/seriju koji su kupljeni, dok administrator ima mogućnosti unosa u bazu korisnika i sadržaja putem jednostavne, intuitivne forme u vodu polja koja treba popuniti

2.5.2 Korisnički interfejs

Korisnik ima jednostavan, i dobar pregled informacija iz baze, sadržaj je koncizno i jasno raspoređen i prikazan tako da svako može lako pronaći željni sadržaj.

2.5.3 Hardverski interfejs

Sistem je veoma „lagan“, (lightweight) i korisniku nije potreban nikakav specijalni hardver kako bi pokrenuo aplikaciju.

2.5.4 Softverski interfejs

Korišćeni jezici: MySQL, Angular, PHP

2.5.5 Komunikacioni interfejs

Korisnik pristupa aplikaciji putem interneta, bez obzira na tehnologiju pristupa. Bilo LAN, Wireless, GRPS..

2.5.6 Memorijski zahtevi

Minimalni RAM i HDD su sasvim dovoljni za nesmetano pokretanje aplikacije, sve što se dešava, dešava se na serverima sistema, korisnik samo mora imati toliko rama da pokrene internet pretraživač bez problema, i flesh player.

2.6 ZAHTEVI KORISNIKA

Aplikacija je veoma intuitivna i prilagođena korisnicima (User-friendly), te i korisnici bez iskustva je mogu koristiti bez problema, pored toga, aplikacija je respozivnog (responsive) dizajna, što znači da ju je moguće i na pametnim uređajima pokrenuti, gde se elementi skaliraju u zavisnosti od uređaja, kako bi korisnik što lakse koristio aplikaciju.

2.7 OPŠTA OGRANIČENJA

Osnovno, i najbitnije ograničenje, bez koga jednostavno ne može, jeste da korisnik ima pristup internetu, kako bi pristupio ovoj veb aplikaciji koja se izvršava u pretraživaču. Baza podataka i sistem moraju biti zaštićeni od malicioznih korisnika, kao i od korisnika koji pokušaju da izbegnu autentifikaciju. Kako bi bazu zaštitili moramo ograničiti svako polje forme za unos sigurnosnim parametrima (escape string). Pored toga, korisnici moraju popuniti svako polje svake forme kako bi poslali (submitovali) željenu akciju. Podatke u bazi posebno štitimo enkripcijom da, ukoliko i dođe do neovlašćenog pristupa zbog nekog propusta u aplikaciji, maliciozni korisnik bude ograničen sa takvim informacijama. Ukoliko korisnik izazove grešku u sistemu, sistem mora to sačuvati i ispisati korisniku da je došlo do greške ali mora grešku i sam ispraviti kako ne bi došlo do „curenja,, informacija.

2.8 PRETPOSTAVKE I OGRANIČENJA

Pretpostavlja se da je korisnik upoznat sa funkcijama i mogućnostima aplikacije, te da je upoznat sa korišćenjem veb aplikacija, kao i da ima pristup internetu. Svaka promena na sistemu treba blagovremeno da obavesti korisnika o tome.

3. Specifični zahtevi

Specifični softverski zahtevi obuhvataju funkcionalne i nefunkcionalne zahteve, a predstavljaju način na koji bi bilo potrebno razvijati neki sistem. Takođe je potrebno sadržati slučajeve korišćenja koji opisuju samu interakciju korisnika i sistema.

3.1 FUNKCIONALNI ZAHTEVI

Funkcionalni zahtevi opisuju šta sistem radi ili se očekuje da radi. Ti zahtevi uključuju opis obrade koju sistem mora da obavlja, detalje ulaza u sistem, detalje izlaza koji se očekuju od sistema, detalje o podacima koji se moraju čuvati u sistemu.

3.2 FUNKCIONALNI ZAHTEVI KORISNIKA

3.	Functional Requirements	REQ_0005	Undefined	0	Undefined	Draft
3.1	Inventory list	REQ_0006	Undefined	0	Undefined	Draft
3.1.1	Entertainment list	REQ_0007	Undefined		Undefined	Draft
3.1.2	Entertainment requested from Inventory list	REQ_0008	Undefined		Undefined	Draft
3.1.3	Addition to Inventory	REQ_0009	Undefined		Undefined	Draft
3.1.4	Remove from Inventory	REQ_0010	Undefined		Undefined	Draft
3.2	Inventory Database	REQ_0011	Undefined	0	Undefined	Draft
3.2.1	Entertainment Viewer	REQ_0012	Undefined		Undefined	Draft
3.2.2	Buy	REQ_0013	Undefined		Undefined	Draft
3.2.3	Rent	REQ_0014	Undefined		Undefined	Draft
3.2.4	Change Note	REQ_0015	Undefined		Undefined	Draft
3.3	Inventory Search	REQ_0016	Undefined		Undefined	Draft
3.4	User Registration	REQ_0017	Undefined	0	Undefined	Draft
3.4.1	User Profile	REQ_0018	Undefined		Undefined	Draft
3.5	Admin Contact	REQ_0024	Undefined		Undefined	Draft
3.6	Remote Access	REQ_0025	Undefined		Undefined	Draft

Slika 1 – Funkcionalni zahtevi korisnika

3.2.1 INVENTORY LIST – Pregled sadržaja

Sistem prikazuje sve dostupne filmove i serije, u okviru svakog filma ili serije koji se prikazuje korisniku se nalaze određene informacije o datom sadržaju

3.2.2 ENTERTAINMENT LIST – Lista sadržaja (filmova i serija)

Lista koja prikazuje korisniku sve filmove i serije iz baze podataka sa sledećim informacijama, neki od njih su, opis filma/serije, naziv, godina kada je izašao/la, lista od nekoliko glavnih glumaca, ocena sa sajta imdb, novac koji je film/serija zaradio/la, trajanje, sertifikat za koga je film/serija namenjen/a.

3.2.3 ENTERTAINMENT REQUESTED FROM INVENTORY LIST – Zahtev sadržaja iz liste

Sistem vraća informacije o izabranom filmu ili seriji koju je korisnik iz liste odabrao.

3.2.4 ADDITION TO INVENTORY – Dodavanje sadržaja u „Moj Sadržaj„

Redovno će se ažurirati svi filmovi i serije ukoliko dođe do potrebe za tim, ukoliko se promeni ocena, ili poraste zarada samog filma/serije i slični zahtevi. Dok će se takođe redovno dodavati kako najnoviji tako i stari filmovi i serije.

3.2.5 INVENTORY DATABASE

Sistem će biti povezan sa bazom podataka koji će čuvati pored korisnika, sve filmove i serije koji su dostupni na sistemu, kao i informacije o svakom filmu ili seriji individualno, te glumcima, žanrovima...

3.2.6 ENTERTAINMENT VIEWER - Pregled

Sistem omogućava korisnicima da filmove ili serije pregledaju po već predefinisanim kategorijama (Drama, Horror), takođe mogu izabrati da se sortiraju po godini, nazivu...

3.2.7 BUY – Kupovina sadržaja

Sistem omogućava korisnicima da kupe film, koji će se dodati u njihov personalni inventar i u bilo kom trenutku mogu pristupiti tom filmu za gledanje. Takođe istovremeno će se taj film povezati sa korisnikom u bazi podataka.

3.2.8 RENT – Iznajmljivanje

Korisnici na sistemu mogu i odabrati opciju iznajmljivanja filma, što će im omogućiti da isti gledaju u naredna 24h nakon čega film nestaje iz personalnog inventara i ukoliko ponovo požele da ga gledaju moraće ili da ga kupe ili iznajme ponovo.

3.2.9 CHANGE NOTE – Komentar

Uz svaki film/seriju kada korisnik kupi/iznajmi dobija note funkciju koja je inicijalno prazna i može je izmeniti ukoliko poželi da doda bilo kakvu belešku uz dati film/seriju (primer: stigao sam do 20og minuta; dosadan film; Odličan film gledaj opet...)

3.2.10 INVENTORY SEARCH – Pretraga

Korisnik putem polja search može vrlo lako i brzo pretražiti sistem za željenim filmom ukoliko to poželi. Opcija je vezana direktno sa bazom tako da korisnik traži film koji se nalazi direktno u bazi podataka. Može pretraživati po već definisanim kategorijama, ili odabrati iz padajućeg menija, godinu, naziv i slično.

3.2.11 USER REGISTRATION – Pravljenje naloga

Svakom korisniku je omogućeno da se vrlo lako registruje na sistem. Unosom e-maila, korisničkog imena i lozinke, korisnik se registruje na sistem (proširivo ukoliko postoji potreba za tim), a potom, sistemu može pristupiti putem korisničkog imena i lozinke. Korisnik nema mogućnosti izmene korisničkog imena. Obavljaće sledeće funkcije:

3.2.12 USER PROFILE – Profil korisnika

Korisnik kada pristupi svom profilu, ima mogućnost pregleda svog imena i prezimena, kao i dodavanja ili izmenu istih. Pored toga korisnik po želji može dodati sliku na svoj profil, i može pregledati osnovne informacije o profilu. Ima takođe mogućnost pregleda svih akcija vezanih za profil.

3.2.13 ADMIN CONTACT – Kontakt administrator

Korisnici sistema, ili korisnici koji nisu još na sistemu, mogu kontaktirati administratora putem predefinisane forme ukoliko postoji potreba za tim, ukoliko imaju bilo kakvo pitanje ili im je potreba pomoć u vezi platforme.

3.2.14 REMOTE ACCESS – Udaljeni pristup

Korisnici mogu sistemu pristupiti samo putem interneta. Sa bilo kog uređaja koji ima internet, pošto je u pitanju platforma u vidu veb strane.

3.3 FUNKCIONALNI ZAHTEVI ADMINISTRATORA

3.5	User Database	REQ_0019	Undefined	0	Undefined	Draft
3.5.1	User View	REQ_0020	Undefined		Undefined	Draft
3.5.2	New User	REQ_0021	Undefined		Undefined	Draft
3.5.3	Change User	REQ_0022	Undefined		Undefined	Draft
3.5.4	Remove User	REQ_0023	Undefined		Undefined	Draft
3.6	Remote Access	REQ_0025	Undefined		Undefined	Draft

Slika 2 – Funkcionalni zahtevi administratora

3.3.1 USER DATABASE

Sistem u bazi podataka čuva i sve informacije o korisnicima kao i korisnike sistema.

3.3.2 USER VIEW – Pregled korisnika

Sistem ima mogućnost pregleda liste svih korisnika, kao i pretragu korisnika. Takođe se mogu sortirati u listi po određenoj kategoriji.

3.3.3 NEW USER – Novi korisnik

Sistem ima mogućnost dodavanja novog korisnika kao i dodavanja filmova ili serija uz korisnički nalog ukoliko dođe do potrebe za tim iz bilo kog razloga (akcija, promocija...)

3.3.4 CHANGE USER – Promena kredencijala korisnika

Sistem ima mogućnost za imenom informacija o samom korisniku, kao i aktivnosti vezanih za taj korisnički nalog.

3.3.5 REMOVE USER – Uklanjanje korisnika

Moguće je obrisati bilo kog korisnika sa sistema iz bilo kog razloga.

3.3.6 REMOTE ACCESS

Administrator se može ulogovati na sistem sa udaljenog računara, i pristupiti svim opcijama koje ima administrator.

3.4 NEFUNKCIONALNI ZAHTEVI

4.	Non-Functional Requirements	REQ_0026	Undefined	0	Undefined	Draft
4.1	Adaptive Design and User-friendly Interface	REQ_0027	Undefined		Undefined	Draft
4.2	Performances	REQ_0028	Undefined		Undefined	Draft
4.3	Security	REQ_0029	Undefined		Undefined	Draft

Slika 3 – Nefunkcionalni zahtevi

3.4.1 Adaptive Design and User-friendly Interface

Platforma ima adaptivni dizajn, jer mora imati mogućnosti pristupa sa raznih uređaja, od tableta i telefona, preko laptopova do personalnih računara koji imaju monitore različitih dimenzija. Pored toga interfejs treba biti tako dizajniran da zajedno sa adaptivnih dizajnom prave (eng. user friendly) dobro korisničko iskustvo. Da se objekti na platformi jasno vide kako korisnici bez greške mogu interagovati sa sistemom uz minimalan broj grešaka na raznim uređajima. Mora izgledati atraktivno kako bi korisnici zadržali.

3.4.2 Performances

Platforma mora biti veoma brza. i u mogućnosti da opsluži veliki broj korisnika, najmanje 100 korisnika istovremeno. Pored toga, ne sme se dozvoliti korisnicima čekanje na prikazivanje sadržaja dužeg od 1 sekunde. Baza će se jednom dnevno ažurirati kako ne bi došlo do usporavanja ili grešaka u sistemu. Korisniku se ne sme dozvoliti da vidi blanko stranicu, pa će se sve skripte ubacivati na sam kraj koda, kako bi korisnik prvo video barem neki sadržaj a onda vremenom kako se budu i same skripte renderovale u pretrađivaču doći će do potpuno vidljivog sadržaja.

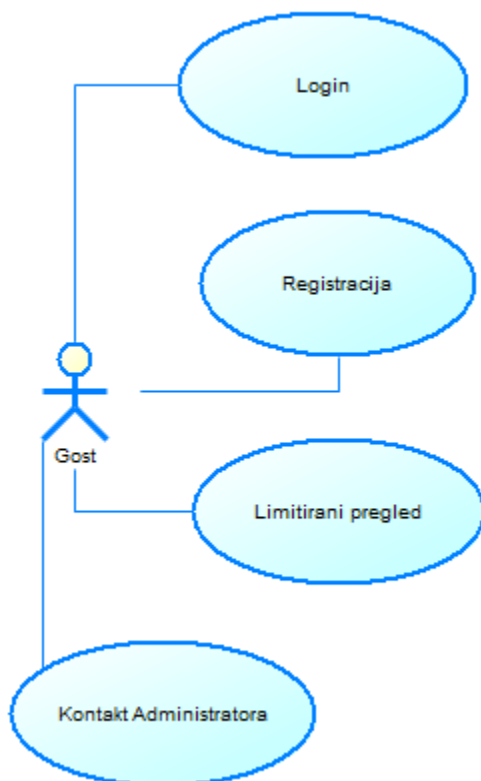
3.4.3 Security

Bezbednost se mora držati na jako visokom nivou. Sistem će imati enkriptovane šifre (md5 hašom) kako ne bi došlo do zloupotrebe, ili ukoliko na neki način dođe do probijanja u bazu podataka korisnici će biti zaštićeni. Sistem onemogućava SQL injection napade tako što će se koristiti bezbedni metodi tokom kodiranja i povezivanja sa bazom podataka (Php PDO). Takođe maliciozni korisnici će biti svedeni na minimum i neće se moći ubaciti maliciozni karakteri ili sadržaj ni u jednu formu u kojoj dozvoljavamo korisniku unos.

3.5 SLUČAJEVI KORIŠĆENJA

KORISNIK (GOST):

Gost, korisnik koji nije ulogovan/registrovan, ima mogućnost pregleda limitiranog sadržaja, i ne može videti detaljno informacije ni o filmovima ni o serijama, limitiran je na nekoliko opcija, može kontaktirati administratora ukoliko naiđe na neki problem, i može se ulogovati ili registrovati na sistem.



Slika 4 – Use Case diagram - gost

Use Case	Preduslovi	Koraci	Izuzetci	Postuslovi
Login(Prijavljivanje)	Gost mora imati korisnički nalog.	Gost popunjava formu za prijavljivanje	Gost nema nalog za prijavljivanje na sistem	Gost je uspešno izvršio prijavljivanje na sistem
Registracija	Gost mora imati email i internet	Gost popunjava formu za registraciju	Korisničko ime ili email adresa već postoje u sistemu.	Gost je uspešno napravio nalog
Ograničen pregled	Gost mora imati pristup internetu.	Gost vrši pregled ograničene liste filmova/serija sa ograničenim prikazom sadržaja	Gost je ulogovan na sistem pa više nije gost nakon čega vidi ceo sadržaj	Gost može videti samo ograničenu listu sadržaja sa ograničenim opisom istog
Kontakt Administratora	Gost mora imati pristup internetu	Gost popunjava formu za kontakt administratora	Gost nije popunio celu formu	Gost je kontaktirao administratora

KORISNIK:

Korisnik se registruje na sistem tako što ostavlja svoje osnovne podatke i bira jedinstveno korisničko ime koje već ne postoji i sigurnosnu šifru, takođe jedan korisnički nalog je dozvoljen po email adresi.

Korisnik se može ulogovati na sistem kako bi koristio samu platformu, bez toga je običan, gost-korisnik.

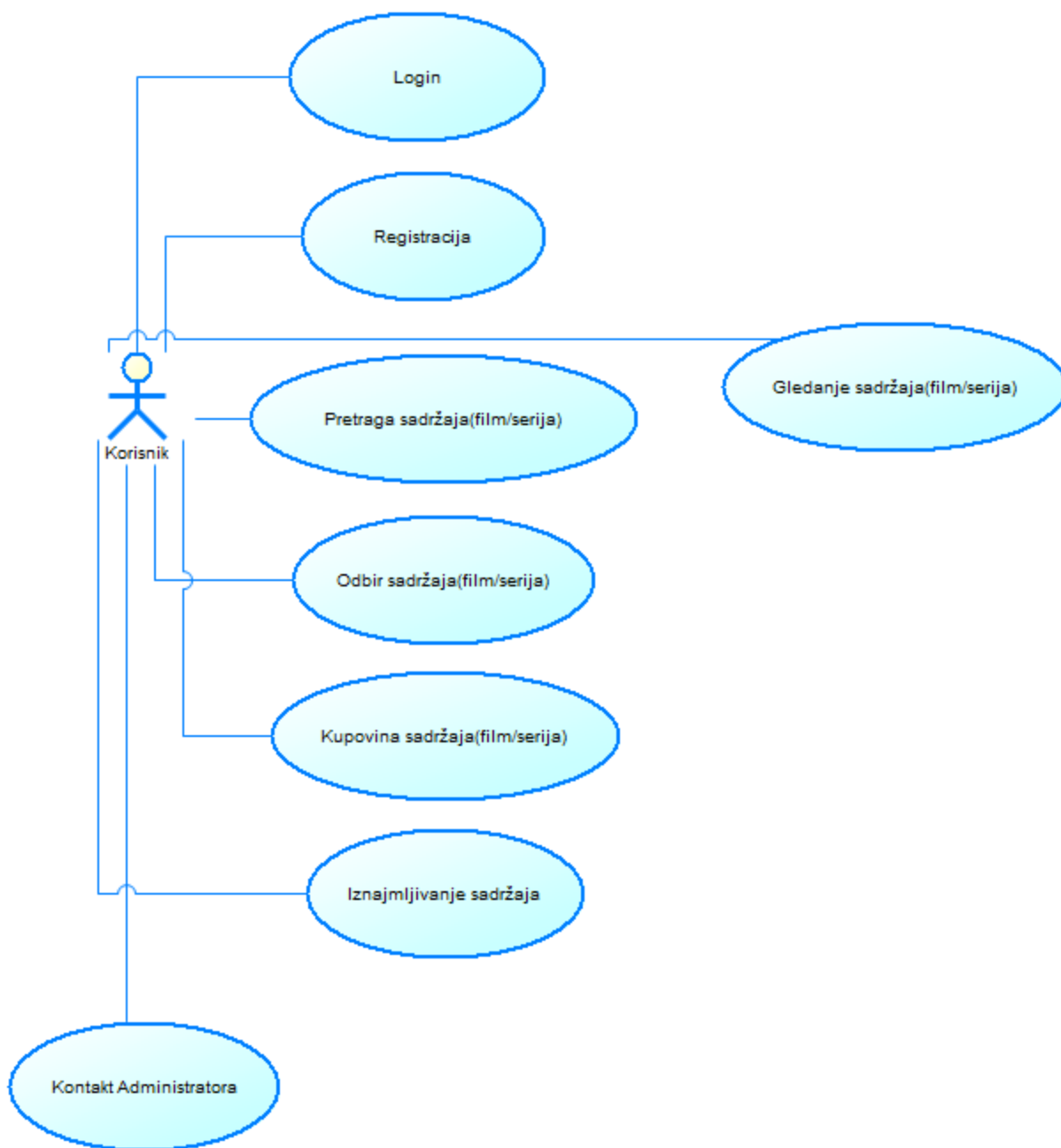
Korisnik ima mogućnost pretrage sadržaja (filmova/serija), ili putem korišćenja ulaznih uređaja ili putem opcije pretraži (search).

Korisnik može odabrati željeni sadržaj i videti detaljne informacije o sadržaju (filmu ili seriji).

Korisnik potom ima mogućnost iznajmljivanja sadržaja koji se vremenski ograničeno zadržava na korisničkom nalogu, posle čega zauvek nestaje.

Korisnik može kupiti sadržaj koji će zauvek biti na tom korisničkom nalogu i može ga pregledati (gledati) u bilo kom trenutku, bilo gde.

Korisnikov glavni slučaj korišćenja je samo gledanje sadržaja, dakle ukoliko je korisnik iznajmio ili kupio film ili seriju on može pristupiti gledanju istog.



Slika 5 – Use Case diagram – Korisnik

Use Case	Preduslovi	Koraci	Izuzetci	Postuslovi
Prijavljivanje(login)	Korisnik mora imati korisnički nalog	Korisnik popunjava formu za prijavljivanje na sistem	Korisnik nema nalog za prijavljivanje na sistem	Korisnik je uspešno prijavljen na sistem
Registracija	Korisnik mora imati email I internet	Korisnik popunjava formu za registraciju	Korisničko ime ili email adresa već postoje u sistemu	Korisnik je uspešno napravio nalog
Pretraga sadržaja(filмова/serija)	Korisnik mora biti prijavljen na sistem	Korisnik vrši pregled celog sadržaja iz baze podataka u okviru stranice za prikaz sadržaja	Korisnik mora biti prijavljen na sistem	Korisnik može videti detaljno listu celog sadržaja koja se direktno ispisuje iz baze podataka
Odabir sadržaja	Korisnik mora biti prijavljen na sistem	Korisnik iz liste bira sadržaj koji želi gde vidi detaljan opis odabranog sadržaja	Korisnik mora biti prijavljen na sistem	Korisnik može odabrati željeni sadržaj
Kupovina sadržaja	Korisnik mora biti prijavljen na sistem	Korisnik bira opciju kupi sadržaj	Korisnik mora biti prijavljen na sistem	Korisnik je kupio sadržaj koji se smešta u deo aplikacije vezan sa njegov nalog
Iznajmljivanje sadržaja	Korisnik mora biti prijavljen na sistem	Korisnik bira opciju iznajmi sadržaj	Korisnik mora biti prijavljen na sistem	Korisnik je iznajmio sadržaj I može ga gledati na svom nalogu naredna 24h
Gledanje sadržaja	Korisnik mora biti prijavljen na sistem	Korisnik u delu aplikacije moj sadržaj bira sadržaj koji želi gledati	Korisnik mora biti prijavljen na sistem	Korisnik može odabrati sadržaj koji se nalazi u delu aplikacije moj sadržaj I koji je vezan za njegov nalog kako bi ga gledao
Kontakt administratora	Korisnik mora imati pristup internetu	Korisnik popunjava formu za kontakt	Korisnik nije popunio validno formu za kontakt	Korisnik je kontaktirao administratora

ADMINISTRATOR:

Administrartor ima mogućnost prijavljivanja na sistem

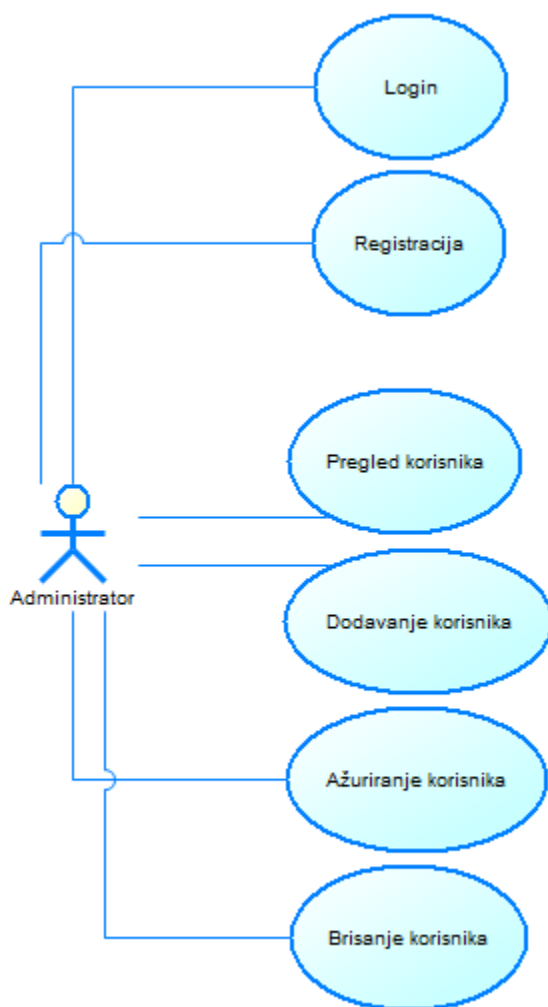
Administrator ima mogućnost registrovanja novih administratora na sistem
(Registracija običnog korisnika pa dodavanje privilegije tj. role, a i dodavanje novog korisnika je zapravo registracija)

Administrator ima mogućnost pregleda svih korisnika sistema.

Administrator može ažurirati korisnike iz bilo kog raloga (dodeljivanje privilegija, blokiranje, dodavanje sadržaja ukoliko je došlo do greške pri kupovini...)

Administrator može dodavati nove korisnike, ukoliko postoji potreba za tim.

Administrator može brisati postojeće korisnike iz sistema.

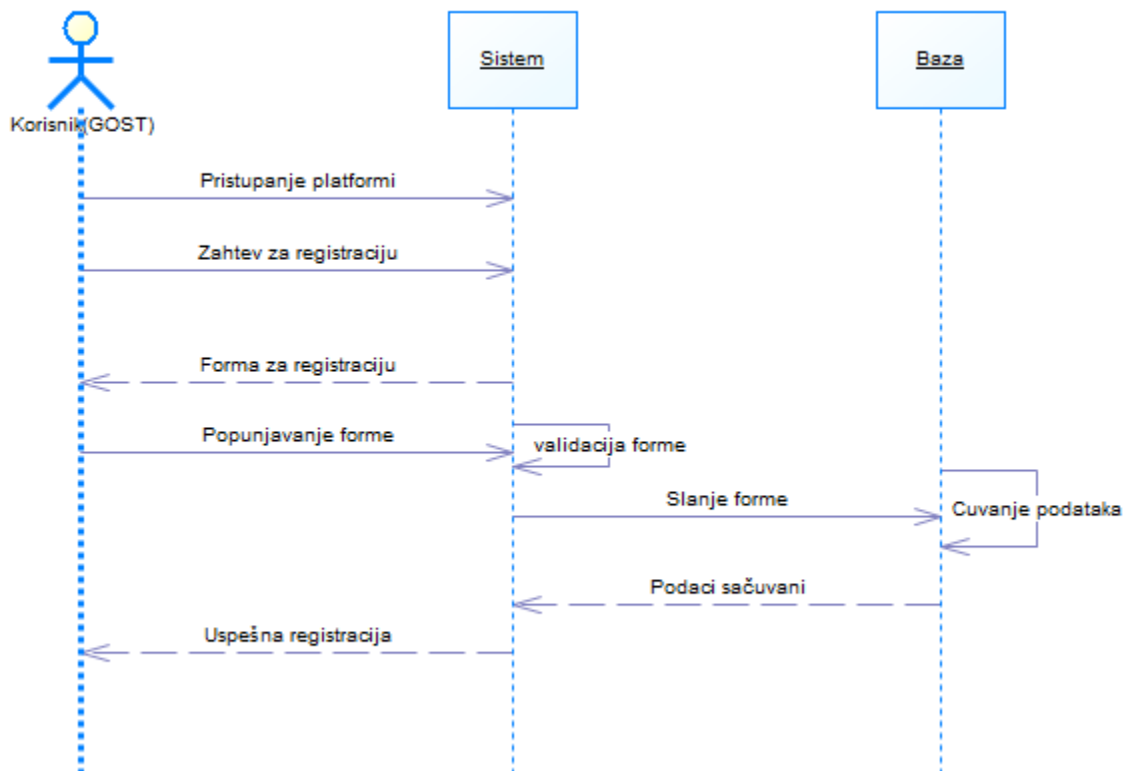


Slika 6 – Use Case dijagram – Administrator

Use Case	Preduslovi	Koraci	Izuzeci	Postuslovi
Prijavljivanje(login)	Administrator mora biti prijavljen na sistem	Administrator popunjava formu za prijavljivanje	Administrator nema korisnički nalog, ili nalog sa privilegijama	Administrator je uspešno prijavljen na sistem
Registracija	Korisnik mora imati internet konekciju I email adresu	Korisnik popunjava formu za registraciju	Korisničko ime ili email adresa već postoje u sistemu	Korisnik je uspešno napravio korisnički nalog
Pregled korisnika	Administrator mora biti prijavljen na sistem.	Administrator otvara deo aplikacije za pregled svih korisnika	Korisnik nema administratorske privilegije.	Administrator dobija listu svih korisnika iz baze podataka
Dodavanje korisnika	Administrator mora biti prijavljen na sistem	Administrator otvara deo za dodavanje korisnika I popunjava formu za dodavanje novog korisnika	Korisnik nema administratorske privilegije	Novi korisnik je dodat (korisnički nalog)
Ažuriranje korisnika	Administrator mora biti prijavljen na sistem	Administrator popunjava formu novim podacima, koji ažuriraju korisnički nalog	Korisnik nema administratorske privilegije. Pogrešno popunjena forma	Korisnički nalog je ažuriran
Brisanje korisnika	Administrator mora biti prijavljen na sistem	Administrator bira korisnički nalog koji želi obrisati	Korisnik nema administratorske privilegije. Pogrešno odabran korisnik. Pokušaj brisanja administratora	Korisnički nalog je obrisani

3.6 APSTRAKTNI DIJAGRAM SEKVENCI (SISTEM SEKVENCIJALNI DIJAGRAM)

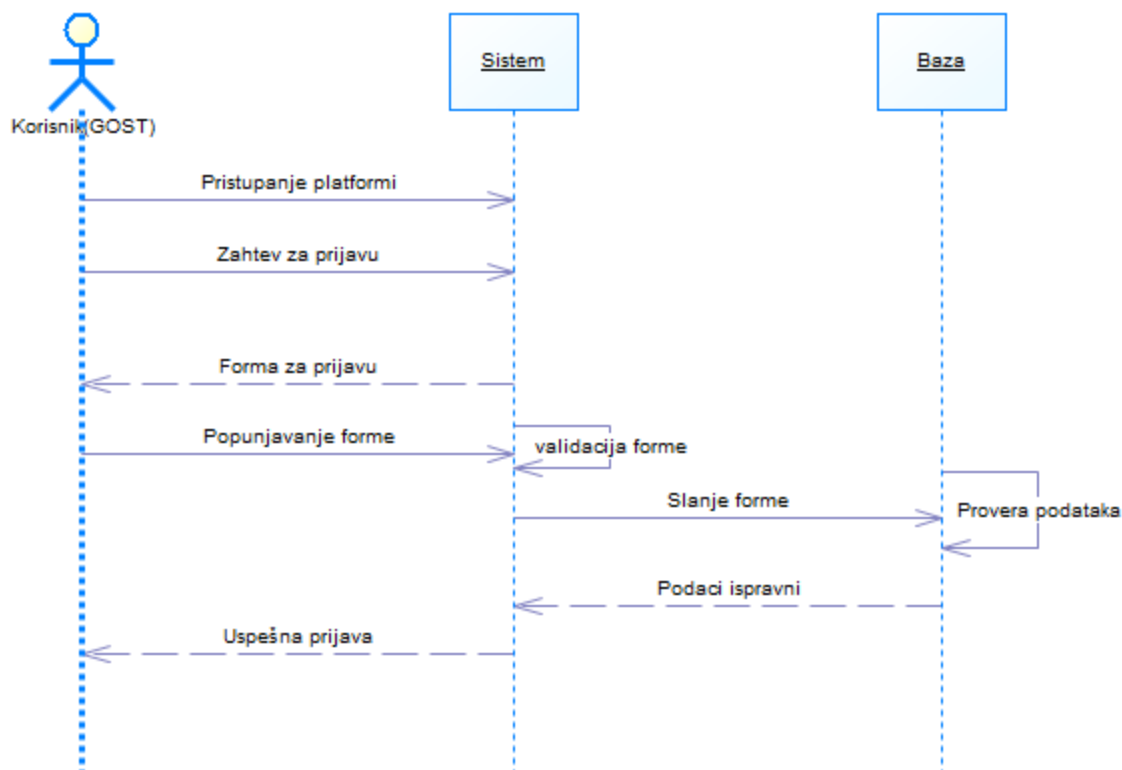
3.6.1 Dijagram sekvenci za registrovanje korisnika (gosta)



Slika 7 – SSD – Registracija gosta

Kada gost pristupi sistemu, otvara stranicu za registrovanje, sistem onda korisniku šalje formu, nakon čega korisnik popunjava tu formu, gde se posle pritiska na dugme registruj vrši validacija unetih podataka. Ukoliko je forma dobra, šalje se ka bazi podataka, gde se čuvaju podatci. Baza šalje odgovor o uspešnoj registraciji sistemu, nakog čega sistem vraća korisniku poruku o uspešnoj registraciji.

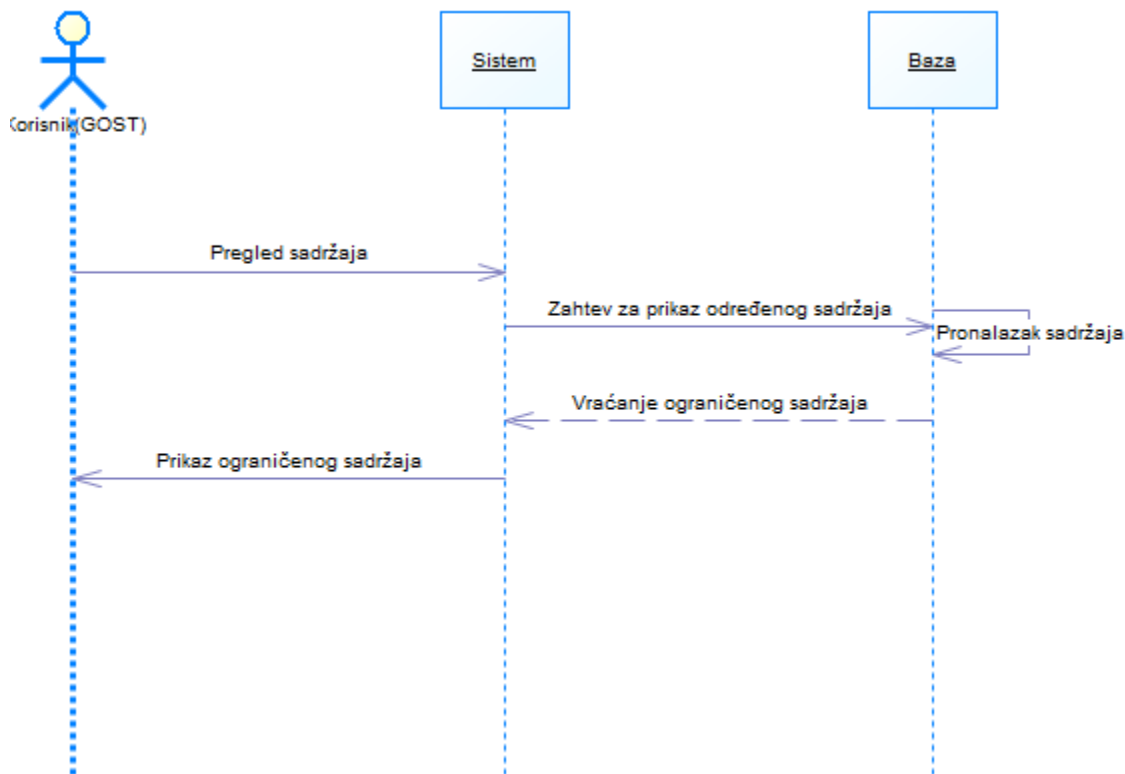
3.6.2 Dijagram sekvenci za prijavu na sistem – GOST



Slika 8 – SSD Prijavljivanje gosta

Gost pristupa sistemu, I ukoliko ima nalog, otvara formu za prijavljivanje na sistem. Sistem proverava formu, I ukoliko je ispravna, proverava sa bazom da li su podatci validni, ukoliko je sve u redu, baza vraća sistemu potvrdnu poruku, nakon čega sistem prijavljuje korisnika, I vraća poruku o uspšnom prijavljivanju na sistem.

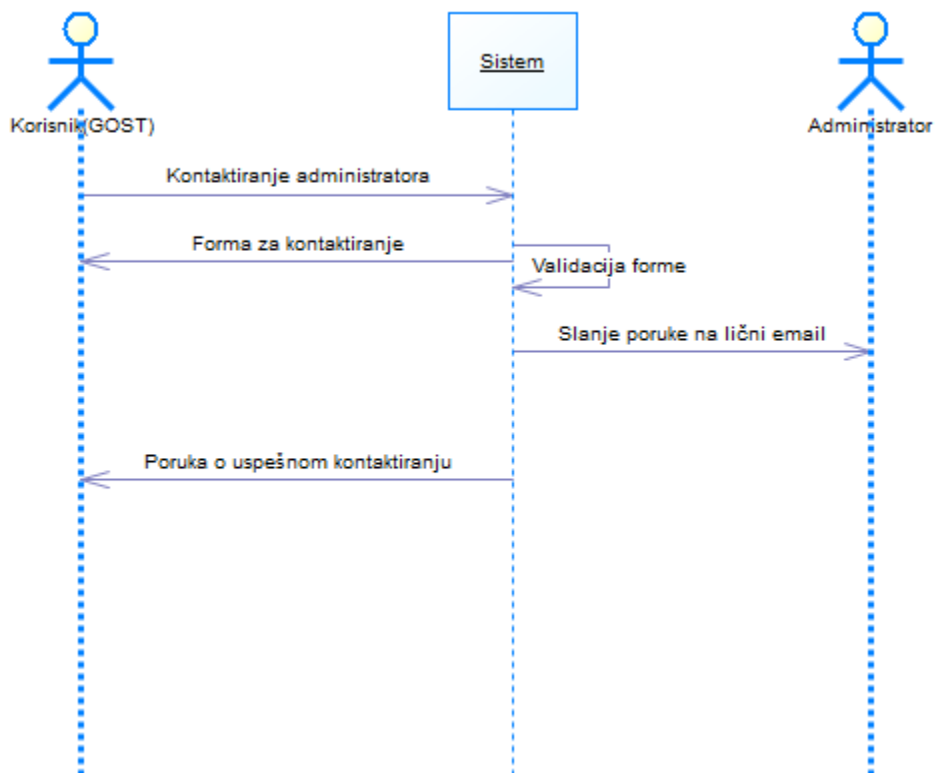
3.6.3 Dijagram sekvenci za pregled limitiranog (određenog) sadržaja – GOST



Slika 9 – SSD ograničen sadržaj za gosta

Gost može otvoriti stranicu za prikaz sadržaja, nakon čega sistem (pošto je gost, nije prijavljen na sistem) šalje zahtev bazi za prikaz određenog sadržaja za gosta (10 nasumično odabranih filmova/serija, sa prikazom osnovnih podataka o istim). Baza vraća sadržaj, nakon čega sistem vrši određene (estetske) izmene i prikazuje takav sadržaj korisniku.

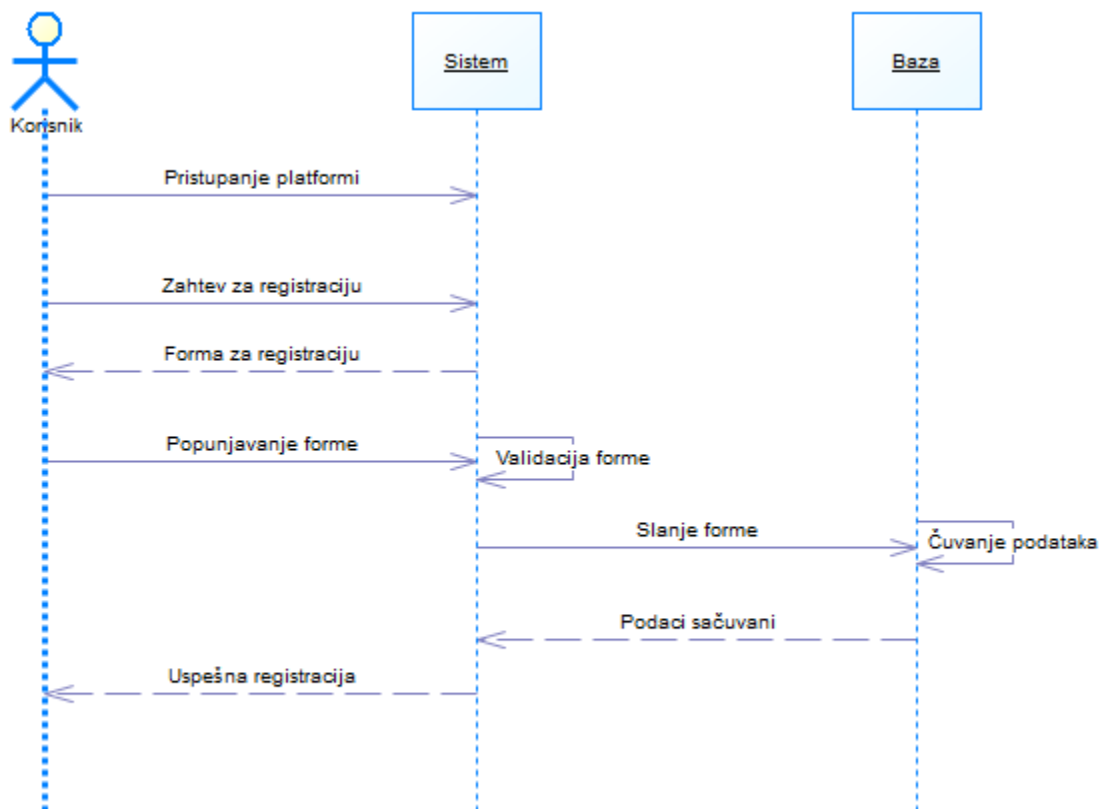
3.6.4 Dijagram sekvenci za kontaktiranje administratora - GOST



Slika 10 – SSD – Kontaktiranje administrator

Svaki gost, ukoliko mu zatreba pomoć može otići na stranu za kontaktiranje administratora, nakog čega dobija formu za kontakt, posle koje, ukoliko je validacija u redu, poruka se prosledi administratoru na njegovu mail adresu. Sistem potom šalje povratnu poruku korisniku o uspesno poslatoj poruci.

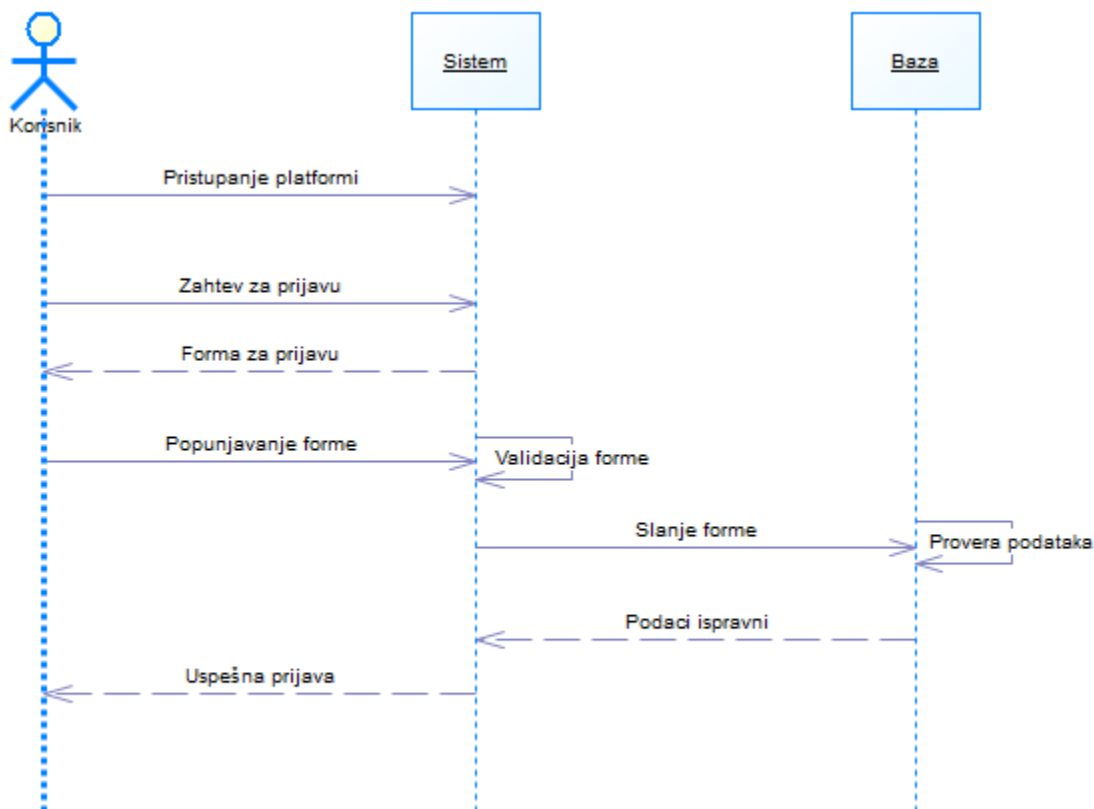
3.6.5 Dijagram sekvenci za registraciju – KORISNIK



Slika 11 – SSD – Registracija korisnika

Slično kao gost, korisnik koji već ima nalog može odabrati da registruje novi, gde se isti koraci dešavaju. Prvo se zahteva forma koju sistem vrati, potom korisnik popunjava formu, pa se vrši validacija. Nakon toga, forma se šalje u bazu gde se poruka vraća sistemu o uspešno upisanim podacima, a dalje korisniku.

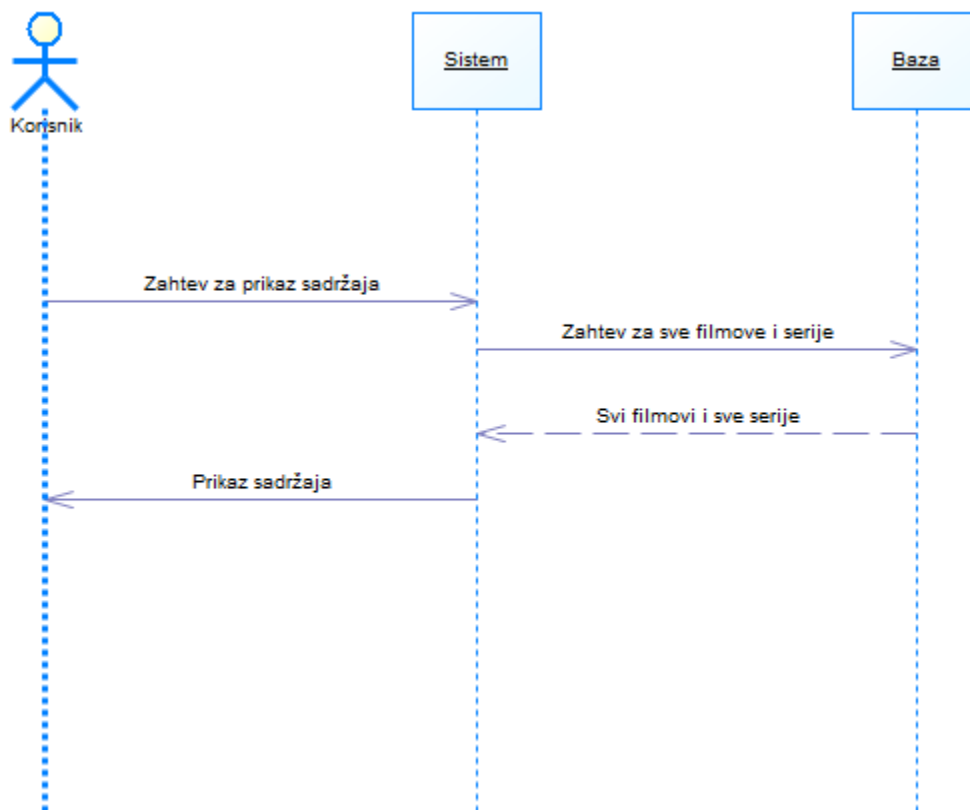
3.6.6 Dijagram sekvenci prijavljivanja na sistem – KORISNIK



Slika 12 – SSD – Prijava korisnika

Korisnik se prijavljuje na sistem, tako što sistem vrati formu, a nakon popunjavanja forme od strane korisnika sistem vrši validaciju. Ukoliko je validna, sistem vrši proveru podataka sa bazom, I ukoliko baza vrati da su podatci ispravni, sistem dalje šalje korisniku poruku o prijavi na sistem, I vrši prijavu korisnika na isti

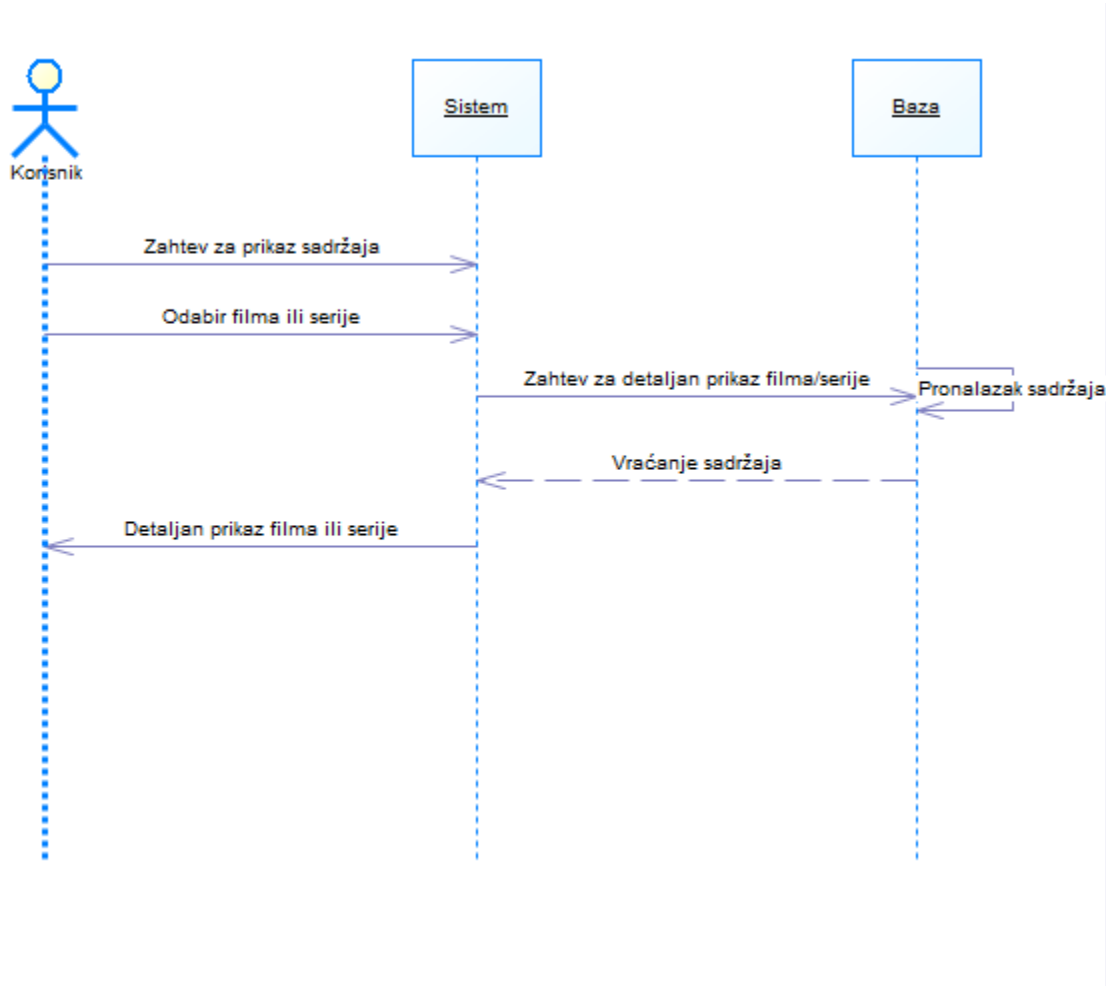
3.6.7 Dijagram sekvenci za pretragu sadržaja - KORISNIK



Slika 13 – SSD – Prikaz sadržaja

Korisnik odlazi na stranicu za prikaz sadržaja, nakon čega sistem šalje bazi zahtev za prikaz svih filmova i serija sa njihovim opisom. Baza vraća sistemu takve podatke, sistem vrši obradu (estetski parsira i prikazuje podatke čitljivim za korisnika), i vraća korisniku tako obrađen sadržaj.

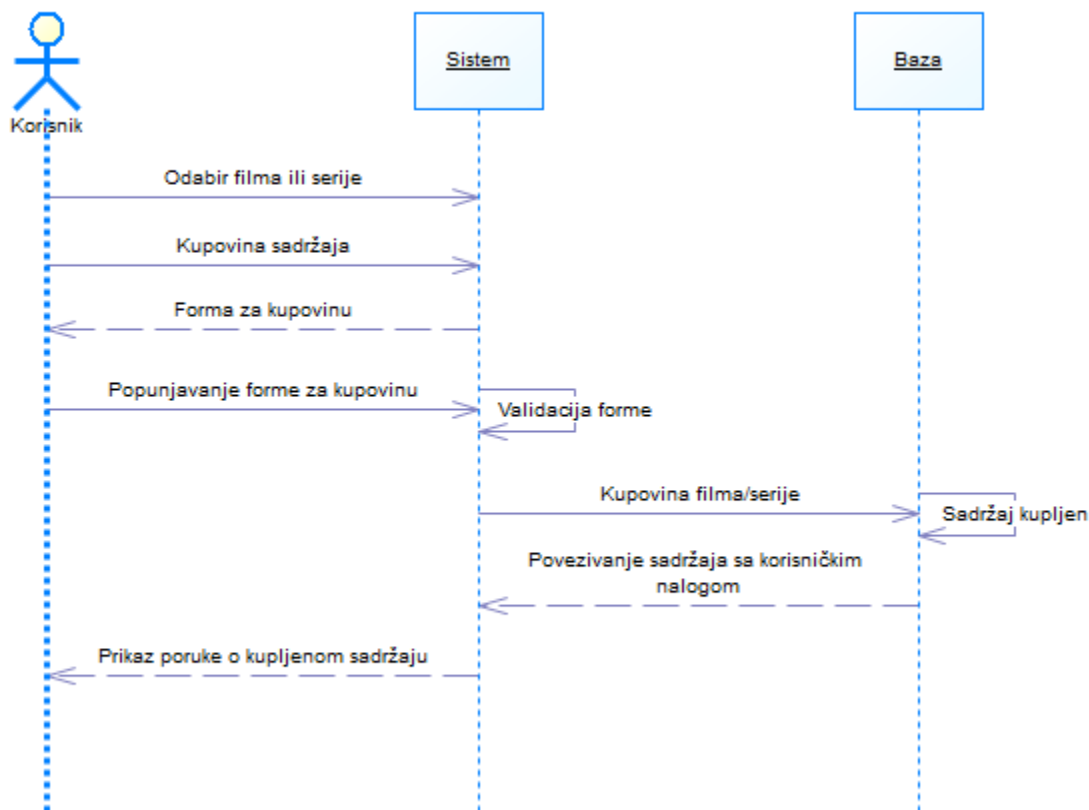
3.6.8 Dijagram sekvenci za odabir sadržaja – KORISNIK



Slika 14 – SSD – Odabir sadržaja

Korisnik iz liste sadržaja bira jedan film ili seriju, kada odabere da otvori taj sadržaj, šalje zahtev sistemu za taj isti sadržaj, sistem potom šalje zahtev bazi. Baza pronalazi određeni film ili seriju, vraća sistemu podatke o istom, nakon čega sistem obrađuje te podatke i prikazuje ih korisniku.

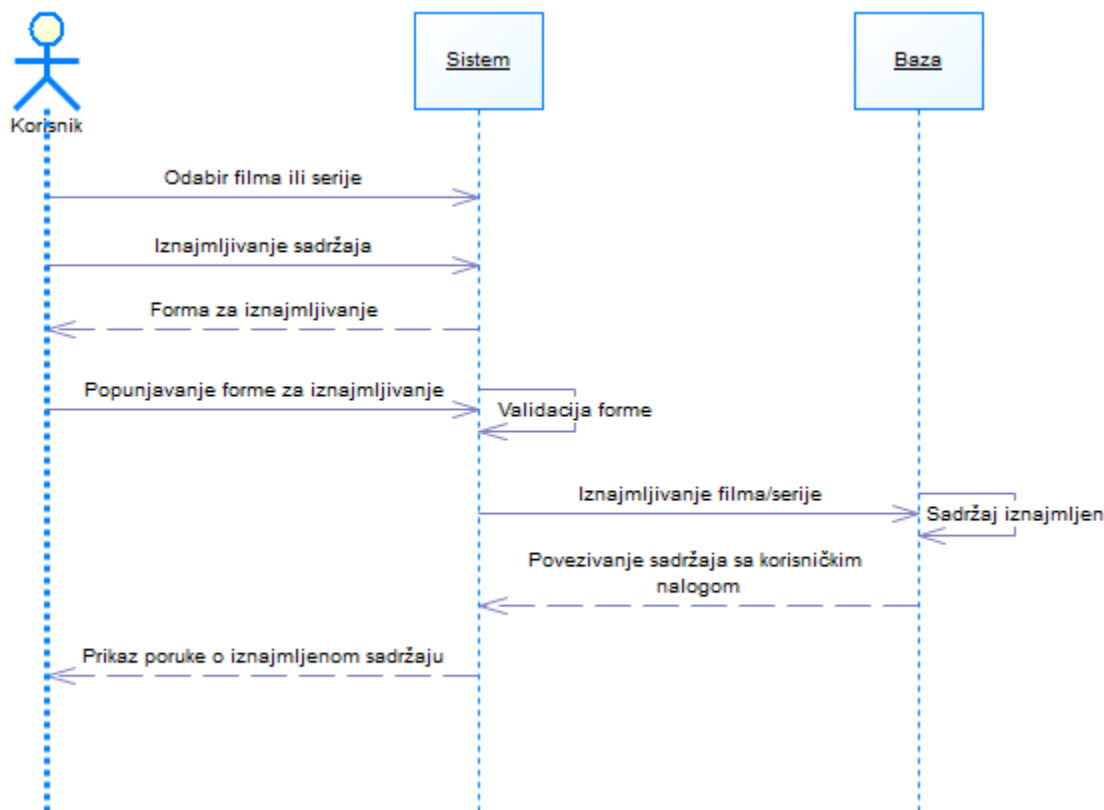
3.6.9 Dijagram sekvenci kupovine sadržaja – KORISNIK



Slika 15 – SSD – Kupovina sadržaja

Korisnik iz liste bira konkretan film ili seriju (sadržaj), nakon toga, bira opciju za kupovinu sadržaja. Sistem korisniku vraća formu u kojoj upisuje podatke sa svoje kreditne kartice, i nakon što se izvrši validacija o verodostojnosti podataka i kartice, sistem šalje bazi zahtev za kupovinu filma/serije. Baza pronalazi sadržaj i povezuje identifikacioni broj filma/serije sa identifikacionim brojem korisničkog naloga u određenu tabelu. Korisnik dobija povratnu poruku o kupljenom filmu ili seriji, nakon čega može pristupiti gledanju sadržaja kroz određenu stranu na platformi.

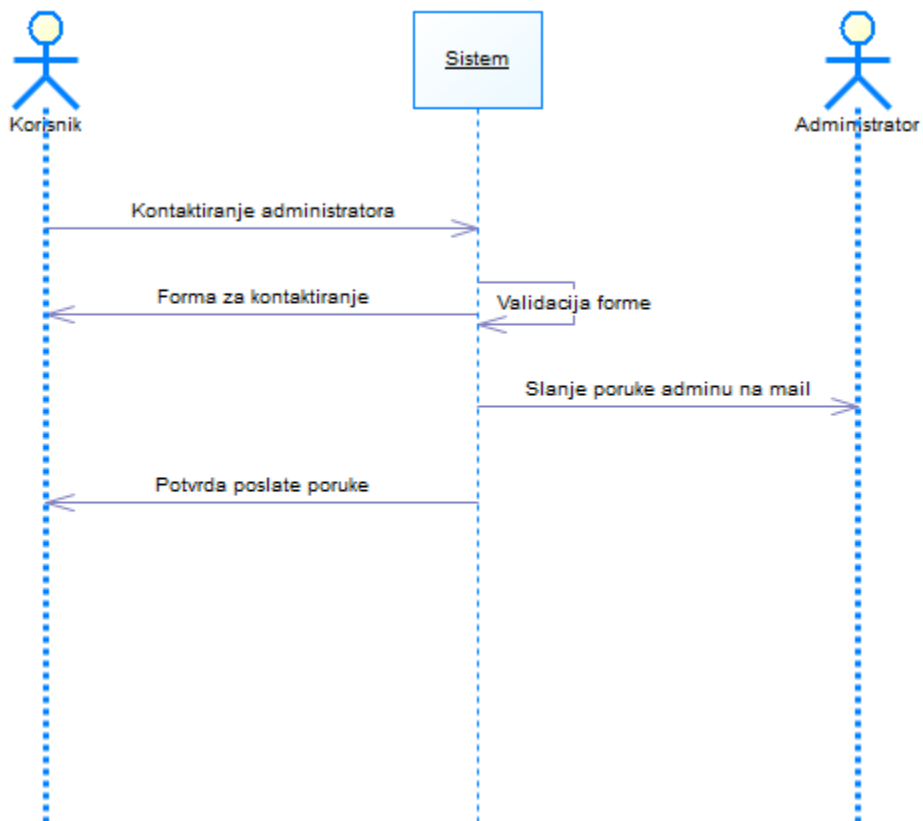
3.6.10 Dijagram sekvenci iznajmljivanja sadržaja – KORISNIK



Slika 16 – SSD – Iznajmi sadržaj

Korisnik bira seriju ili film iz liste sadržaja, nakon čega bira iznajmi (rent). Sistem onda vraća korisniku formu za iznajmljivanje (kupovinu), vrši se validacija popunjene forme, i dalje sistem i baza povezuju sadržaj sa korisnikom u tabelu za iznajmljivanje sadržaja. Baza vraća povratnu poruku sistemu o uspešnosti, nakon čega sistem prikazuje sadržaj u određenom delu platforme za korisnika, te mu vraća povratnu poruku o iznajmljenom sadržaju, i vremenu dostupnosti za gledanje (24h).

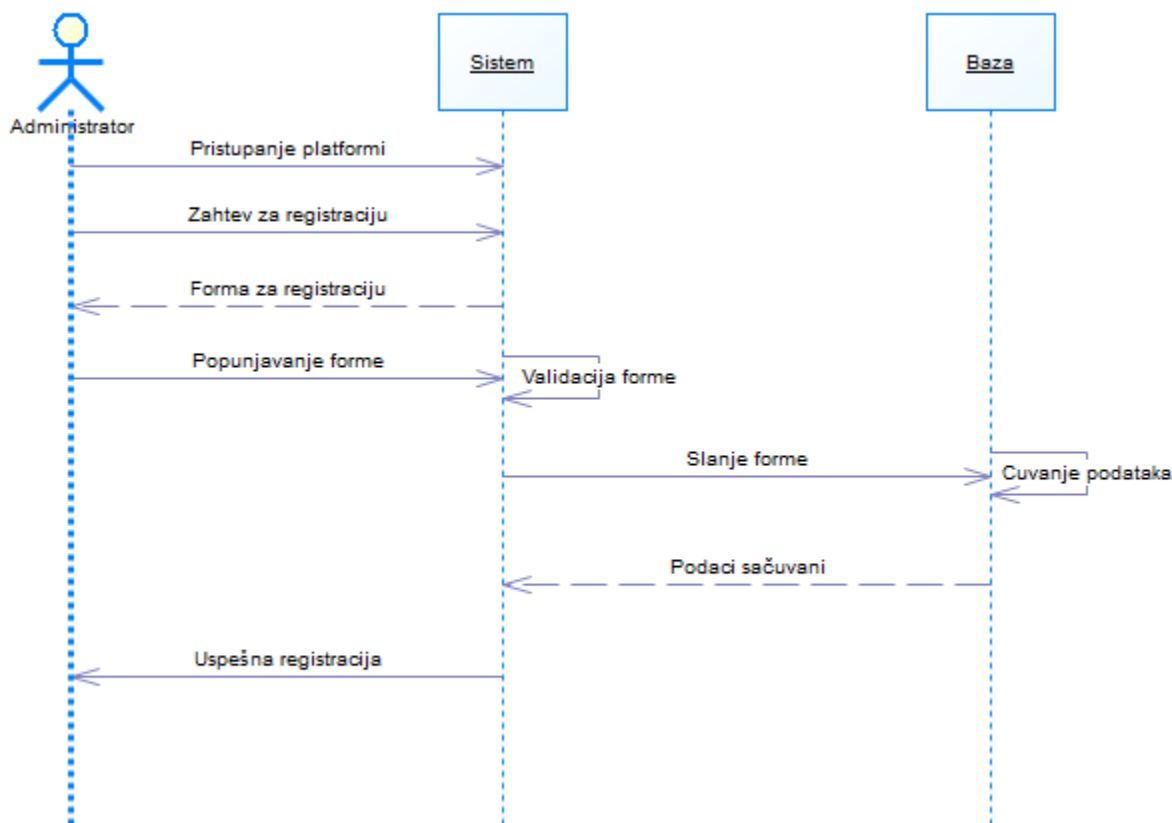
3.6.11 Dijagram sekvenci za kontakt administratora – KORISNIK



Slika 17 – SSD – Kontaktiranje admina

Korisnik ima mogućnost I kada je ulogovan I kada nije, da kontaktira administratora u vezi bilo kakvog problema. U konkretnom slučaju kada je korisnik ulogovan, na dugme kontakt, od sistema dobija formu, koja, kada se popuni I izvrši validacija, biva dalje poslata administratoru. Administrator poruku dobija na mail, dok korisnik dobija povratnu poruku od sistema, da je poruka uspešno poslata.

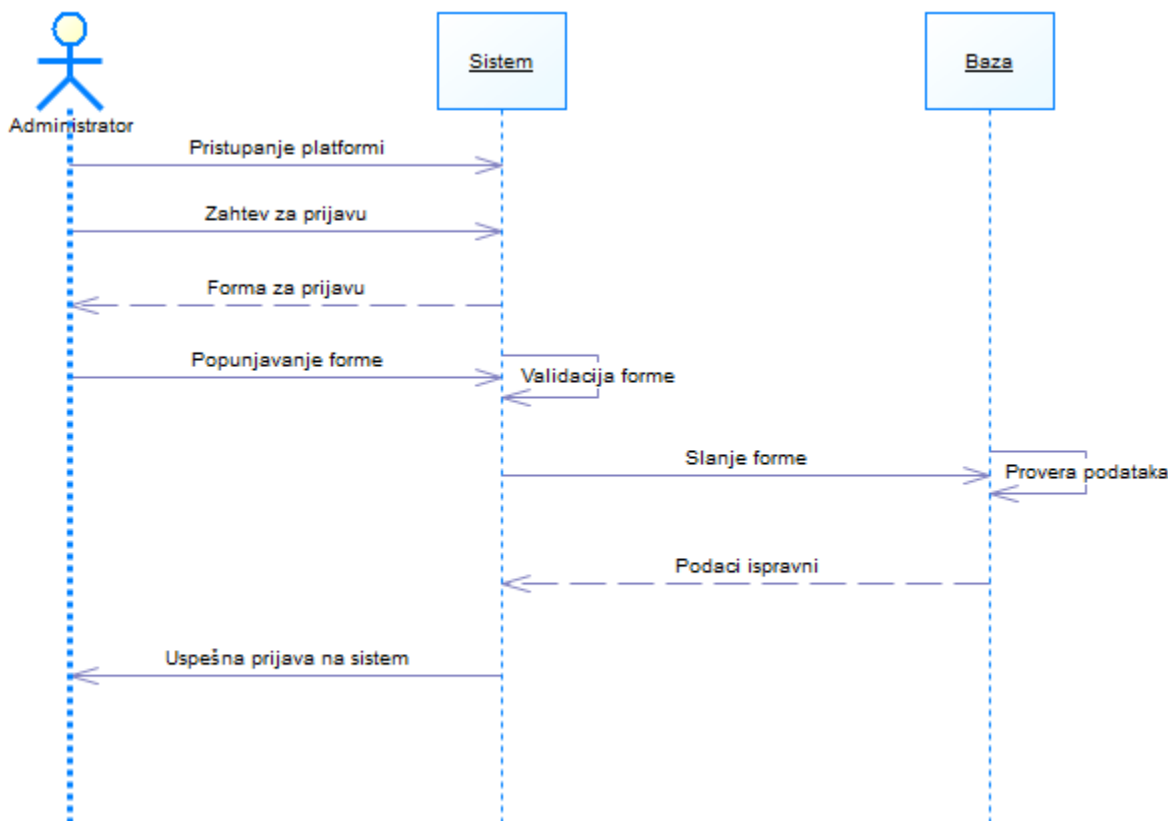
3.6.12 Dijagram sekvenci za registraciju na sistem – Administrator



Slika 18 – SSD – Registracija administrator

Registracija administratora se odvija na potpuno isti način kao i ostale registracije, dakle, korisnik pravi nalog. Jedina razlika je što će mu kasnije drugi administrator promeniti ulogu iz korisnika u administrator. Dakle, korisnik u tom slučaju šalje zahtev sistemu za stranicu registracije, sistem vraća formu, forma mora biti popunjena, vrši se validacija, nakon čega se šalje forma u bazu. Baza se popunjuje tim podacima, i vraća povratnu poruku sistemu o uspešnosti registracije nakon čega sistem dalje vraća korisniku poruku o uspešnoj registraciji.

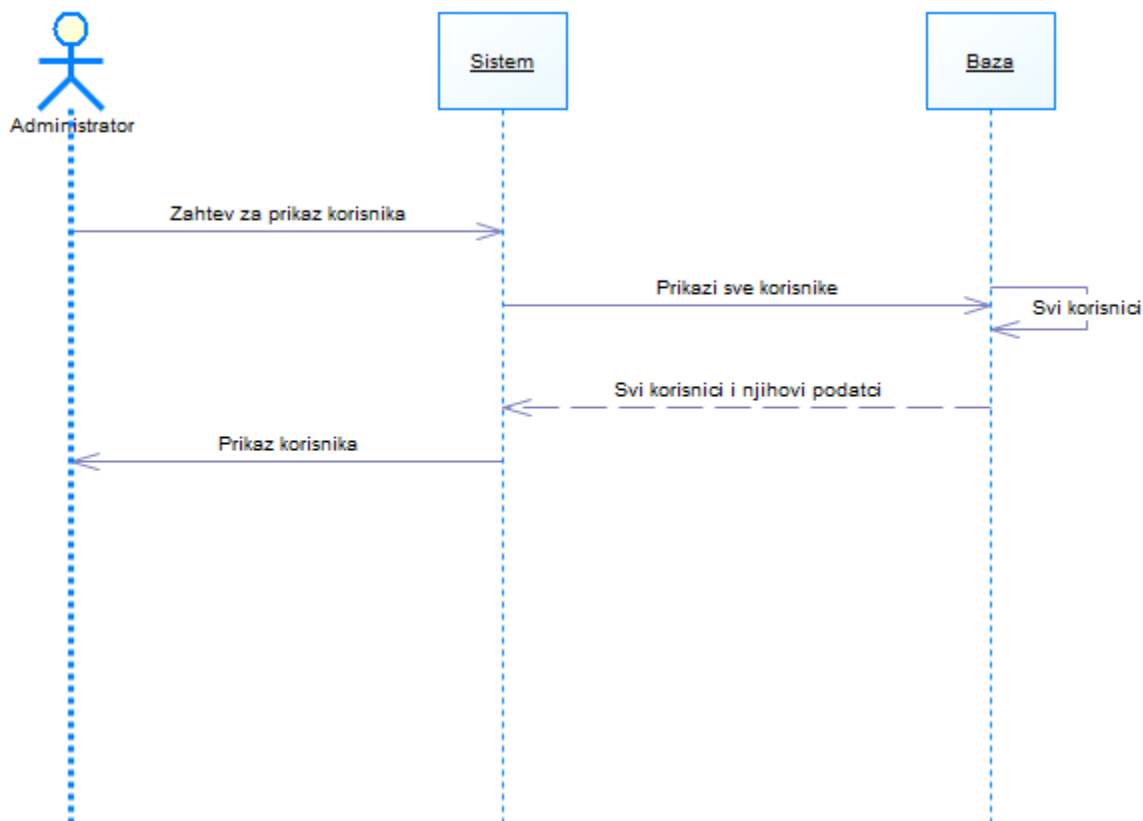
3.6.13 Dijagram sekvenci za prijavljivanje na sistem – Administrator



Slika 19 – SSD – Prijava administrator

Administrator se prijavljuje na sistem, kao i ostali korisnici, jedina razlika je u tome što sistem proverava ulogu (rolu), i ukoliko je ona podešena na administrator, onda se tom korisniku prikazuju drugačije stranice, u odnosu na običnog korisnika. Pored toga administrator ima sve funkcije kao i korisnik, jer i on sam može biti korisnik sistema. Međutim ne može sam sebi dodavati sadržaj bez validne kupovine ili iznajmljivanja. Dakle, administrator šalje zahtev sistemu za prijavu, sistem vraća formu, koju ovaj popunjava, te se vrši validacija forme, nakon čega se forma šalje u bazu podataka na proveru tih podataka, i ukoliko su administratorski kredencijali ispravni, baza šalje potvrdu da to zaista jeste administrator, nakon čega je administrator uspešno prijavljen na sistem.

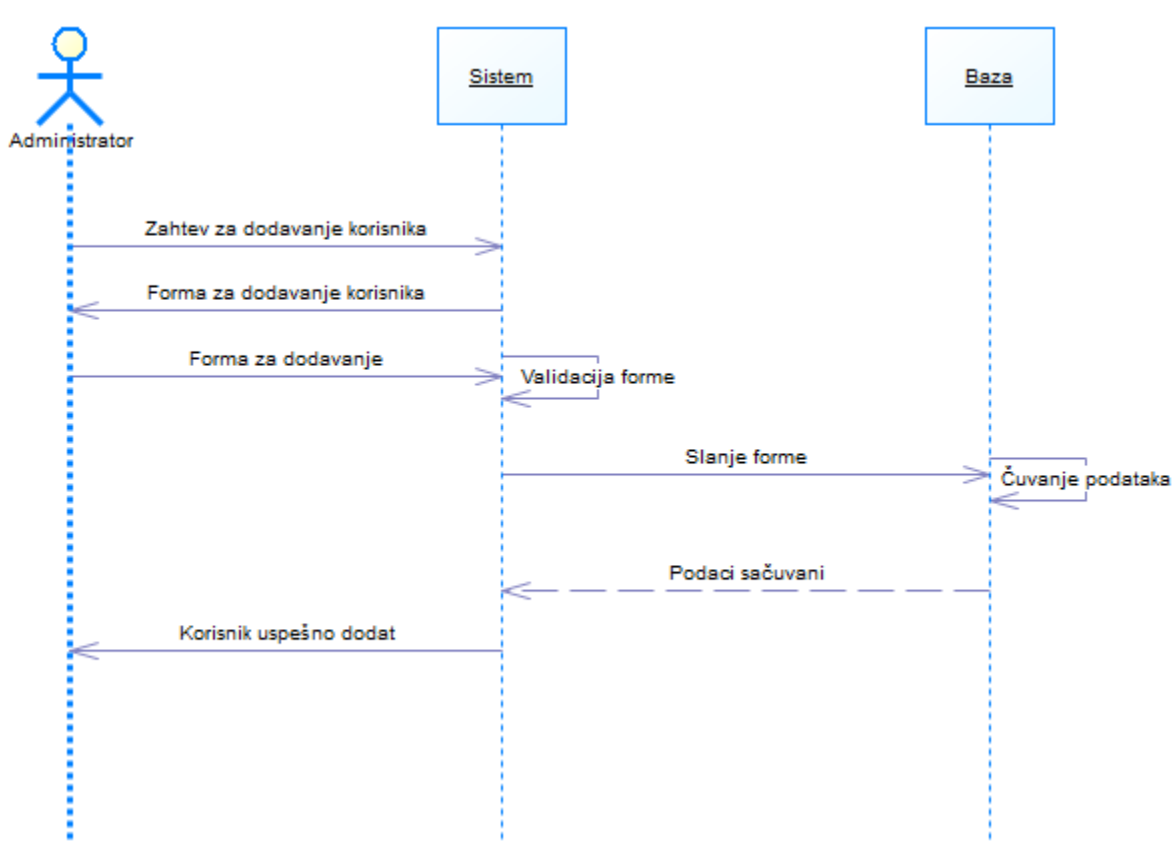
3.6.14 Dijagram sekvenci za pregled korisnika – Administrator



Slika 20 – SSD – Prikaz korisnika

Administrator pristupa delu aplikacije za prikaz svih korisnika. Sistem traži od baze da prikaže sve korisnike, baza pronalazi sve korisnike kao i njihove podatke koje je zahtevao administrator (ime, prezime, mail, korisničko ime – bez šifre i senzitivnih podataka), vraća te podatke sistemu, sistem ih parsira i prikazuje administratoru.

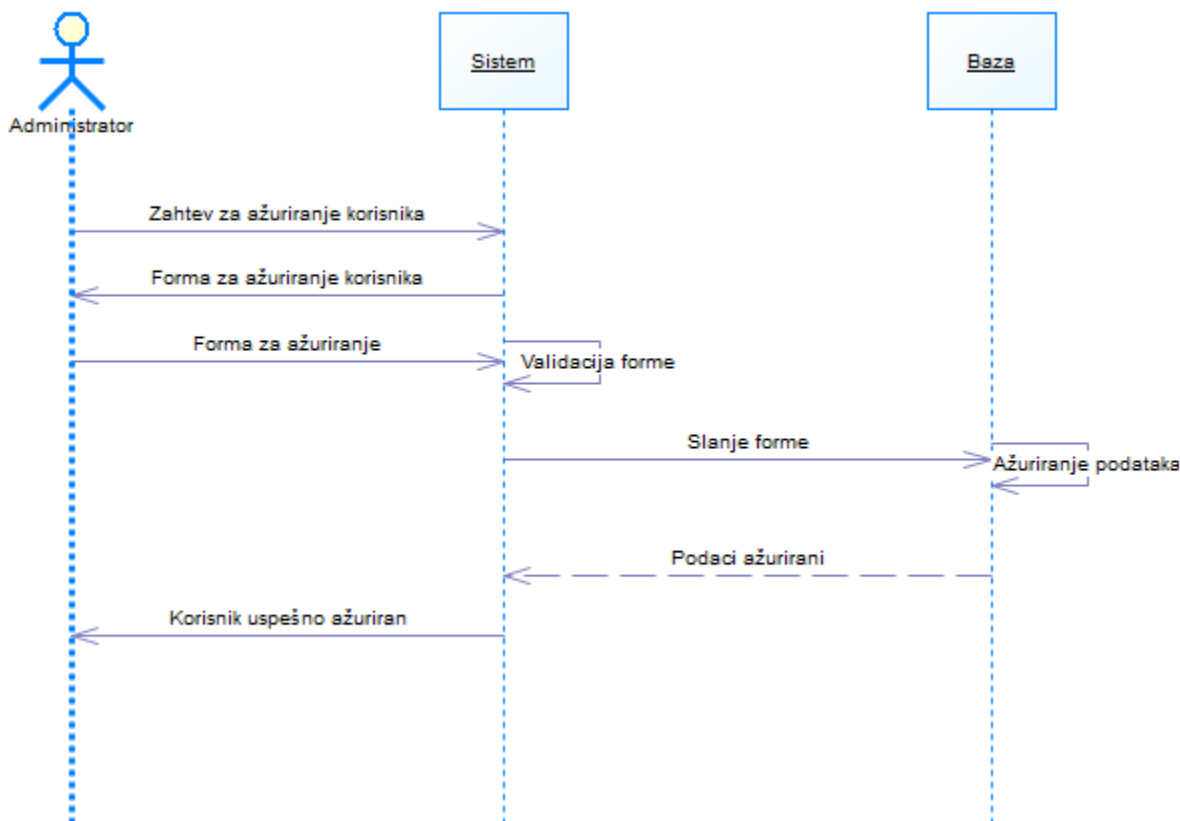
3.6.15 Dijagram sekvenci za dodavanje korisnika – Administrator



Slika 21 – SSD – Dodavanje korisnika

Dodavanje korisnika nije ništa drugo do sama registracija novog naloga, dakle, administrator će dobiti formu za registraciju kada pošalje zahtev sistemu za dodavanje novog korisnika. Nakon validacije podatci se šalju u bazu, baza čuva podatke, i vraća povratnu poruku administratoru o uspešnom kreiranju novog korisničkog naloga.

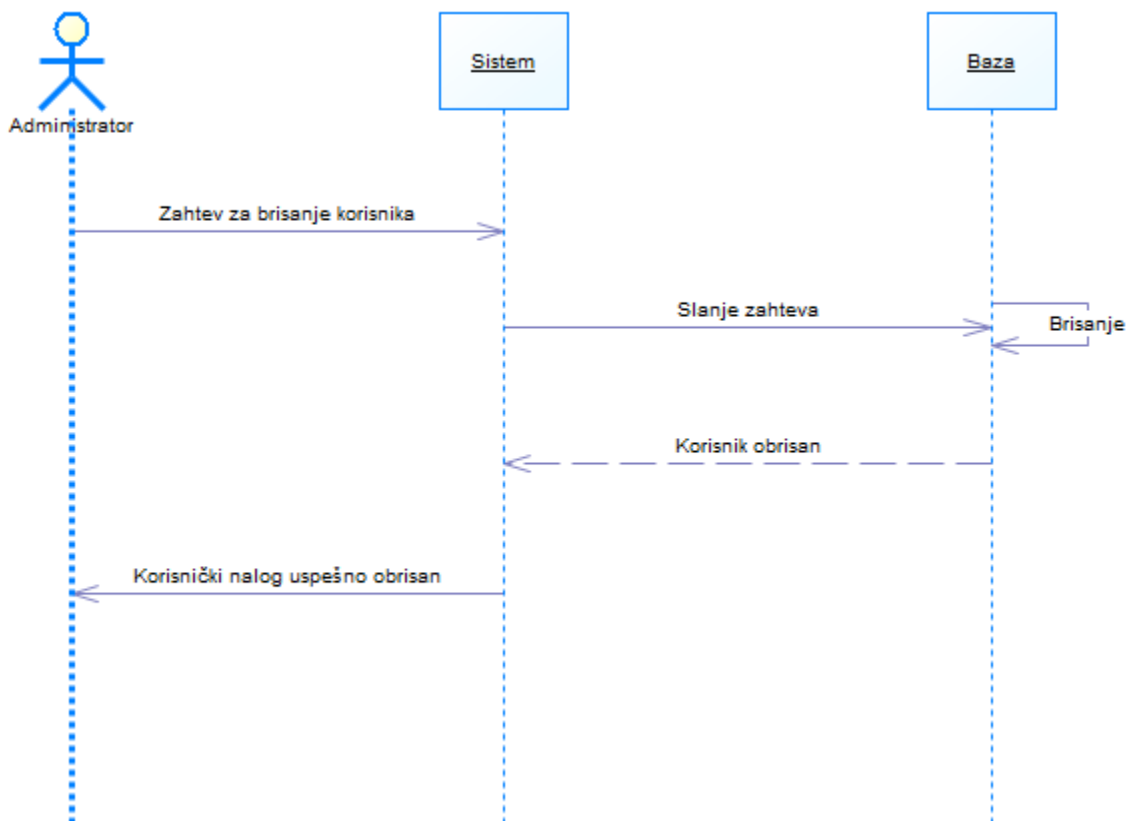
3.6.16 Dijagram sekvenci za ažuriranje korisnika – Administrator



Slika 22 – SSD – Ažuriranje korisnika

Administrator šalje sistemu zahtev za ažuriranje korisničkog naloga (korisnika), tako što iz liste svih korisnika odabere konkretnog korisnika. Sistem vraća formu za ažuriranje. Korisnik ispravlja željene podatke, šalje formu dalje sistemu, sistem nakon toga vrši validaciju forme, pa šalje podatke u bazu, baza odgovara sa potvrdnog porukom o ažuriranju, a sistem dalje, odgovara administratoru potvrdnom porukom o ažuriranju korisnika.

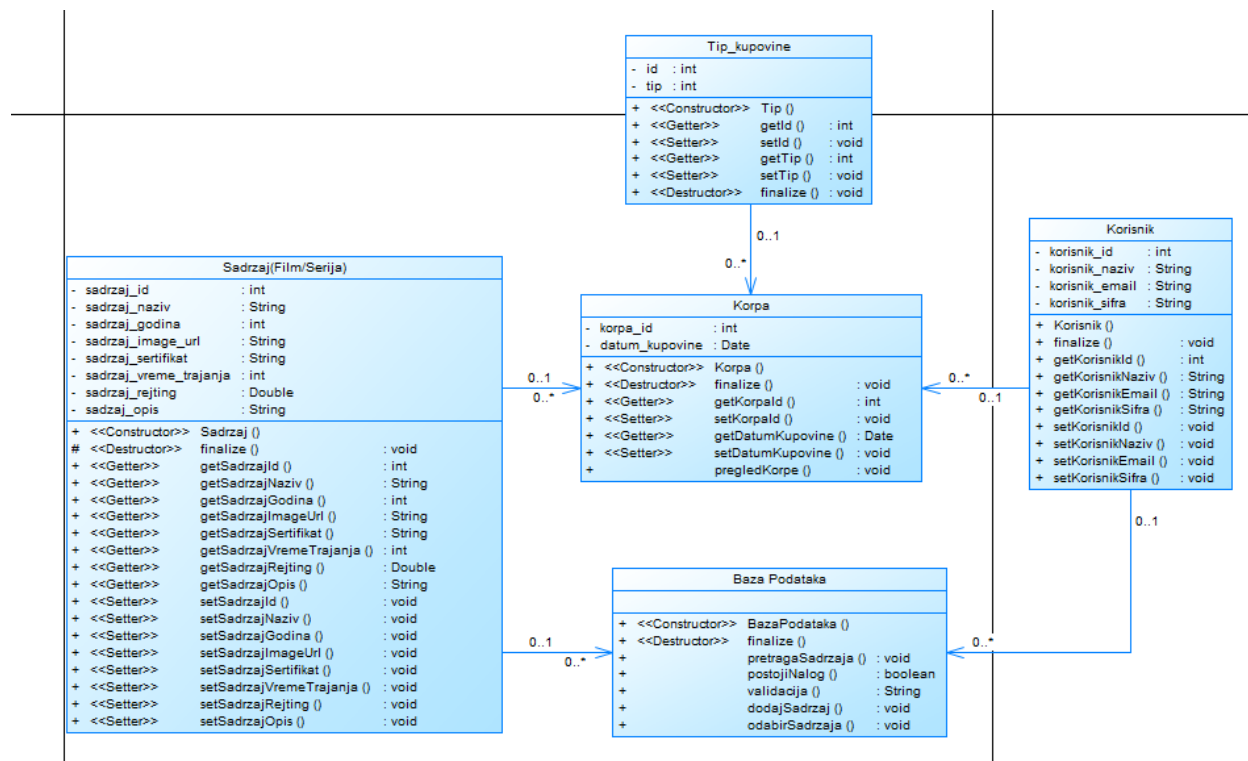
3.6.17 Dijagram sekvenci za brisanje korisnika – Administrator



Slika 23 – SSD – Brisanje korisnika

Administrator iz liste korisnika bira jednog korisnika, nakon čega bira dugme obriši. Sistem šalje bazi zahtev za brisanje korisnika sa tim identifikacionim brojem. Baza briše korisnika, te šalje odgovor sistemu o uspešno obrisanom korisniku. Dalje, sistem šalje administratoru povratnu poruku o uspešno obrisanom korisničkom nalogu.

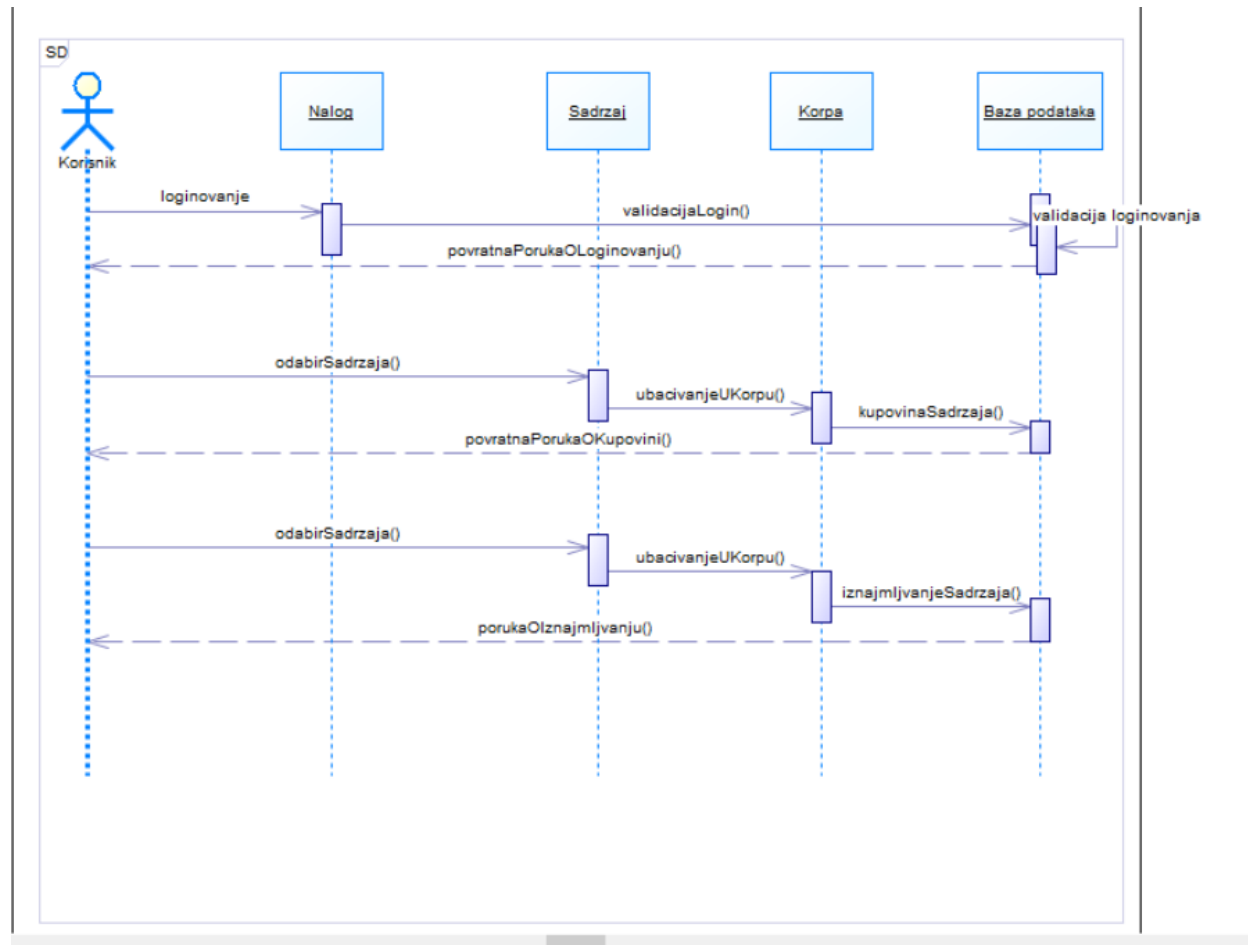
3.7 KLASSE I OBJEKTI



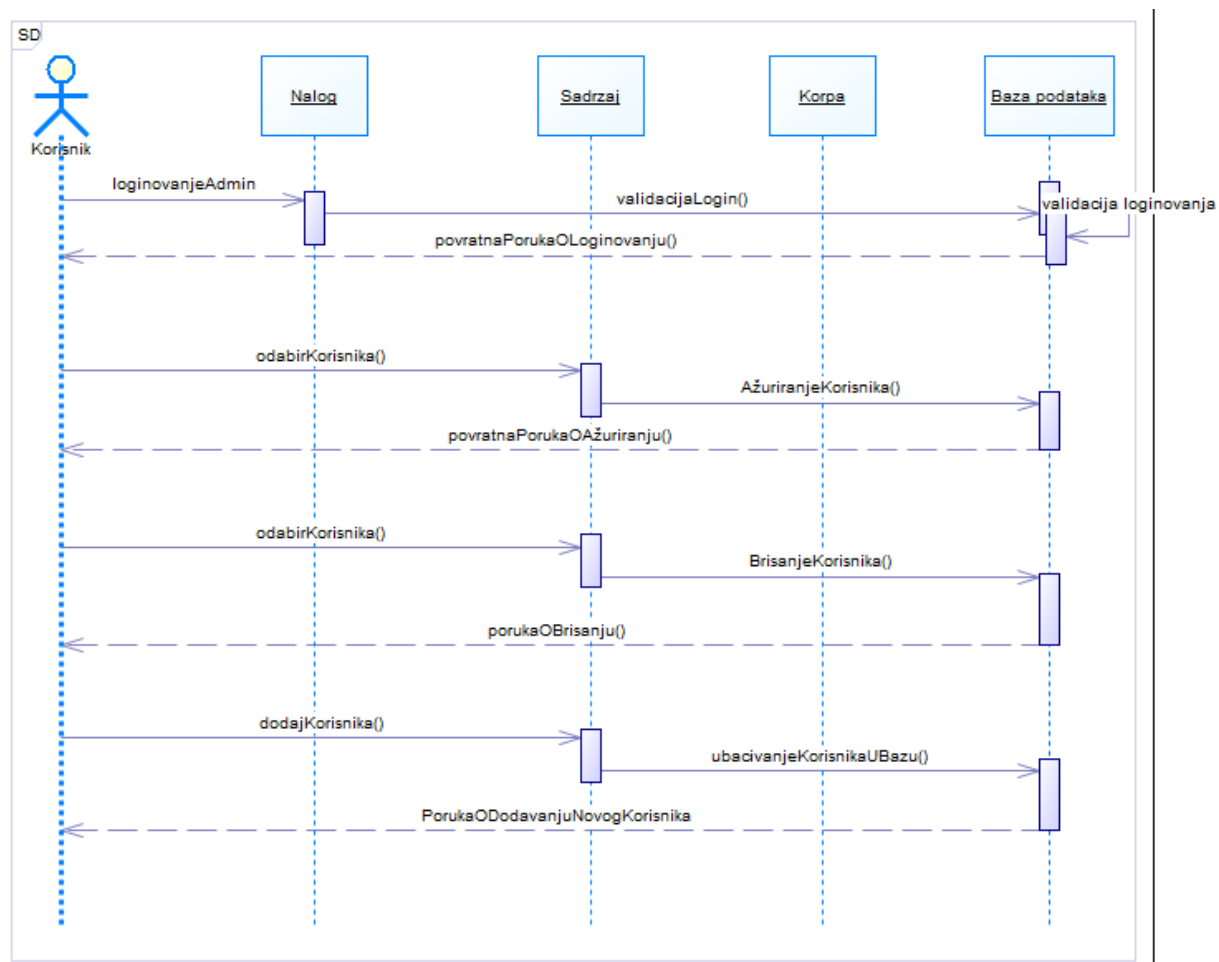
Slika 24 – CD – Klasni Dijagram (Class Diagram)

Slika predstavlja klasni dijagram, koji se sastoji od klasa Korisnik, Sadržaj, Korpa, Tip kupovine, i baza podataka. Klase se sastoje od atributa, kao i njihovih getter i setter metoda, dok neke klase imaju i druge metode, koje vrše određenu funkciju u sistemu.

3.8 SEKVENCIJALNI DIJAGRAM(NIVO UPOTREBLJENIH KLASA)

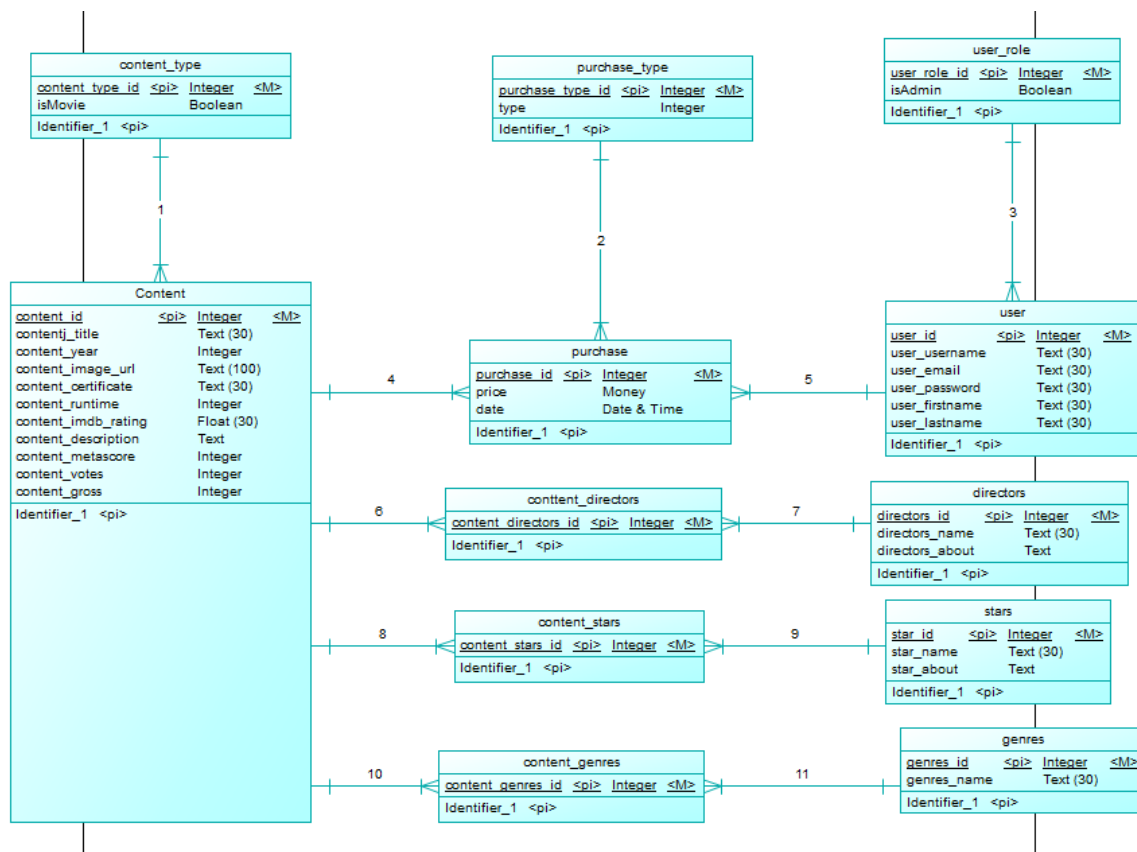


Slika 25 – SD – Sekvencijalni dijagram – Korisnik



Slika 26 – SD – Sekvencijalni dijagram – Administrator

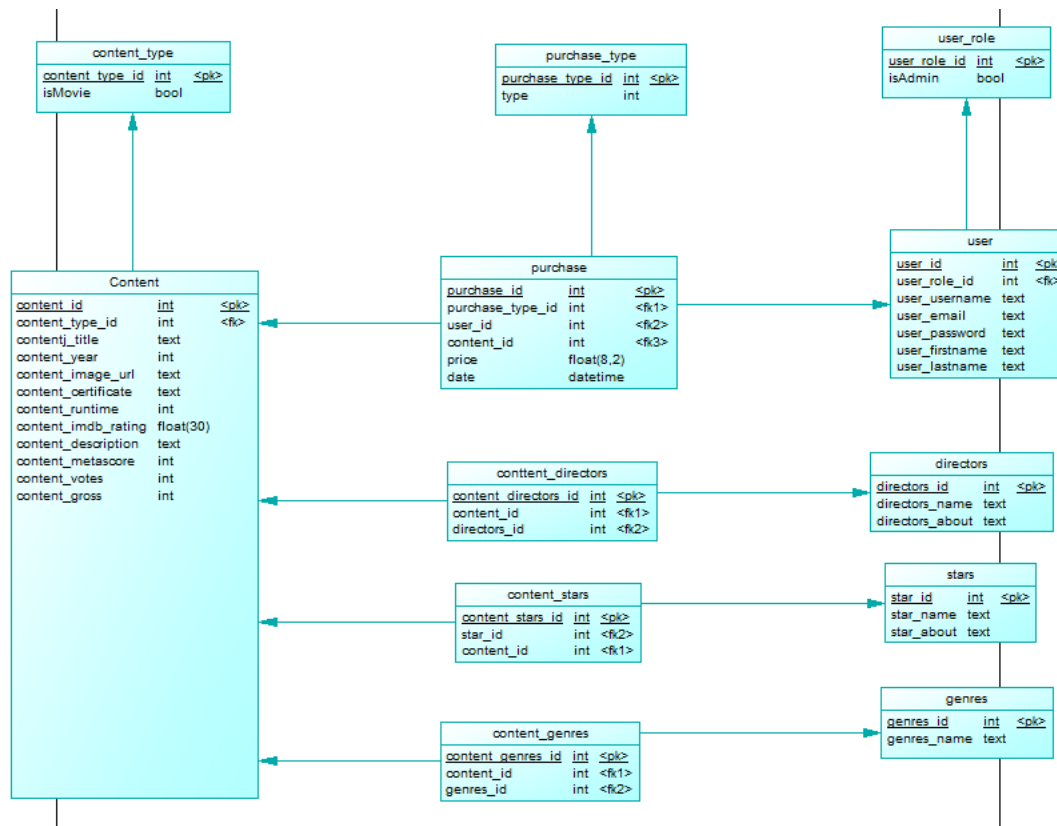
3.9 MODEL BAZE PODATAKA



Slika 27 – Baza podataka – Konceptualni model

Konceptualni model baze podataka

Model sadrži 12 entiteta. Osnovni entiteti su Content (Sadržaj – Film ili Serija), Korisnik (Users) i Kupovina (Purchase - korpa). Ostali entiteti su dodati iz razloga što jedan film ili serija može imati više glumaca, sadrži se od više žanrova, i mogu je snimati više direktora. Pored toga, svaki korisnik sistema mora imati svoju ulogu, da li je administrator ili običan korisnik. Takođe, mora se znati da li se sadržaj iznajmljuje (rent) ili kupuje (buy), što se postiže tabelom purchase_type. Konačno, da bi smo znali da li se radi o seriji ili filmu, postoji dodatna tabela tj. entitet koji pokazuje da li se radi o filmu ili seriji. Veze između ovih relacija su uglavnom više na više, te se iz tog razloga pojavljuje među tabela. S druge strane, tabele jedan na više, su definisane tako da, user_role (npr) može pripadati više korisnika, a minimum jednom, dok jedan korisnik može imati jednu ulogu u trenutku minimum (Uloga administrator automatski obuhvata i ulogu user i administrator u jednom).



Slika 28 – Baza podataka – Fizički model

Fizički model baze podataka

U fizičkom modelu se primarni ključevi iz svojih primarnih entiteta spuštaju u entitete(tabele) sa kojima su povezani tj. u kojima su sekundarni ključevi.

4. ARHITEKTURA SISTEMA

4.1 MVC arhitektura

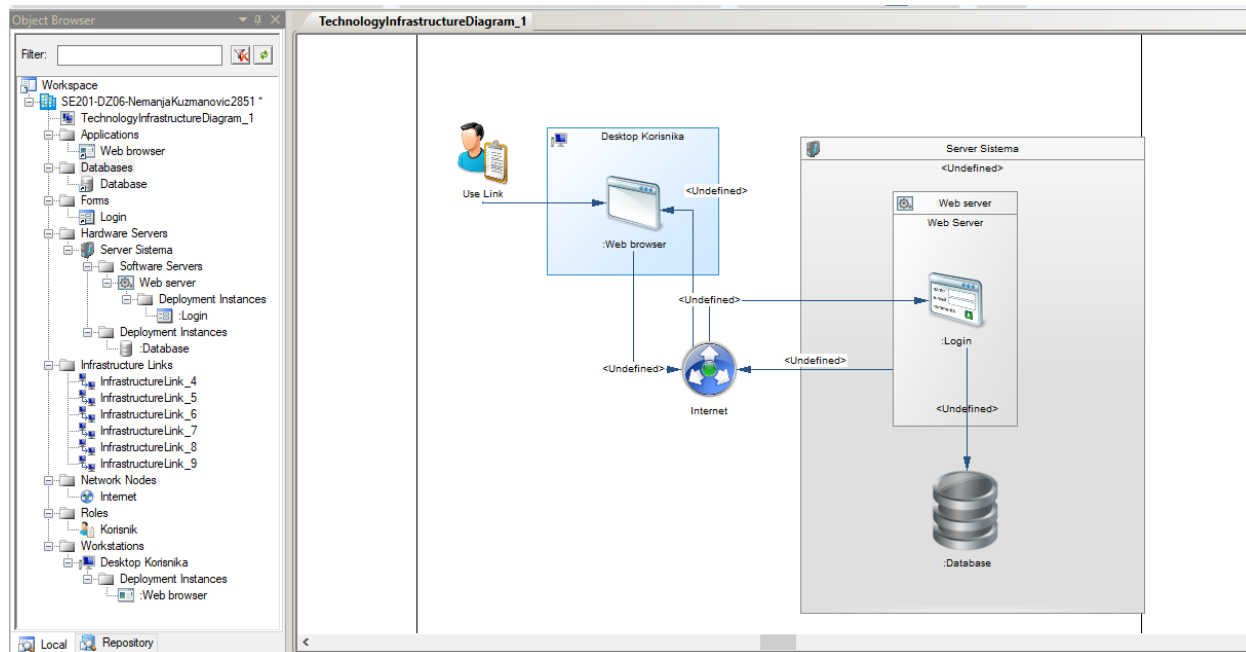
MVC arhitektura je softverski patern Model-view-controller koji odvaja prikaz informacija od interakcije korisnika sa tim informacijama. Model se sastoji od podataka aplikacije, poslovnih pravila, logike i funkcija. View može da bude bilo koji izlazni prikaz podataka, kao što je dijagram ili grafik. Više prikaza istog podatka je moguće, kao što je grafik sa barovima za menadžment i tabelarni prikaz za računovođe. Controller uzima ulazne podatke i konvertuje ih u komande za model ili view.

Arhitektura je korišćena iz razloga što se radi web aplikacija, a MVC je naširoko prihvaćen kao patern za aplikacije na svetskom nivou, i pored Framework-a, postoji i u mnogim programskim jezicima.

Prednosti MVC arhitekture su između ostalog, ponovno korišćenje koda, podela sistema na međusobno nezavisne celine, iz čega dobijamo i najviše i najveće pogodnosti. Sami projekti postaju sistematičniji, a njihovi pojedinačni delovi se mogu lako menjati i poboljšavati, dok kod pisan na ovakav način je jako lako testirati. Takođe, paralelan razvoj aplikacije se lako može organizovati i samim tim povećati produktivnost.

Kao najveća mana je prevelika kompleksnost kod razvoja manjih aplikacija, što u tim slučajevima dovodi do pogoršanja i dizajna i performansi. Ponekad se može desiti da usled čestih promena modela, view bude preplavljen zahtevima za izmenu. Ukoliko on služi za prikazivanje sadržaja, kome je naravno potrebno neko vreme za renderovanje, to može značajno usporiti rad sistema, i dovesti do kašnjenja.

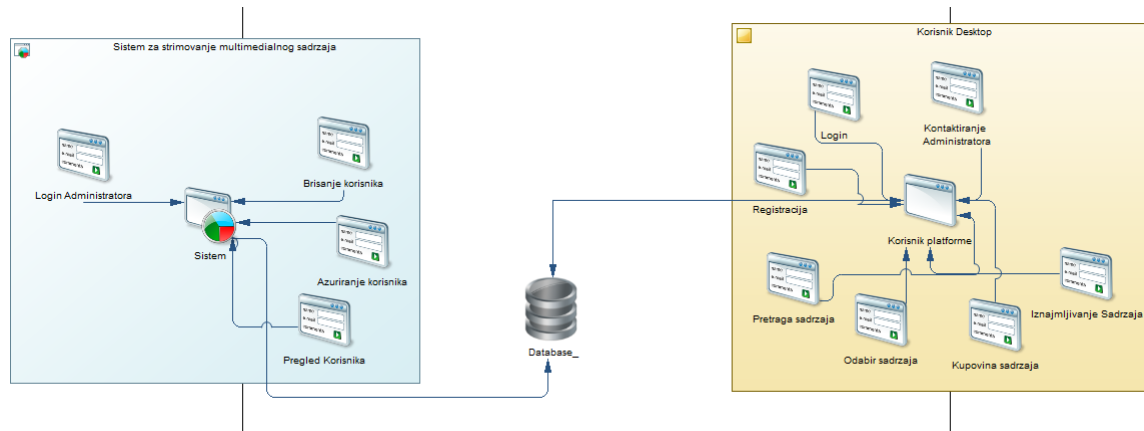
4.2 INFRASTRUKTURNI DIJAGRAM



Slika 29 – Infraskrukturini dijagram

Na priloženoj slici možemo videti vizuelno kako sve funkcioniše, na infrakstrukturnom nivou. Korisnik prvenstveno ulazi u svoj pretraživač, i putem interneta pristupa platformi. Nakon toga, prijavljuje se na sistem, gde se, opet preko interneta, njegovi kredencijalni enkriptovano(sigurno) putem ne bezbedne mreže prosleđuju do našeg veb servera, gde se raspakuju, i proveravaju sa bazom podataka. Ukoliko se sve u redu, veb server putem interneta, direktno do pretraživača korisnika, šalje potvrdu poruku, i pušta ga da se prijavi na sistem. Istom logikom se sve funkcionalnosti izvršavaju.

4.3 DIJAGRAM ARHITEKTURE



Slika 30 – Dijagram arhitekture

5. TESTIRANJE

Testiranje je izvršeno pomoću Selenium IDE plugin-a za google chrome, skinuto sa chrome websotre-a.

Ovaj plugin je namenjen automatizaciji interakcije čoveka i veb sajta, i jako ga je jednostavno koristiti, pri čemu smanjuje ručno otvaranje i testiranje svakog dela same stranice. Za naprednija testiranja bi trebalo koristiti na primer Protractor, jer je u pitanju Agnular veb aplikacija, i u zvaničnoj dokumentaciji google-a, baš ovaj alat za testiranje je preporučen. Međum, Selenium je takođe odličan za testiranje kako aplikacije interaguje sa korisnikom.

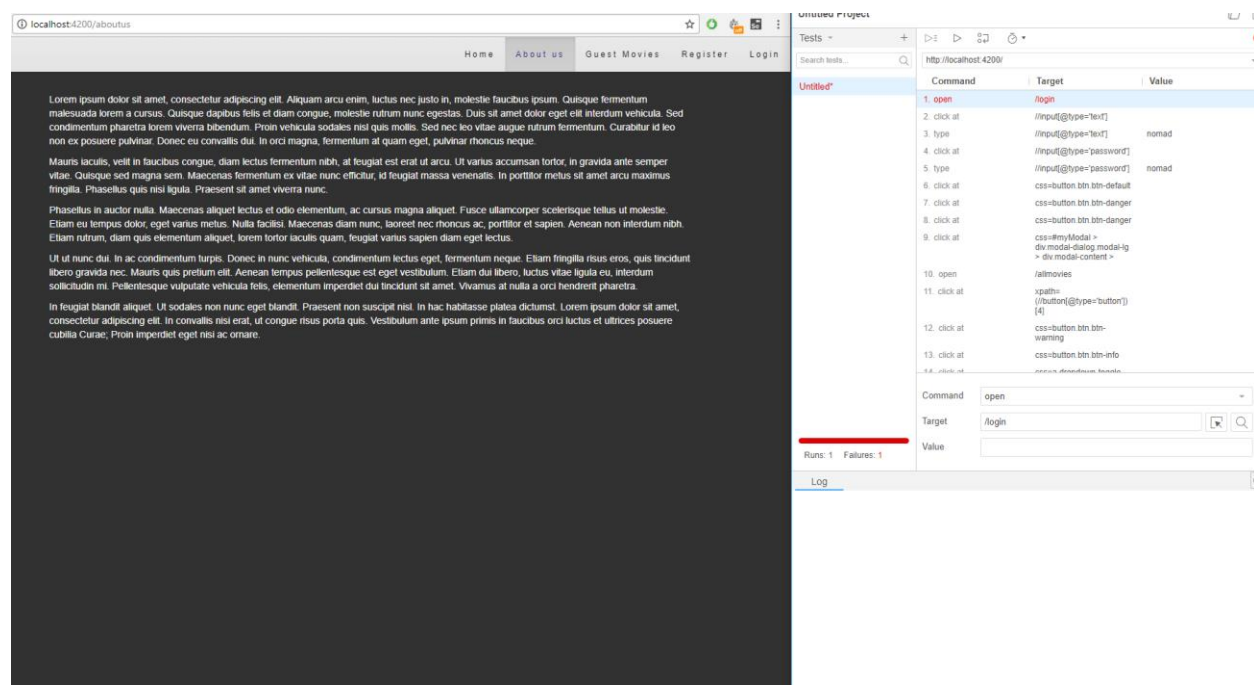
U nastavku, biće testirana aplikacija sa sledećim koracima:

Prvenstveno, korisnik će otvoriti login stranu, ukucati svoj username i password (nomad , nomad), nakon toga otvara stranu allmovies, gde će odabrati sadržaj iz liste koji će prvo kupiti, potom će ponovo otvoriti listu sadržaja, iznajmiti neki drugi sadržaj, i konacno izlogovati se sa sistem, ponovo ulogovati, i prikazati listu sadržaja koju je kupio, da je zaista tu, vezana sa njegov nalog.

*Mymovies – Moj sadržaj

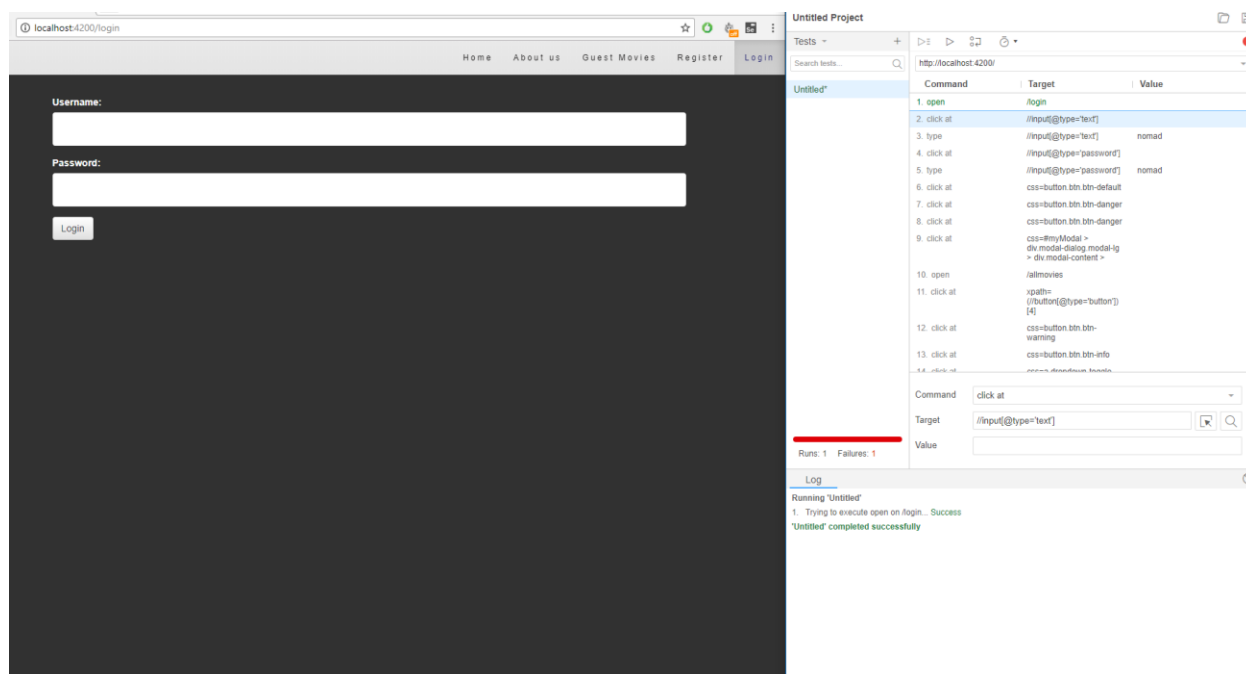
*Allmovies – Ceo sadržaj

(Aplikacija kada je prvobitno rađena, u vidu su bili filmovi, međutim dodate su i serije, ali nazivi su ostali stari, dakle, gde god piše film, to se zapravo odnosi na sadržaj, bilo film ili seriju. Daljim razvojem aplikacije bi se to svakako „patch,, -ovalo, i trenutno, nije u fokusu izrada te veb aplikacije, nego dokumentacija).



Slika 31 – Testiranje – about us

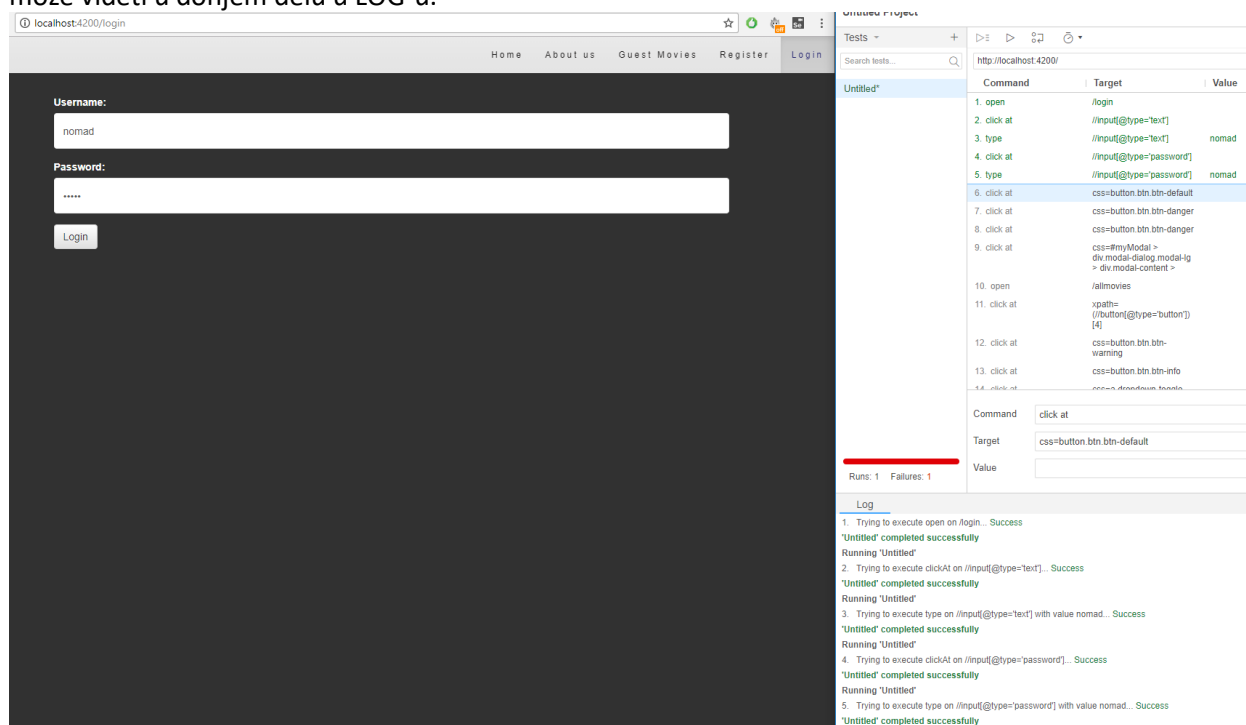
Testiranje je počelo sa stranice aboutus. U selenium IDE, bira se komanda open, i upisuje se strana na koju se želi otići. Pre toga je u gornjem delu selenium ide, ubačena bazna adresa aplikacije: localhost:4200



Slika 32 – Testiranje – Login

Kako bi se testiranje što vernije prikazalo, pokretala se komanda po komandu, i slika po slika se ubacivala, sa leve strane se nalazi aplikacija, sa desne selenium IDE.

Prvi korak koji se zahteva od plugin-a je pokretanje ove /login strane, koje je i odrađeno uspešno. To se može videti u donjem delu u LOG-u.



Slika 33 – Testiranje – Login korisnika

Korisnik sa kredencijalima (username: nomad, password: nomad) se loguje, preko selenium testa, tako što se odaberu komande click at i onda se odabere polje gde se želi da selenium klikne, potom se odabere komanda type, i tekst koji treba uneti. Tih nekoliko koraka se na slici vidi. Dakle, test je uspešno prošao, plugin je upisao korisničko ime, lozinku, i pritisnuo dugme login.

The screenshot displays a Selenium test execution environment. On the left, a web browser shows the 'allmovies' website at 'localhost:4200'. The website has a search bar, navigation links (Home, About us, My Movies, All Movies, My Account), and a list of movies including 'The Shape of Water (2017)' and 'Star Wars: Poslednji džedaji (2017)'. On the right, the 'Untitled Project' test runner is visible. It shows a list of test steps with commands like 'open', 'click at', and 'type'. The 'Value' column contains the test data, such as '/login', '//input[@type='text']', 'nomad', and '//input[@type='password']'. The 'Log' section at the bottom shows the execution results, indicating that the test was successful.

Command	Target	Value
1. open	/login	
2. click at	//input[@type='text']	
3. type	//input[@type='text']	nomad
4. click at	//input[@type='password']	
5. type	//input[@type='password']	nomad
6. click at	css=button.btn.btn-default	
7. click at	css=button.btn.btn-danger	
8. click at	css=button.btn.btn-danger	
9. click at	css=#myModal > div.modal-dialog.modal-lg > div.modal-content >	
10. open	/allmovies	
11. click at	xpath=(//button[@type='button'])[4]	
12. click at	css=button.btn.btn-warning	
13. click at	css=button.btn.btn-info	

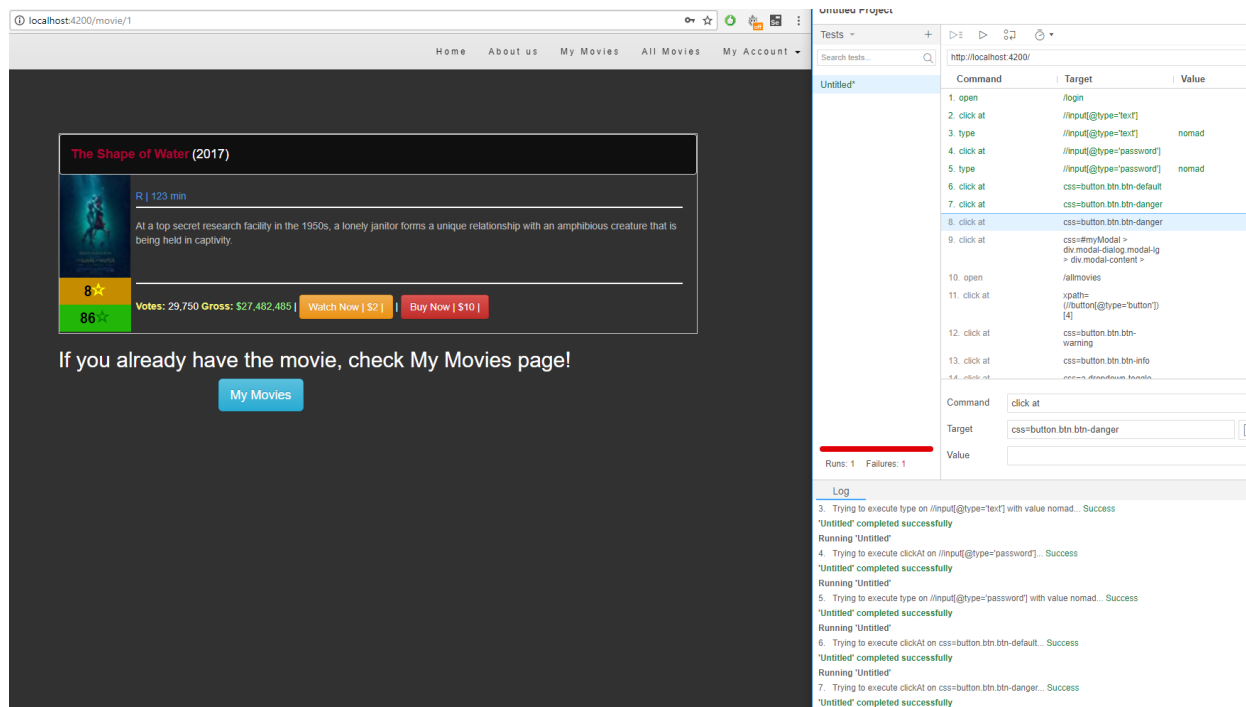
Log

```

2. Trying to execute clickAt on //input[@type='text']... Success
'Untitled' completed successfully
Running 'Untitled'
3. Trying to execute type on //input[@type='text'] with value nomad... Success
'Untitled' completed successfully
Running 'Untitled'
4. Trying to execute clickAt on //input[@type='password']... Success
'Untitled' completed successfully
Running 'Untitled'
5. Trying to execute type on //input[@type='password'] with value nomad... Success
'Untitled' completed successfully
Running 'Untitled'
6. Trying to execute clickAt on css=button.btn.btn-default... Success
'Untitled' completed successfully
  
```

Slika 34 – Testiranje – Login uspešan

Pritiskom na dugme myaccount, može se videti da se zaista korisnik nomad ulogovao.



The screenshot shows a web browser at `localhost:4200/movie/1` displaying a movie page for "The Shape of Water (2017)". The page includes a movie poster, a rating of 8.6, and buttons for "Watch Now | \$2" and "Buy Now | \$10". A message prompts the user to check their "My Movies" page if they already own the movie.

On the right, the Selenium IDE interface shows a test suite named "Untitled" with the following commands:

Command	Target	Value
1. open	/login	
2. click at	//input[@type='text']	
3. type	//input[@type='text']	nomad
4. click at	//input[@type='password']	
5. type	//input[@type='password']	nomad
6. click at	css=button.btn.btn-default	
7. click at	css=button.btn.btn-danger	
8. click at	css=button.btn.btn-danger	
9. click at	css=#myModal > div.modal-dialog.modal-lg > div.modal-content >	
10. open	/allmovies	
11. click at	xpath=//button[@type='button']	[4]
12. click at	css=button.btn.btn-warning	
13. click at	css=button.btn.btn-info	

The log shows the following messages:

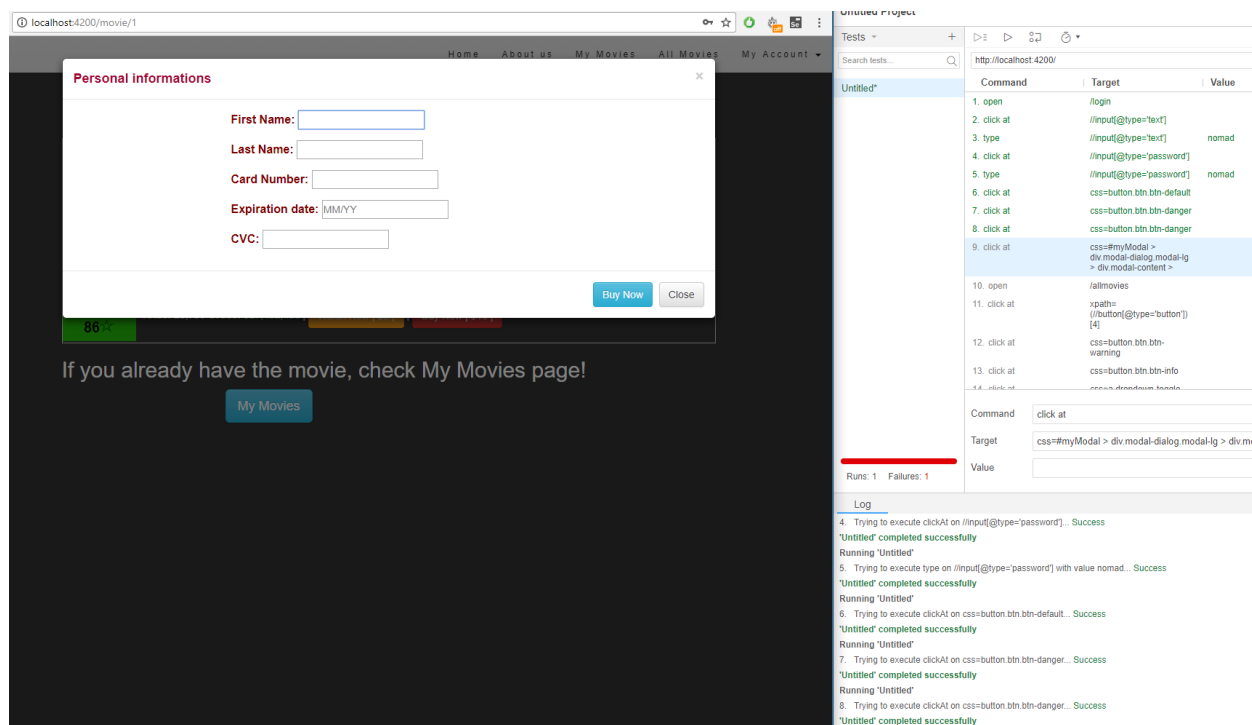
```

3. Trying to execute type on //input[@type='text'] with value nomad... Success
'Untitled' completed successfully
Running 'Untitled'
4. Trying to execute clickAt on //input[@type='password']... Success
'Untitled' completed successfully
Running 'Untitled'
5. Trying to execute type on //input[@type='password'] with value nomad... Success
'Untitled' completed successfully
Running 'Untitled'
6. Trying to execute clickAt on css=button.btn.btn-default... Success
'Untitled' completed successfully
Running 'Untitled'
7. Trying to execute clickAt on css=button.btn.btn-danger... Success
'Untitled' completed successfully

```

Slika 35 – Testiranje – Kupovina

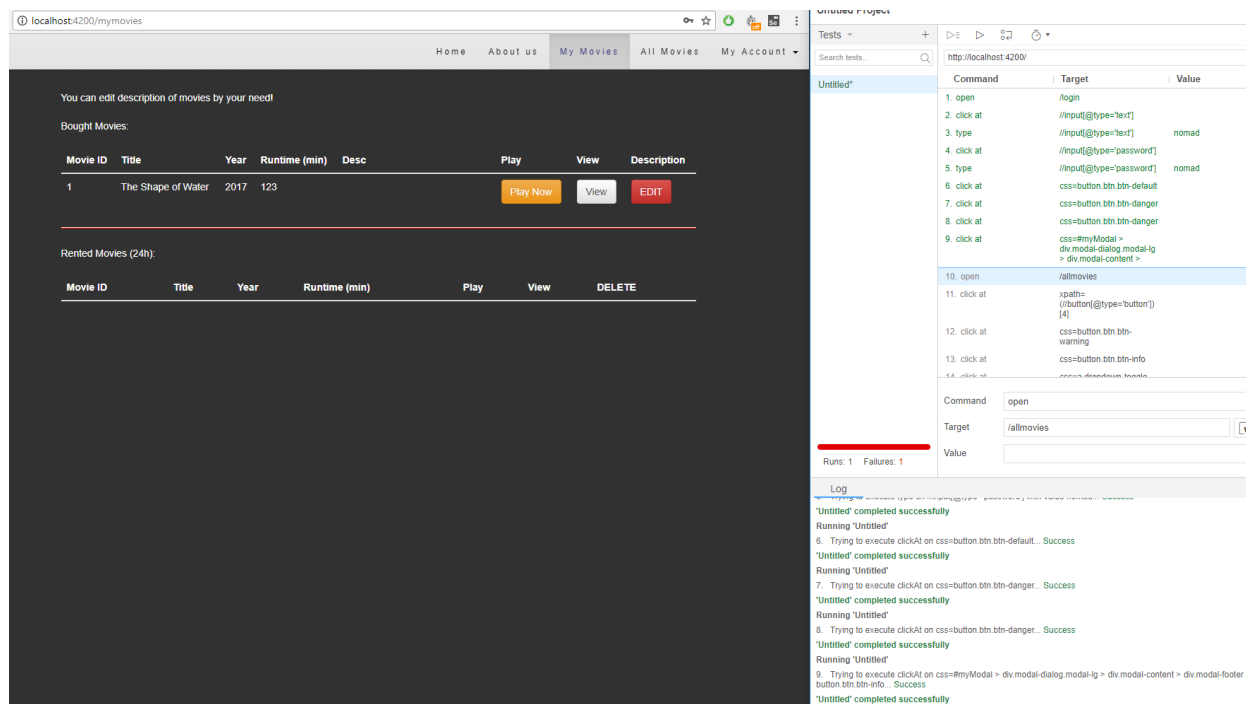
Od testa se sada zahteva da otvori neki sadržaj, potom klikne na Buy Now tj. Kupi. Što je uspešno i izvršeno kao što se može videti i sa leve strane slike, a i sa desne, u log sekciji.



Slika 36 – Testiranje – about us

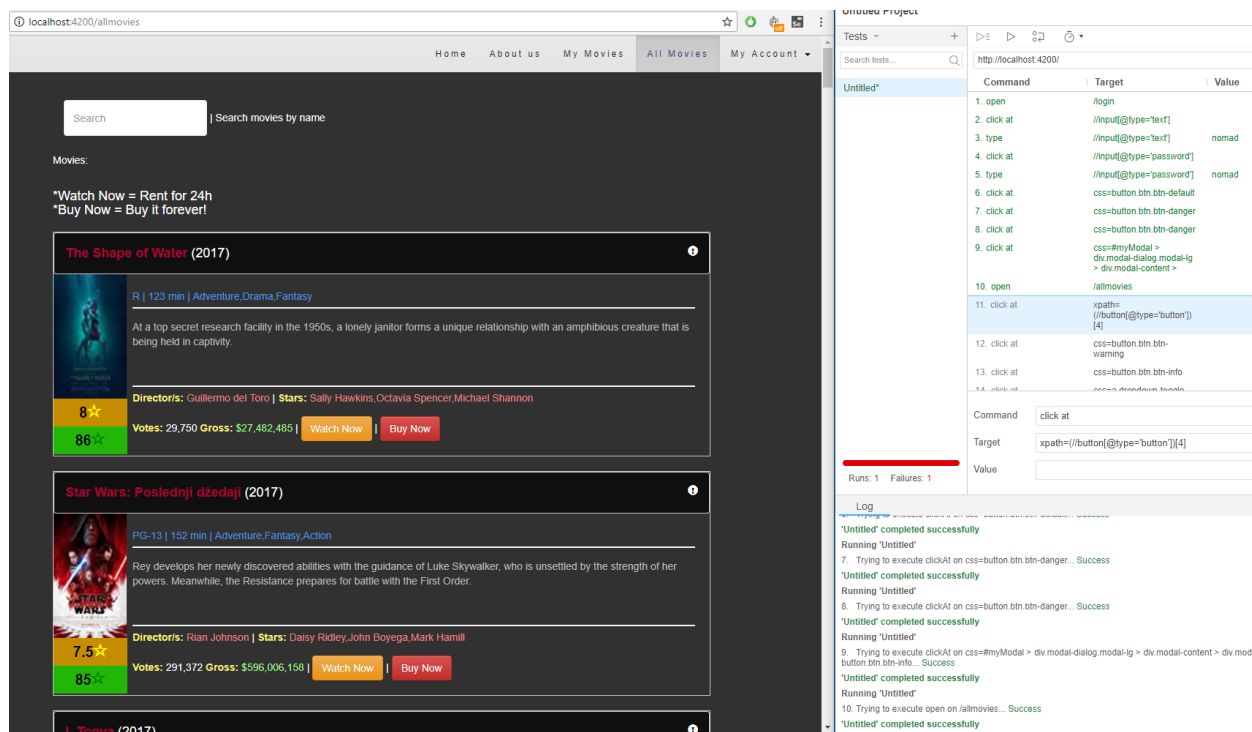
Kada se pritisne dugme kupi, dobija se forma za popunjavanje informacija o kreditnoj kartici kako bi kupili sadržaj, međutim to nije implementirano, jer bi se u realnoj aplikaciji verovatno koristio neki API, vezan za visa, mastercard, paypal i slično.

Od seleniuma se traži da pritisne dugme buy now, i veruje se da su uneti podatci da ima para na kartici dovoljno za kupovinu, i da je sve ispravno prošlo u tom delu.



Slika 37 – Testiranje – Kupovina uspešna

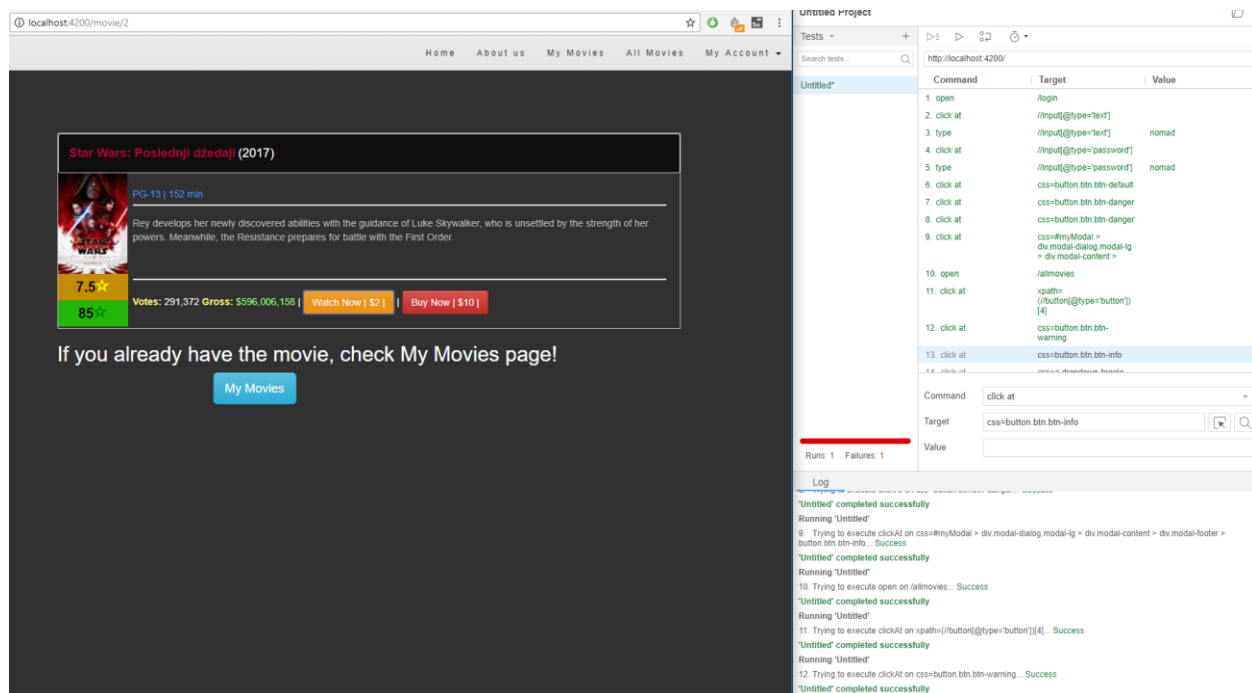
Nakon prethodnih koraka, vidimo da je kupovina sadržaja uspeła. Korisnik bi sada mogao da odabere da vidi detaljno film i njegov opis, da ga gleda, ili da doda neku poruku uz film (npr. odličan film, stigao do 35min 20sec, i slično).



Slika 38 – Testiranje – Iznajmljivanje (RENT)

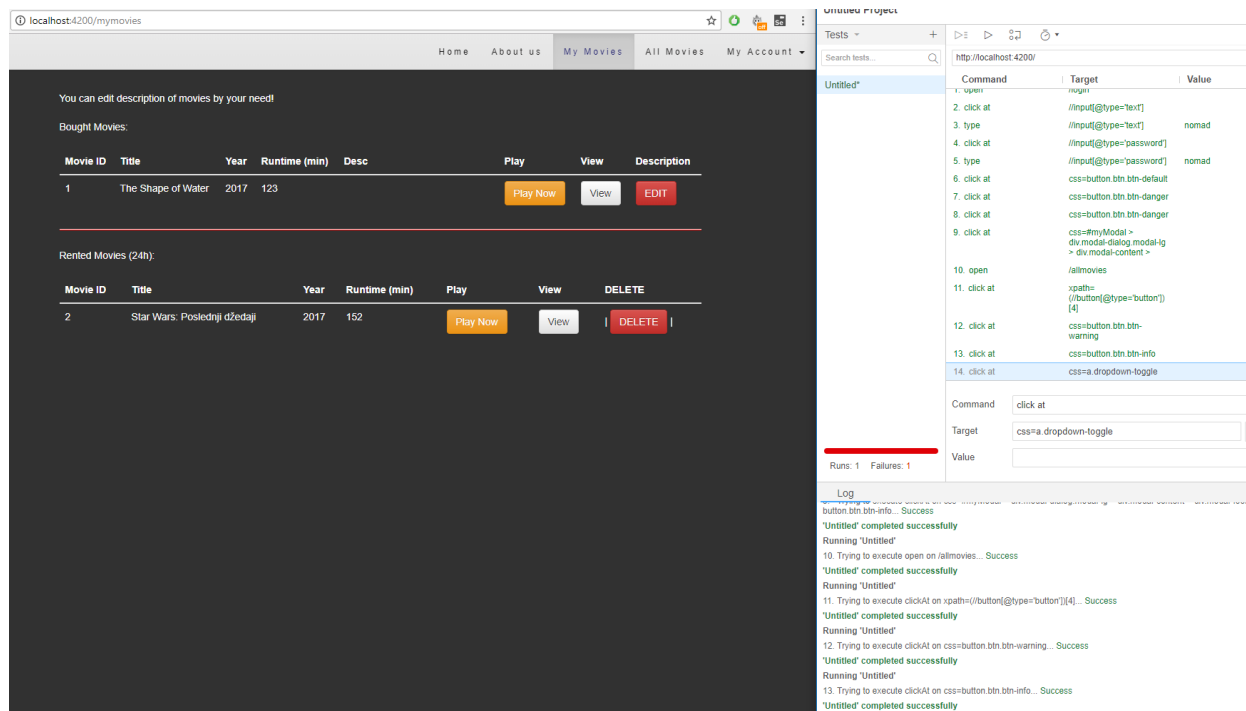
Sledeće će se testirati iznajmljivanje nekog sadržaja. Korisnik bi trebao da klikne na sadržaj potom na Rent, gde bi posle unosa informacija o kreditnoj kartici i kupovini dobio sadržaj na 24h da gleda.

U selenium IDE smo upisali komande gde bi korisnik kliknuo i šta uradio, kako bi testirali automatski da li ovo sve funkcioniše kako je zamišljeno.



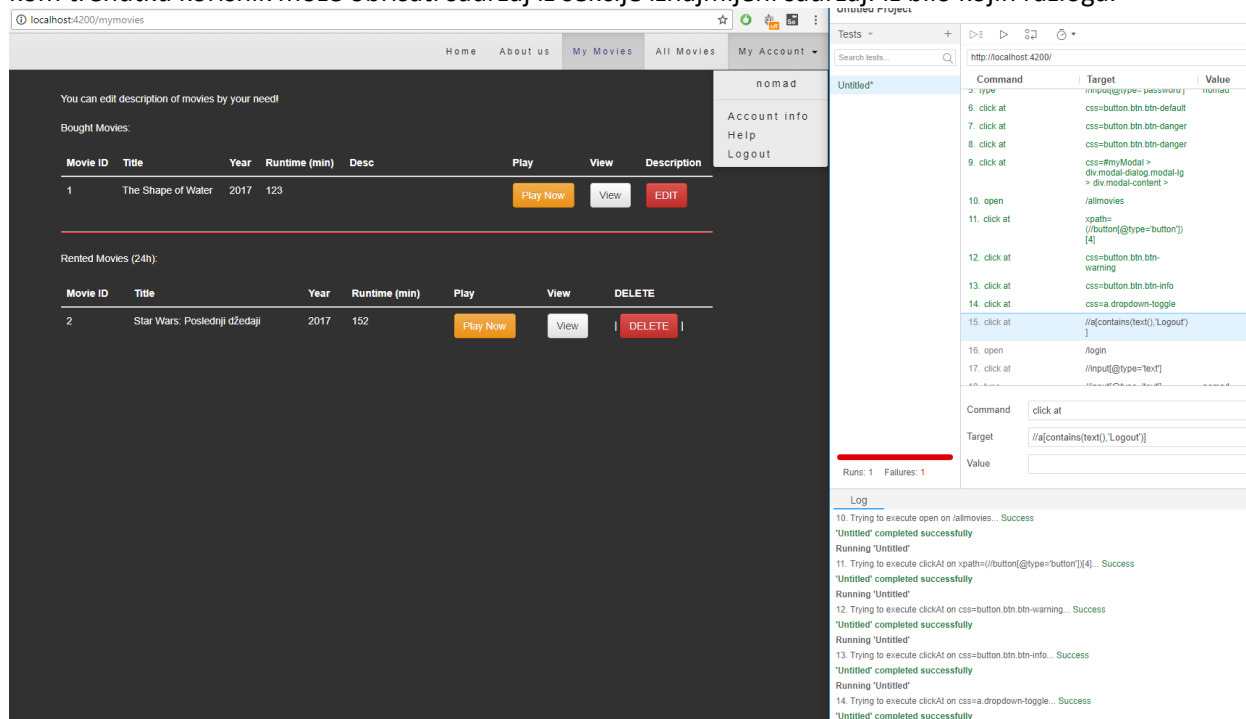
Slika 39 – Testiranje – Iznajmljivanje filma

Ukoliko bi korisnik, pokušao da kupi ili iznajmi sadržaj koji se već nalazi u njegovoj listi, dobio bi povratnu poruku da taj sadržaj već ima kupljen i ne bi se nastavilo sa kupovinom, nego bi išao na sekciju MyMovies(moj sadržaj) gde bi zaista mogao videti taj sadržaj tamo.

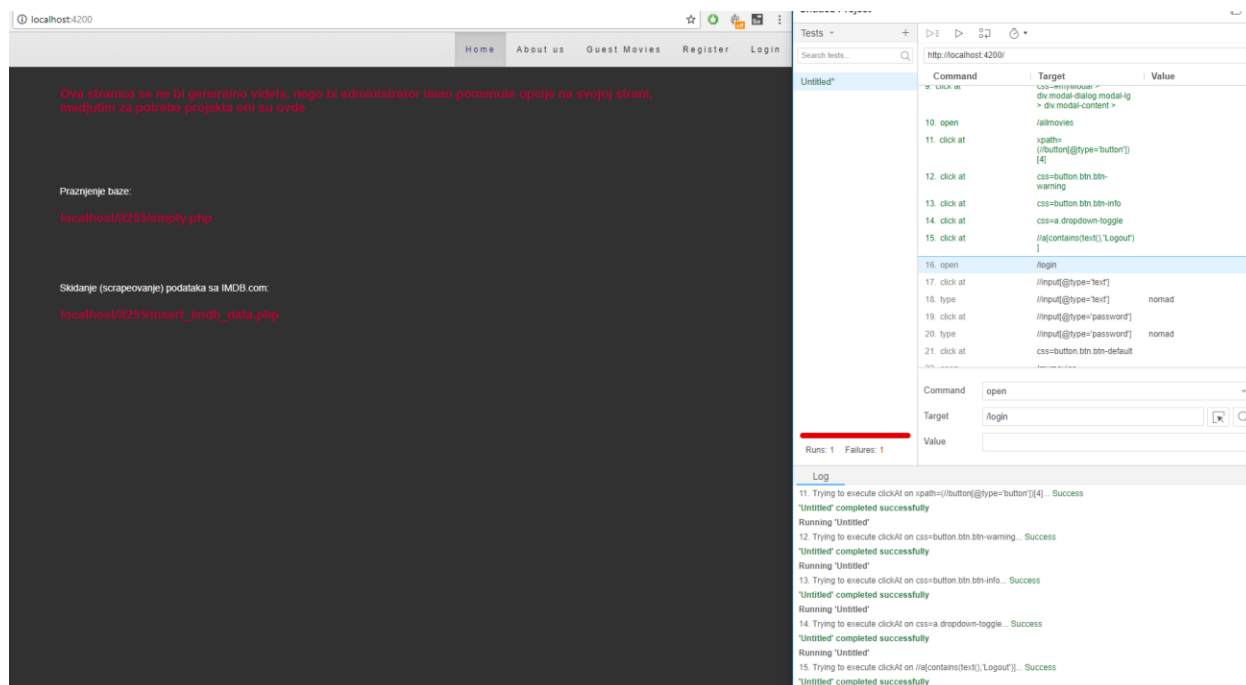


Slika 40 – Testiranje – Uspešno iznajmljivanje

Kao i prethodni i ovaj test je prošao bez problema. Korisnik je uspeo da iznajmi i kupi neki sadržaj, što se može i videti u njegovoj listi sadržaja. Sadržaj koji je iznajmio će se nalaziti na njegovom nalogu naredna 24h nakon čega se automatski briše, a stranica obnavlja(refrešuje) bez tog sadržaja na nalogu. U bilo kom trenutku korisnik može obrisati sadržaj iz sekcije iznajmljeni sadržaj. Iz bilo kojih razloga.

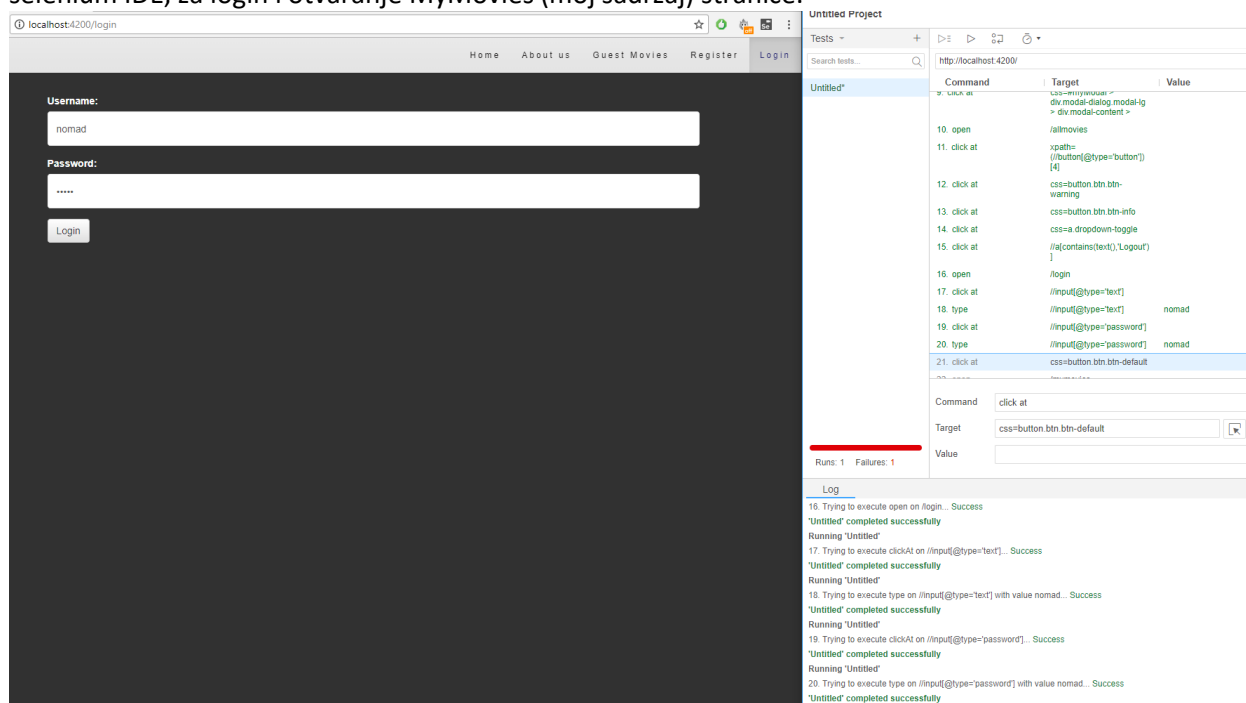


Slika 41 – Testiranje – Logout

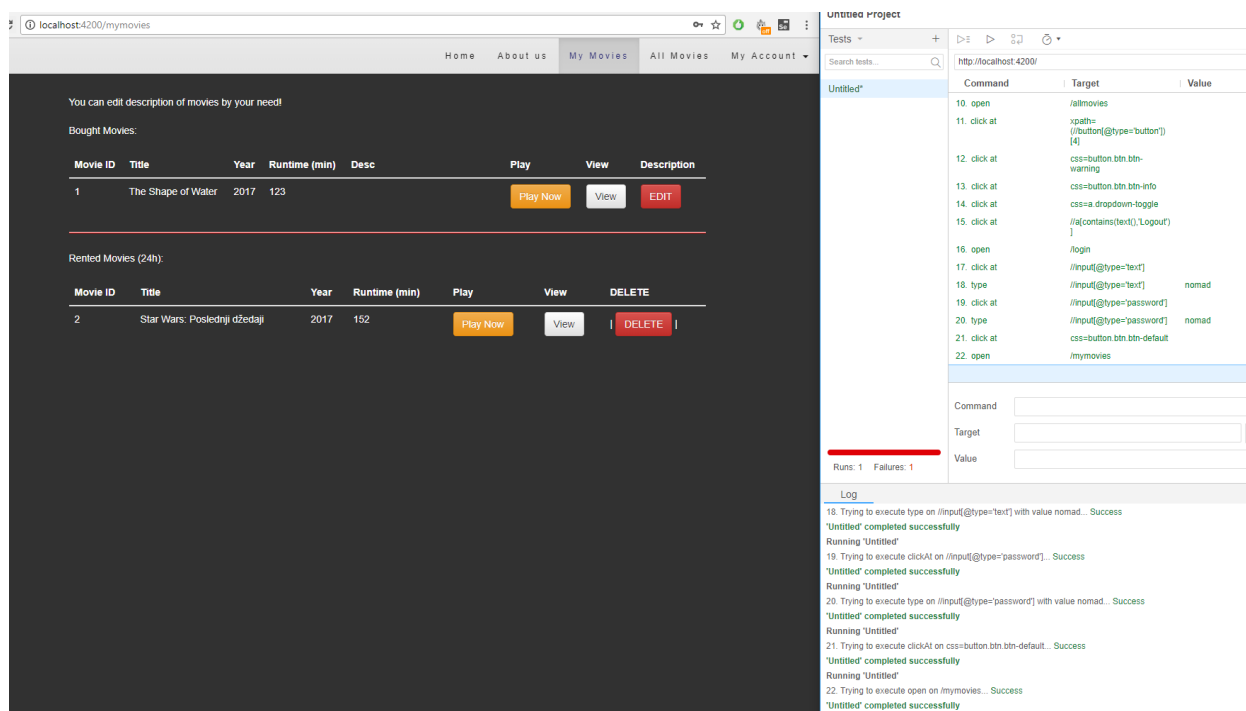


Slika 42 – Testiranje – Home

Da bi smo se uverili da sve funkcioniše kao što je predviđeno, ponovo ćemo napisati nekoliko komandi u selenium IDE, za login i otvaranje MyMovies (moj sadržaj) stranice.



Slika 43 – Testiranje – Ponovan Login



Slika 44 – Testiranje – Sadržaj korisnika

Kroz samo nekoliko komandi uspele smo da automatizujemo testiranje aplikacije, gde bi korisniku znatno duže trebalo da sve istestira. Selenium IDE plugin kroz nekoliko sekundi u realnom vremenu pokrene komande, gde mi sa druge strane možemo takođe u realnom vremenu gledati kako sam plugin otvara stranice i kuca podatke, kao da je pravi korisnik u pitanju i da koristi sistem. Sve u svemu, odličan alat za testiranje veb aplikacija, koji nam je pomogao da potvrdimo da sistem funkcioniše kako je zamišljen.

Selenium IDE - Untitled Project*

Untitled Project*

Tests ▾ +

Search tests... 🔍

http://localhost:4200/ ▾

Command	Target	Value
1. open	/login	
2. click at	//input[@type='text']	
3. type	//input[@type='text']	nomad
4. click at	//input[@type='password']	
5. type	//input[@type='password']	nomad
6. click at	css=button.btn.btn-default	
7. click at	css=button.btn.btn-danger	
8. click at	css=button.btn.btn-danger	
9. click at	css=#myModal > div.modal-dialog.modal-lg > div.modal-content > ...	
10. open	/allmovies	
11. click at	xpath= (//button[@type='button'])[4]	
12. click at	css=button.btn.btn-warning	
13. click at	css=button.btn.btn-info	
14. click at	css=a.dropdown-toggle	

Runs: 1 Failures: 0

Log

- 1. Trying to execute open on /login... Success
- 2. Trying to execute clickAt on //input[@type='text']... Success
- 3. Trying to execute type on //input[@type='text'] with value nomad... Success
- 4. Trying to execute clickAt on //input[@type='password']... Success
- 5. Trying to execute type on //input[@type='password'] with value nomad... Success
- 6. Trying to execute clickAt on css=button.btn.btn-default... Success
- 7. Trying to execute clickAt on css=button.btn.btn-danger... Success
- 8. Success
- 9. Trying to execute clickAt on css=#myModal > div.modal-dialog.modal-lg > div.modal-content > div.modal-footer > button.btn.btn-info... Success
- 10. Trying to execute open on /allmovies... Success
- 11. Trying to execute clickAt on xpath=(//button[@type='button'])[4]... Success
- 12. Trying to execute clickAt on css=button.btn.btn-warning... Success
- 13. Trying to execute clickAt on css=button.btn.btn-info... Success

Slika 46 – Testiranje – Selenium IDE komande i log

Selenium IDE - Untitled Project*

Untitled Project*

Tests ▾ +

Search tests... 🔍

http://localhost:4200/ ▾

Command	Target	Value
10. open	/allmovies	
11. click at	xpath= (//button[@type='button'])[4]	
12. click at	css=button.btn.btn-warning	
13. click at	css=button.btn.btn-info	
14. click at	css=a.dropdown-toggle	
15. click at	//a[contains(text(),'Logout')]	
16. open	/login	
17. click at	//input[@type='text']	
18. type	//input[@type='text']	nomad
19. click at	//input[@type='password']	
20. type	//input[@type='password']	nomad
21. click at	css=button.btn.btn-default	
22. open	/mymovies	

Command: open ▾

Target: /login 🔍

Value:

Runs: 1 Failures: 0

Log

10. Trying to execute open on /allmovies... Success

11. Trying to execute clickAt on xpath=(//button[@type='button'])[4]... Success

12. Trying to execute clickAt on css=button.btn.btn-warning... Success

13. Trying to execute clickAt on css=button.btn.btn-info... Success

14. Trying to execute clickAt on css=a.dropdown-toggle... Success

15. Trying to execute clickAt on //a[contains(text(),'Logout')]... Success

16. Trying to execute open on /login... Success

17. Trying to execute clickAt on //input[@type='text']... Success

18. Trying to execute type on //input[@type='text'] with value nomad... Success

19. Trying to execute clickAt on //input[@type='password']... Success

20. Trying to execute type on //input[@type='password'] with value nomad... Success

21. Trying to execute clickAt on css=button.btn.btn-default... Success

22. Trying to execute open on /mymovies... Success

'Untitled' completed successfully

Slika 47 – Testiranje – Selenium komande i log

6. ZAKLJUČAK

Kroz dokumentaciju projekta, i izrade samog softvera, student je prošao kroz sve što se radilo tokom semestra i obnovio znanje. Od raznih zahteva, preko dijagrama, funkcionalnosti i slično, student bi trebalo da je ovladao tehnikama i tehnologijama izrade dokumentacije koja je možda i najvažniji deo jednog softverskog sistema, bez koga ne može. Ovakvo iskustvo će svakako svima pomoći da vide šta se zapravo traži od njih, i sami da shvati delove aplikacije koje su pre „napamet,, radili, gde sada dobijaju svoju svrhu i poentu, i gde drugi korisnik, ili klijent, može jasno uvideti šta i kako sistem treba zapravo da radi. Dakle, sama izrada projekta, upotrebila je iskustva i veštine samog predmeta kroz petnaest nedelja izučavanja. Svakako će biti jako korisna i neophodna u daljem usavršavanju svakog studenta.

7. LITERATURA

- Dragan Domazet, Aleksandar Slijepčević. Uvod u Softversko Inženjerstvo (Prvo izdanje), Beograd 2014.
- Angular.io – zvanična dokumentacija
- Software engineering – design patterns - https://sourcemaking.com/design_patterns
- MVC arhitektura - <https://www.mint.rs/mintblog/web/web-development/mvc-arhitektura>
- Lams, SE201 – Uvod u softversko inženjerstvo