

Programsko inženjerstvo

Ak. god. 2020./2021.

Kamp Mlade Nade

Dokumentacija, Rev. 1

Grupa: *MajstoriSFera*

Voditelj: *Ivan Sabolić*

Datum predaje: *13. 11. 2020.*

Nastavnik: *Eugen Vušak*

Sadržaj

1. Dnevnik promjena dokumentacije

	Rev.	Opis promjene/dodatka	Autori	Datum
0.0		Napravljen template	Ivan	14/10/2020
0.1		Dodan UML dijagram	Ivan i Ivana	21/10/2020
0.2		Dodana baza podataka	Maja i Katarina	09/11/2020
0.3		Dodani funkcionalni zahtjevi	Maja i Katarina	10/11/2020
0.4		Obrasci i sekvencijski dijagrami	Ivan, Zvonimir i Martin	11/11/2020
0.4.1		Dodani obrasci i sekvencijski dijagrami	Ivana	11/11/2020
0.4.2		Dodani obrasci i sekvencijski dijagrami	Filip	11/11/2020
0.5		Uređena dokumentacija za reviziju	Ivan	12/11/2020

2. Opis projektnog zadatka

2.1 Uvodno

Cilj ovog projekta je razviti online platformu kojom bi se olakšala organizacija kampa računarstva „Mlade nade“.

2.2 Prijave na kamp

U opsegu ove platforme omogućene su prijave za sudjelovanje na kampu, ali i prijave za animatore. Prijave su omogućene samo za vrijeme trajanja prijava koje se zadaje od strane organizatora. Prijave za animatore i sudionike su u istim vremenskim intervalima. I animatori i sudionici moraju prilikom prijave upisati svoje puno ime, e-mail adresu, broj telefona, datum i godinu rođenja te kratko motivacijsko pismo. Sudionici mlađi od 18 godina moraju dodatno unijeti i broj telefona odgovorne osobe. Organizatori moraju moći vidjeti popis prijava te jednostavno odbiti ili prihvatiti prijavu.

2.3 Registracija korisničkog računa

Nakon što se nečija prijava prihvati, automatski se stvara račun za tu osobu te mu se pridodaje korisničko ime generirano tako da se uzme prvo slovo imena i prezime bez diakritičkih znakova (npr. „Ana Anić” - „aanic”). Ako je nečija prijava prihvaćena, tu osobu se o tome obavještava mailom, te mu se šalju podatci potrebni za registraciju (npr. korisničko ime i link za registraciju). Ako prijava nije bila prihvaćena, osoba se o tome isto obavještava e-mail adresom. Korisnici zatim prilikom registracije za dobiveno korisničko ime upisuju lozinku.

2.4 Grupe

Nakon završetka odabira prijava, organizatori određuju broj grupa u koje će sudionici biti razvrstani. Razvrstavanje u grupe izvodi se slučajnim odabirom, no mora postojati mogućnost razmještanja sudionika. Nakon što su formirane grupe potrebno je raspored popuniti s aktivnostima te njima pridružiti grupe.

2.5 Aktivnosti

Organizatori, koji su evidentirani direktno u bazi, definiraju aktivnosti. Svaka aktivnost ima svoje ime, kratki opis i trajanje, a postoji nekoliko tipova aktivnosti:

- aktivnosti na kojima sudjeluje samo jedna grupa,
- aktivnosti na kojima sudjeluje točno N grupa,
- aktivnosti na kojima sudjeluje maksimalno N grupa,
- aktivnosti na kojima nužno sudjeluju sve grupe.

Aktivnosti se mogu izvršavati više puta u raznim vremenima, ali jedna aktivnost uvijek traje jednako te se svaka aktivnost mora izvršavati maksimalno jednom u danom trenutku. Prilikom stvaranja instance aktivnosti potrebno je provjeriti i upozoriti na kršenje sljedećih uvjeta:

- aktivnost se neće preklapati s aktivnošću istog tipa,
- pridružen je minimalno jedan animator,
- pridružen je odgovarajući broj grupa,
- nijedna od pridruženih grupa neće imati konflikte s drugim aktivnostima koje su već navedene,
- nijedna od pridruženih grupa nije već pridružena jednakoj aktivnosti,
- pridruženi animatori neće imati konflikte s drugim aktivnostima na koje su pridruženi.

Prilikom završetka dodavanja aktivnosti treba dodatno provjeriti jesu li sve grupe sudjelovale na svakoj aktivnosti točno jednom. Svaki dan postoje tri aktivnosti koje su nepomične a one su: doručak u 8h, ručak u 12h i večera u 18h. Na tim aktivnostima sudjeluju sve grupe i svi animatori te one traju 1h.

2.6 Početna stranica

Na početnoj stranici prezentirane su osnovne informacije o kampu: vrijeme održavanja, trajanje i aktivnosti. Sudionici i animatori nakon prijave u sustav, prije početka kampa vide samo odbrojavanje do početka kampa i imaju mogućnost kontaktiranja organizatora. Nakon početka prijava (za oba tipa korisnika) vodi na stranicu koja pokazuje njihov raspored ili agendu (odabrati jedno). Animatori moraju vidjeti popis svih grupa, njihovih članova i drugih animatora kao i njihove kontakt podatke, dok sudionici vide iste podatke samo za svoju grupu i animatore. Dodatno, i sudionici i animatori moraju vidjeti popis aktivnosti na kojima su sudjelovali te moraju imati opciju ocjenjivanja aktivnosti (1-10) te ostavljanja kratkog opisa njihovog dojma aktivnosti. Nakon

što je kamp završio, sudionicima i animatorima je potrebno omogućiti ocjenjivanje i ostavljanje vlastitog dojma za cjelokupno iskustvo. Organizatori trebaju imati popis svih povratnih ocjena po aktivnostima te ih moraju moći pretraživati prema sljedećim atributima: korisnik, grupa i/ili aktivnost.

3. Specifikacija programske potpore

3.1 Funkcionalni zahtjevi

Dionici:

1. Organizator kampa (naručitelj)
2. Sudionici kampa
3. Animatori
4. Administrator
5. Razvojni tim

Aktori i njihovi funkcionalni zahtjevi:

1. Neregistrirani/neprijavljeni korisnik (inicijator) može:
 - (a) vidjeti osnovne informacije o kampu
 - (b) vidjeti vrijeme održavanja, trajanja i popis aktivnosti
 - (c) prijaviti se na prijavu za sudjelovanje u kampu ili na prijavu za animatora gdje upisuju svoje puno ime, e-mail adresu, broj telefona, datum i godinu rođenja i motivacijsko pismo
 - (d) sudionici koji imaju manje od 18 godina moraju unijeti i broj telefona odgovorne osobe
 - (e) biti obaviješten o tome je li njegova prijava prihvaćena putem e-maila
2. Sudionik (inicijator) može:

Nakon prijave u sustav prije početka kampa:

 - (a) prilikom registracije upisati lozinku za dobiveno korisničko ime
 - (b) vidjeti odbrojavanje do početka kampa
 - (c) kontaktirati organizatora

Nakon početka kampa:

 - (a) vidjeti stranicu s rasporedom ili agendom
 - (b) vidjeti popis članova svoje grupe i animatore i njihove kontakt podatke

- (c) vidjeti popis aktivnosti na kojima su sudjelovali i ocijeniti aktivnost (1-10) i ostaviti kratak opis dojma

Nakon završetka kampa:

- (a) ocijeniti cjelokupno iskustvo i ostaviti vlastiti dojam

3. Animator (inicijator) može:

Nakon prijave u sustav prije početka kampa:

- (a) prilikom registracije upisati lozinku za dobiveno korisničko ime
- (b) vidjeti odbrojanje do početka kampa
- (c) kontaktirati organizatora

Nakon početka kampa:

- (a) vidjeti stranicu s rasporedom ili agendom
- (b) vidjeti popis svih grupa, njihovih članova i popis drugih animatora te njihove kontakt podatke
- (c) vidjeti popis aktivnosti na kojima su animirali i ocijeniti aktivnost (1-10) i ostaviti kratak opis dojma

Nakon završetka kampa:

- (a) ocijeniti cjelokupno iskustvo i ostaviti vlastiti dojam

4. Organizator (inicijator) može:

- (a) zadati početne informacije o kampu (vrijeme održavanja, trajanje, aktivnosti)
- (b) jednostavno definirati aktivnost (ime, kratki opis, trajanje)
- (c) omogućiti prijave za kamp i odrediti vrijeme trajanja prijavi
- (d) vidjeti popis prijavi te ih odbiti ili prihvatiti
- (e) nakon završetka odabira prijavi određivati broj grupa u koje će sudionici biti raspoređeni
- (f) popuniti raspored s aktivnostima uz određene uvjete i aktivnostima pridružiti grupe
- (g) vidjeti popis svih povratnih ocjena po aktivnostima
- (h) pretraživati povratne ocjene po atributima korisnik, grupa i/ili aktivnost

5. Administrator (inicijator) može:

- (a) vidjeti popis svih registriranih korisnika

- (b) brisati i dodavati korisnike i davati im ovlaštenja i mijenjati im razinu pristupa (organizator, sudionik, animator)
- (c) slučajnim odabirom rasporediti sudionike u grupe
- (d) premještati sudionika iz grupe u grupu
- (e) nakon prihvaćene prijave stvoriti korisnički račun i poslati e-mail korisniku
- (f) ako je odbijena prijava obavijestiti korisnika e-mailom
- (g) micati dojmove čije riječi nisu sukladne pravilima korištenja stranice

6. Baza podataka (sudionik):

- (a) pohranjuje sve podatke o korisnicima i njihovim ovlastima
- (b) pohranjuje sve podatke o aktivnostima i grupama
- (c) pohranjuje krajnje dojmove

3.1.1 Obrasci uporabe

Opis obrazaca uporabe

UC<1> - <Organizacija kampa>

- **Glavni sudionik:** <Organizator>
- **Cilj:** <Postaviti osnovne informacije o kampu>
- **Sudionici:** <Baza podataka>
- **Preduvjet:** <Organizator mora biti evidentiran u bazi>
- **Opis osnovnog tijeka:**
 1. <Organizator se prijavljuje kroz web aplikaciju>
 2. <Organizator kroz sučelje za izradu kampa unosi podatke>
 3. <Organizator odabire gumb kreacije kampa>

UC<2> - <Kreiranje aktivnosti>

- **Glavni sudionik:** <Organizator>
- **Cilj:** <Napraviti novu aktivnost>
- **Sudionici:** <Baza podataka>
- **Preduvjet:** <->
- **Opis osnovnog tijeka:**
 1. <Organizator odabire sučelje za kreaciju nove aktivnosti>
 2. <Organizator odabire detalje nove aktivnosti>
 3. <Detalji nove aktivnosti se prikazuju organizatoru koji ih može potvrditi te tako završiti proces kreacije >

UC<3> - <Zadavanje trajanja prijave>

- **Glavni sudionik:** <Organizator>
- **Cilj:** <Odrediti vrijeme prijave za sudionike i animatore>
- **Sudionici:** <Baza podataka>
- **Preduvjet:** <Organizator mora biti evidentiran u bazi>
- **Opis osnovnog tijeka:**
 1. <Organizator odabire sučelje za određivanje trajanja prijave>
 2. <Organizator odabire vrijeme trajanja prijave i za koga je prijava namijenjena>
 3. <Otvara se mogućnost prijave u zadanom terminu>

UC<4> - <Prijava za sudjelovanje>

- **Glavni sudionik:** <Potencijalni sudionik (neregistrirani korisnik)>
- **Cilj:** <Prijava za sudjelovanje na kampu>
- **Sudionici:** <Baza podataka>
- **Preduvjet:** <->
- **Opis osnovnog tijeka:**
 1. <Korisnik odabire sučelje za prijavu za sudionika na kamp>
 2. <Korisnik upisuje tražene podatke>
 3. <Prijava se prikazuje organizatoru koji dalje može prihvatiti ili odbiti prijavu>

UC<5> - <Potvrda prijave sudionika>

- **Glavni sudionik:** <Organizator>
- **Cilj:** <Potvrditi prijavu sudionika>
- **Sudionici:** <Baza podataka>
- **Preduvjet:** <Sudionik mora poslati prijavu>
- **Opis osnovnog tijeka:**
 1. <Organizator prihvaća prijavu>
 2. <Organizator šalje prihvaćenom sudioniku link za registraciju na e-mail naveden u prijavi>

UC<6> - <Odbijanje prijave sudionika>

- **Glavni sudionik:** <Organizator>
- **Cilj:** <Odbiti prijavu sudionika>
- **Sudionici:** <Baza podataka>
- **Preduvjet:** <Sudionik mora poslati prijavu>
- **Opis osnovnog tijeka:**
 1. <Organizator odbija prijavu>
 2. <Organizator šalje sudioniku obavijest o odbijanju na e-mail naveden u prijavi>

UC<7> - <Prijava za animatore>

- **Glavni sudionik:** <Potencijalni animator (neregistrirani korisnik)>
- **Cilj:** <Prijava za posao animatora na kampu>

- **Sudionici:** <Baza podataka>
- **Preduvjet:** <->
- **Opis osnovnog tijeka:**
 1. <Korisnik odabire sučelje za prijavu za animatora na kamp>
 2. <Korisnik upisuje tražene podatke>
 3. <Prijava se prikazuje organizatoru koji dalje može prihvatiti ili odbiti prijavu>

UC<8> - <Registracija sudionika>

- **Glavni sudionik:** <Sudionik>
- **Cilj:** <Registracija sudionika>
- **Sudionici:** <Baza podataka>
- **Preduvjet:** <Prihvaćena prijava od strane organizatora>
- **Opis osnovnog tijeka:**
 1. <Korisnik dobiva mail sa podacima za registraciju>
 2. <Korisnik upisuje proizvoljnu lozinku na sučelju za registraciju>
 3. <Registracija se evidentira u bazi podataka>

UC<9> - <Razmještaj sudionika po grupama>

- **Glavni sudionik:** <Organizator>
- **Cilj:** <Razmjestiti sudionike po grupama>
- **Sudionici:** <Baza podataka>
- **Preduvjet:** <Organizator je završio odabir prijava>
- **Opis osnovnog tijeka:**
 1. <Organizator određuje broj grupa za sudionike>
 2. <Grupa svakog sudionika se evidentira u bazi podataka>

UC<10> - <Zahtjev za premještanje>

- **Glavni sudionik:** <Sudionik>
- **Cilj:** <Zatražiti premještanje u drugu grupu>
- **Sudionici:** <Baza podataka>
- **Preduvjet:** <Sudionici su raspoređeni po grupama>
- **Opis osnovnog tijeka:**
 1. <Sudionik odabire opciju zahtjeva za premještanje >

2. <Sudioniku se izlistaju ostale grupe>
3. <Sudionik odabire željenu grupu>

UC<11> - <Premještaj pojedinog sudionika u drugu grupu>

- **Glavni sudionik:** <Organizator>
- **Cilj:** <Premjestiti sudionika u drugu grupu>
- **Sudionici:** <Baza podataka>
- **Preduvjet:** <Zahtjev za premještaj>
- **Opis osnovnog tijeka:**
 1. <Organizator zaprima zahtjev za premještaj sudionika u drugu grupu>
 2. <Premještaj se evidentira u bazi podataka>

UC<12> - <Punjenje rasporeda i provjera aktivnosti>

- **Glavni sudionik:** <Organizator>
- **Cilj:** <Popuniti raspored s aktivnostima i provjeriti kršenje uvjeta aktivnosti>
- **Sudionici:** <Baza podataka>
- **Preduvjet:** <Formirane su grupe>
- **Opis osnovnog tijeka:**
 1. <Aktivnost se neće preklapati s aktivnošću istog tipa>
 2. <Pridružen je minimalno jedan animator>
 3. <Pridružen je odgovarajući broj grupa>
 4. <Nijedna od pridruženih grupa neće imati konflikte s drugim aktivnostima koje su već navedene>
 5. <Nijedna od pridruženih grupa nije već pridružena jednakoj aktivnosti>
 6. <Pridruženi animatori neće imati konflikte s drugim aktivnostima na koje su pridruženi>

UC<13> - <Ocjenjivanje aktivnosti>

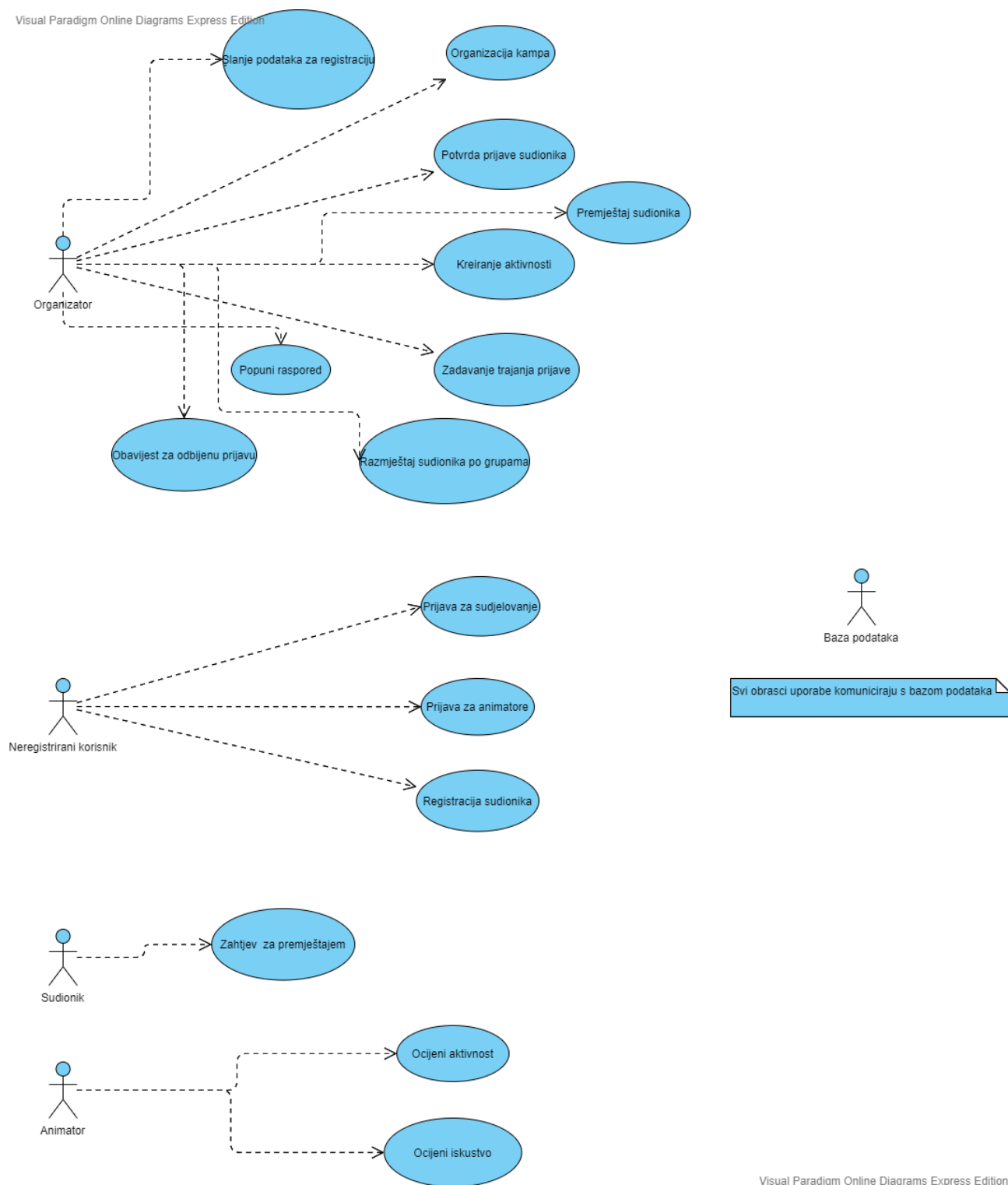
- **Glavni sudionik:** <Sudionici i animatori>
- **Cilj:** <Dati povratnu ocjenu za aktivnosti>
- **Sudionici:** <Baza podataka>
- **Preduvjet:** <Aktivnost je završila>
- **Opis osnovnog tijeka:**
 1. <Sudionici i organizatori ocjenjuju aktivnost>

2. <Ocjene se evidentiraju u bazi podataka>
3. <Organizator može pretraživati ocjene>

UC<14> - <Ocjenjivanje cjelokupnog iskustva>

- **Glavni sudionik:** <Sudionici i animatori>
- **Cilj:** <Dati povratnu ocjenu za cjelokupno iskustvo>
- **Sudionici:** <Baza podataka>
- **Preduvjet:** <Kamp je završio>
- **Opis osnovnog tijeka:**
 1. <Sudionici i organizatori ocjenjuju cjelokupno iskustvo>
 2. <Ocjene se evidentiraju u bazi podataka>
 3. <Organizator može pretraživati ocjene>

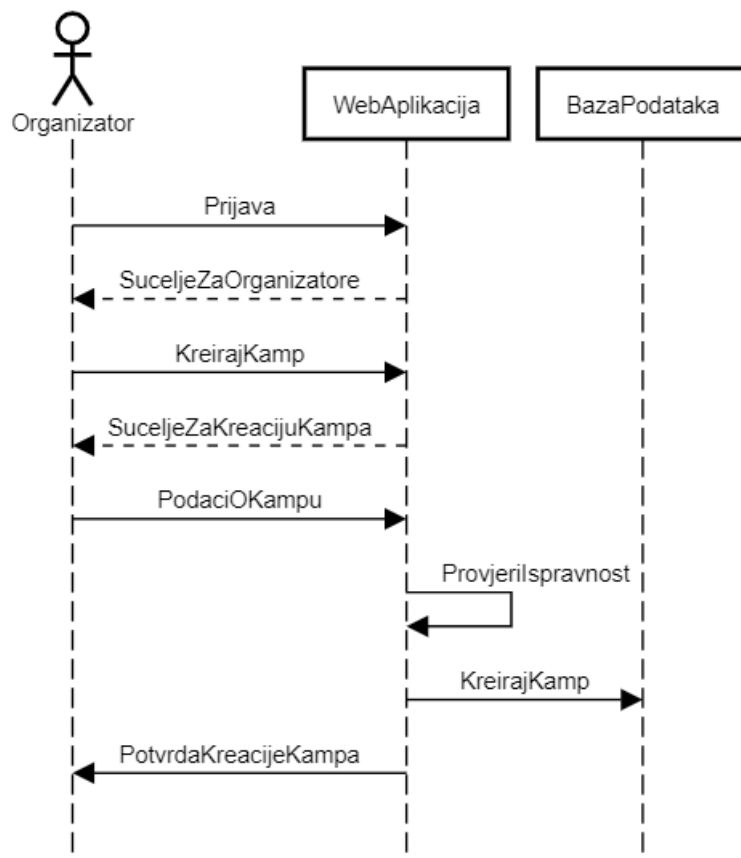
Dijagrami obrazaca uporabe



Slika 3.1: Dijagram obrazaca uporabe

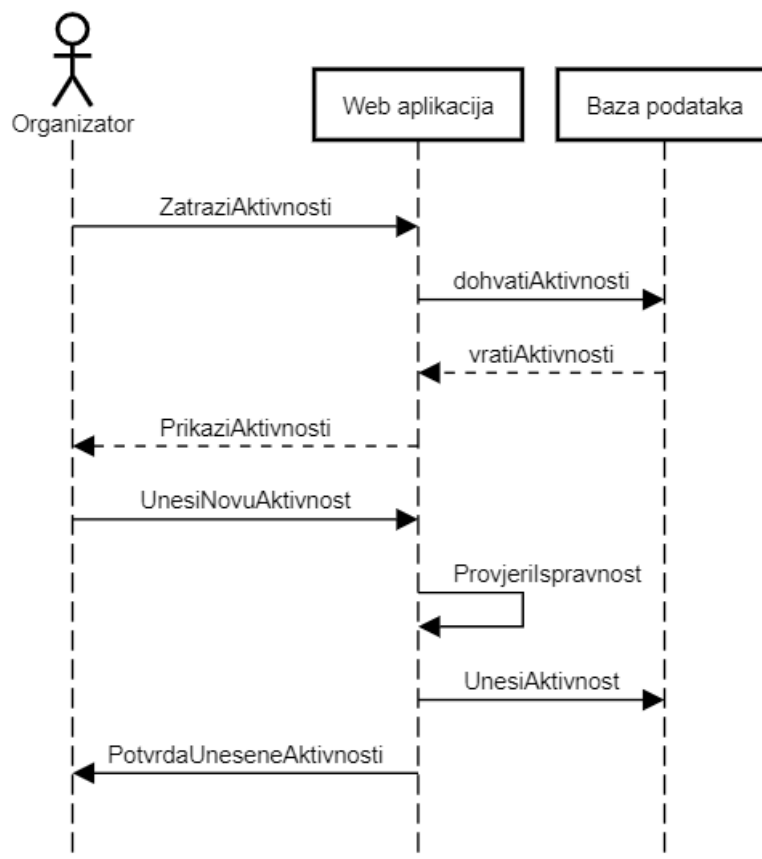
3.1.2 Sekvencijski dijagrami

Obrazac upotrebe UC1 - Organizacija kampa



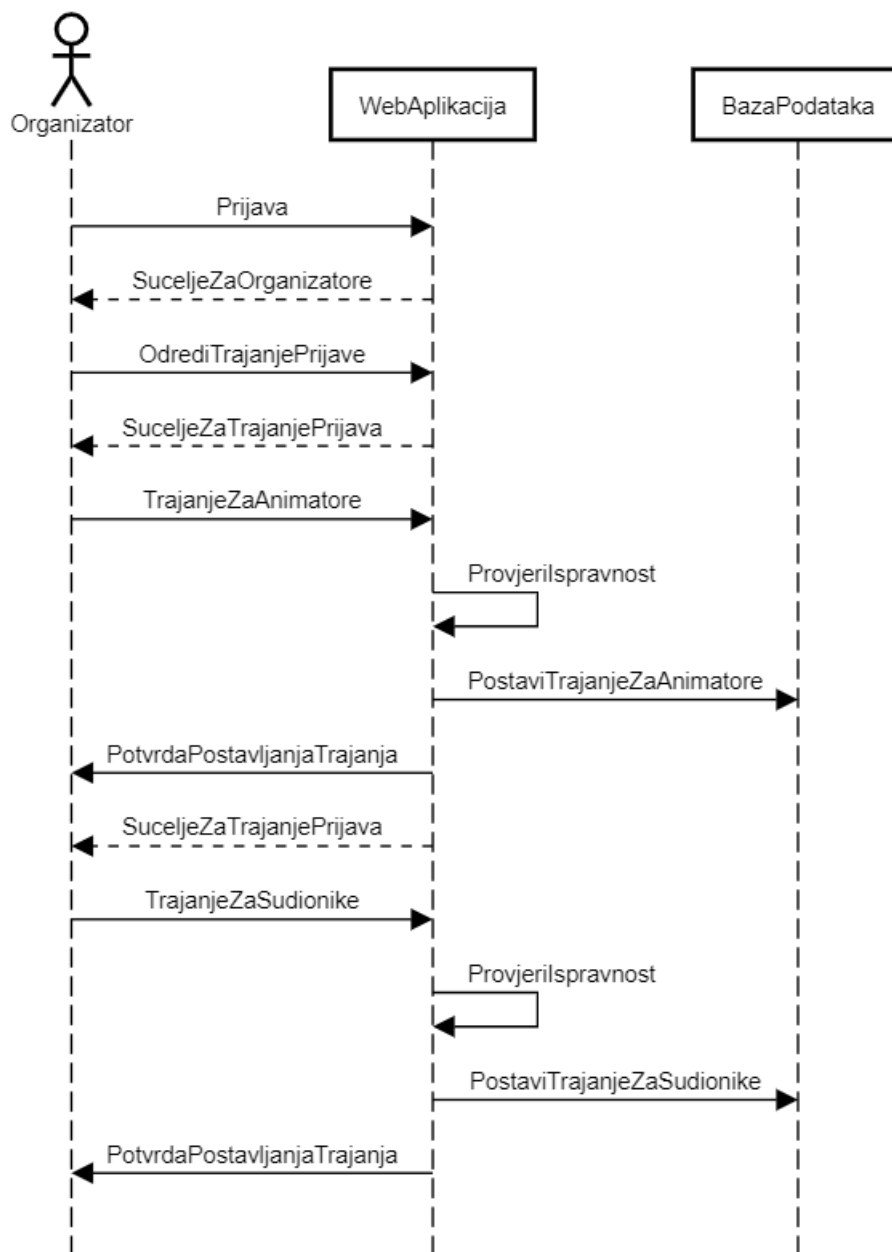
Slika 3.2: Sekvencijski dijagram za UC1

Obrazac upotrebe UC2 - Kreiranje aktivnosti



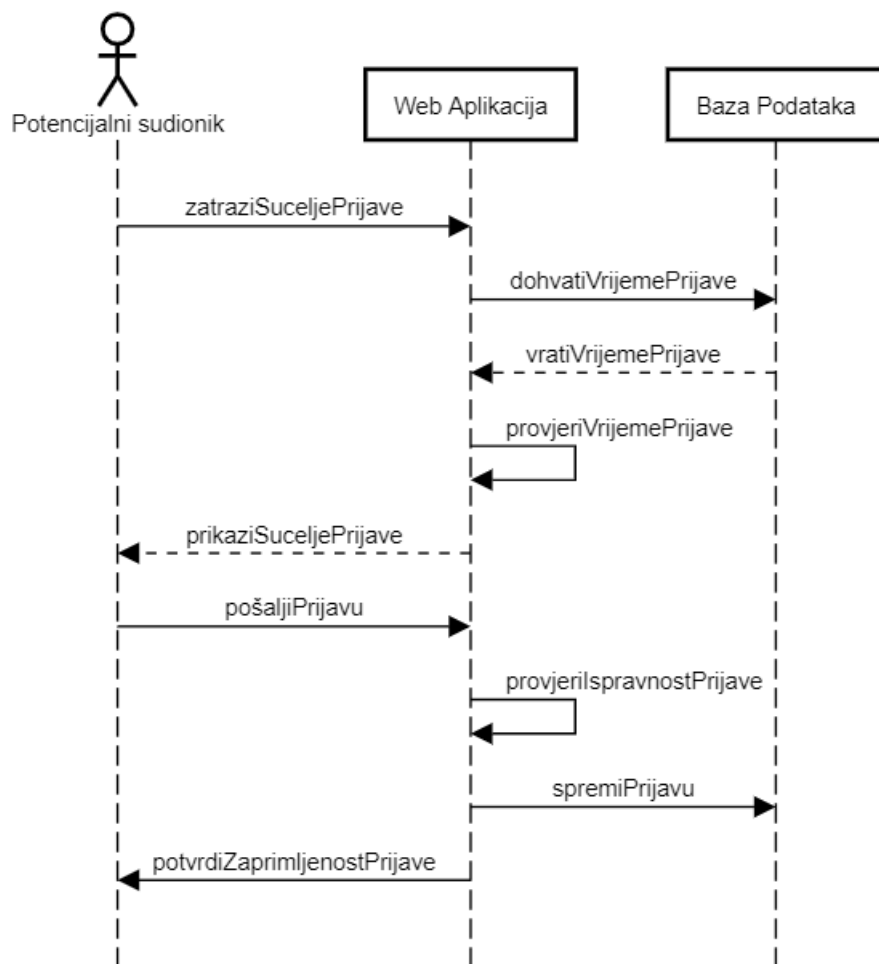
Slika 3.3: Sekvencijski dijagram za UC2

Obrazac upotrebe UC3 - Zadavanje trajanja prijave



Slika 3.4: Sekvencijski dijagram za UC3

Obrazac upotrebe UC4 - Prijava za sudjelovanje



Slika 3.5: Sekvencijski dijagram za UC4

3.2 Ostali zahtjevi

- sustav treba biti implementiran kao web aplikacija koristeći objektno-orijentirane jezike
- korisničko sučelje i sustav moraju podržavati hrvatsku abecedu (dijakritičke znakove) pri unosu i prikazu tekstualnog sadržaja
- nadogradnja sustava ne smije narušavati postojeće funkcionalnosti sustava
- neispravno korištenje korisničkog sučelja ne smije narušiti funkcionalnost i rad sustava
- sustav treba biti jednostavan za korištenje, korisnici se moraju znati koristiti sučeljem bez opširnih uputa
- sustav treba omogućiti rad više korisnika u stvarnom vremenu
- izvršavanje dijela programa u kojem se pristupa bazi podataka ne smije trajati duže od nekoliko sekundi
- pristup sustavu mora biti omogućen iz javne mreže

4. Arhitektura i dizajn sustava

Arhitektura se može podijeliti na tri podsustava:

- web poslužitelj
- web aplikacija
- baza podataka

Web preglednik je program koji korisniku omogućuje pregled web-stranica i multimedijalnih sadržaja vezanih uz njih. Svaka stranica pisana je u kodu, a web preglednik je pretvara u ono što mi vidimo. Dakle, svaki internetski preglednik je prevoditelj. Korisnik putem web preglednika šalje zahtjev web poslužitelju.

Web poslužitelj osnova je rada web aplikacije. On šalje i prima podatke od mnogostrukih klijenata. Komunikacija između njega i korisnika se odvija preko HTTP (engl. Hyper Text Transfer Protocol) protokola, što je protokol u prijenosu informacija na webu. Poslužitelj je onaj koji pokreće web aplikaciju te joj prosljeđuje zahtjev.

Za obrađivanje željenih zahtjeva koristi se web aplikacija. Ako je potrebno pristupa bazi podataka te preko poslužitelja korisniku vraća odgovor u obliku HTML dokumenta vidljivog u web pregledniku.

Programski jezik koji smo odabrali za izradu naše web aplikacije je JavaScript zajedno sa Bootstrap radnim okvirom te HTML i CSS programske jezike za oblikovanje. Oda-brano razvojno okruženje programske potpore je Visual studio code.

Temelj arhitekture sustava ležat će na MVC (Model-View-Controller) konceptu. Naime, taj koncept ima već napravljene predloške koji nam pomažu u izradi web aplikacije te je podržan od radnog okvira. Velika prednost MVC koncepta je da omogućuje programeru da razvija komponente aplikacije nezavisno jedne o drugima, što olakšava testiranje, traženje grešaka i dodavanje novih funkcionalnosti.

MVC koncept sastoji se od 3 dijela:

- Model – centralni dio sustava koji direktno upravlja podacima, logikom i pravilima sustava. Isto tako prima ulazne podatke Controllera
- View – glavna uloga mu je da prikazuje podatke. Ista informacija može se prikazati na nekoliko različitih načina, poput grafova, tablica i sl.
- Controller – bavi se prilagodbom ulaza koje prosljeđuje Modelu i Viewu te upravlja zahtjevima korisnika i pomoću njih djeluje na ostale sustave.

4.1 Baza podataka

Za potrebe našeg sustava za organizaciju kampa računarstva „Mlade nade” koristimo relacijsku bazu podataka. Relacijska baza sastoji se od relacija, tj. tablica koje sadrže naziv i skup atributa. Ovakva baza nam omogućuje brzu i jednostavnu pohranu i izmjenu podataka te dohvat podataka za daljnju obradu. Dijagram baze olakšava razumijevanje namjene podataka i njihove povezanosti.

Baza podataka ove aplikacije sastoji se od sljedećih entiteta:

- Osoba
 - * Animator
 - * Sudionik
 - * Organizator
- Aktivnost
 - * Aktivnost1
 - * AktivnostSve
 - * AktivnostMaxN
 - * AktivnostN
- Grupa
- Sudjeluje
- Račun
- Dojam
- Prijava
- Kamp

4.1.1 Opis tablica

osoba - ovaj entitet sadrži podatke o osobama koje na bilo koji način sudjeluju u kampu. Sadrži attribute: puno ime osobe, ID osobe, motivacijsko pismo, datum rođenja, broj telefona odgovorne osobe (za sudionike mlađe od 18 godina), e-mail i broj telefona. Generalizacija je entiteta organizator, animator i sudionik. U vezi je one-to-one s entitetom dojam preko atributa Idosobe.

osoba		
Idosobe	INT	identifikacijski broj osobe
punoIme	VARCHAR	ime i prezime osobe
motPismo	VARCHAR	motivacijsko pismo
datumRod	DATE	datum rođenja osobe
brojTelefonaOdgOsobe	VARCHAR	broj telefona odgovorne osobe
Email	VARCHAR	e-mail osobe
brojTel	VARCHAR	broj telefona osobe

animator - ovaj entitet sadrži podatke o animatorima koji sudjeluju u raznim aktivnostima. Specijalizacija je entiteta osoba. Uz attribute entiteta osoba sadrži još i entitet naziv aktivnosti. U vezi je many-to-one s entitetom aktivnost preko atributa nazivAkt.

animator		
Idosobe	INT	identifikacijski broj osobe
nazivAkt	VARCHAR	naziv aktivnosti u kojoj sudjeluje
punoIme	VARCHAR	ime i prezime osobe
motPismo	VARCHAR	motivacijsko pismo
datumRod	DATE	datum rođenja osobe
brojTelefonaOdgOsobe	VARCHAR	broj telefona odgovorne osobe
Email	VARCHAR	e-mail osobe
brojTel	VARCHAR	broj telefona osobe

sudionik - ovaj entitet sadrži podatke o sudionicima kampa. Specijalizacija je entiteta osoba. Uz attribute entiteta osoba sadrži još i entitet naziv grupe. U vezi je many-to-one s entitetom grupa preko atributa nazivGrupa.

Sudionik		
Idosobe	INT	identifikacijski broj osobe
nazivGrupa	VARCHAR	naziv grupe čiji je član
punoIme	VARCHAR	ime i prezime osobe
motPismo	VARCHAR	motivacijsko pismo
datumRod	DATE	datum rođenja osobe
brojTelefonaOdgOsobe	VARCHAR	broj telefona odgovorne osobe
Email	VARCHAR	e-mail osobe
brojTel	VARCHAR	broj telefona osobe

organizator - ovaj entitet sadrži informacije o osobi koja organizira kamp. Specijalizacija je entiteta osoba. Uz attribute entiteta osoba sadrži još i entitet naziv-Grupa.

organizator		
Idorganizatora	INT	identifikacijski broj organizatora
punoIme	VARCHAR	ime i prezime osobe
motPismo	VARCHAR	motivacijsko pismo
datumRod	DATE	datum rođenja osobe
brojTelefonaOdgOsobe	VARCHAR	broj telefona odgovorne osobe
Email	VARCHAR	e-mail osobe
brojTel	VARCHAR	broj telefona osobe

aktivnost - ovaj entitet sadrži podatke o aktivnostima koje se odvijaju u kampu. Generalizacija je entiteta aktivnost1, aktivnostSve, aktivnostMaxN i aktivnostN. Sadrži attribute naziv aktivnosti, opis i trajanje aktivnosti. U vezi je many-to-one s entitetom sudjeluje preko atributa nazivAkt.

aktivnost		
nazivAkt	VARCHAR	Naziv aktivnosti
opis	VARCHAR	Kratak opis aktivnosti
trajanje	INTERVAL	Trajanje aktivnosti

aktivnost1 - ovaj entitet sadrži podatke o aktivnostima u kojima sudjeluje samo jedna grupa sudionika. Specijalizacija je entiteta aktivnost. Sadrži sve attribute entiteta aktivnost tj. naziv aktivnosti, opis i trajanje.

aktivnost1		
nazivAkt	VARCHAR	naziv aktivnosti
opis	VARCHAR	kratak opis aktivnosti
trajanje	INTERVAL	trajanje aktivnosti

aktivnostSve - ovaj entitet sadrži podatke o aktivnostima u kojima sudjeluju sve postojeće grupe sudionika. Specijalizacija je entiteta aktivnost. Sadrži sve attribute entiteta aktivnost tj. naziv aktivnosti opis i trajanje.

aktivnostSve		
nazivAkt	VARCHAR	naziv aktivnosti
opis	VARCHAR	kratak opis aktivnosti
trajanje	INTERVAL	trajanje aktivnosti

aktivnostMaxN - ovaj entitet sadrži podatke o aktivnostima u kojima sudjeluje maksimalno N grupa sudionika. Specijalizacija je entiteta aktivnost. Sadrži sve attribute entiteta aktivnost tj. naziv aktivnosti, opis i trajanje.

aktivnostMaxN		
nazivAkt	VARCHAR	naziv aktivnosti
opis	VARCHAR	kratak opis aktivnosti
trajanje	INTERVAL	trajanje aktivnosti

aktivnostN - ovaj entitet sadrži podatke o aktivnostima u kojima sudjeluje točno N grupa sudionika. Specijalizacija je entiteta aktivnost. Sadrži sve attribute entiteta aktivnost tj. naziv aktivnosti, opis i trajanje.

aktivnostN		
nazivAkt	VARCHAR	naziv aktivnosti
opis	VARCHAR	kratak opis aktivnosti
trajanje	INTERVAL	trajanje aktivnosti

grupa - ovaj entitet opisuje grupe sudionika koji sudjeluju u aktivnostima. Sadrži atribut naziv grupe. U vezi je many-to-one s entitetom sudjeluje preko atributa nazivGrupa.

grupa		
nazivGrupa	VARCHAR	naziv grupe sudionika

sudjeluje - ovaj entitet sadrži sve informacije o odnosu grupa i aktivnosti, odnosno daje informacije o tome koja grupa sudjeluje u kojoj od aktivnosti. Sadrži atribut naziv grupe te je preko tog atributa u vezi one-to-many s entitetom sudionik.

sudjeluje		
nazivGrupa	VARCHAR	naziv grupe sudionika
nazivAkt	VARCHAR	naziv aktivnosti

račun - ovaj entitet sadrži podatke o korisničkom računu koji je se izrađuje na temelju prihvaćene prijave za svakog sudionika i animatora. Sadrži attribute korisnickoIme, lozinka, poveznica i Idosobe. U vezi je one-to-one s entitetom osoba preko atributa Idosobe.

račun		
korisnickoIme	VARCHAR	korisničko ime osobe
Idosobe	INT	identifikacijski broj osobe
lozinka	VARCHAR	lozinka korisničkog računa
poveznica	VARCHAR	poveznica za registraciju

dojam - ovaj entitet sadrži informacije o osvrtu i ocjeni koju sudionici i animatori ostavljaju nakon završetka kampa. Sadrži entitete ID osobe, ocjena, komentar. U vezi je one-to-one s entitetom osoba preko atributa Idosobe.

dojam		
Idosobe	INT	identifikacijski broj osobe
ocjena	INT	ocjena kampa
komentar	VARCHAR	kratki osvrt

prijava - ovaj entitet sadrži podatke o prijavama za sudjelovanje u kampu. Sadrži attribute prijavaZa (određuje prijavljuje li se osoba za sudionika ili animatora), vrijeme početka prijave i vrijeme trajanja prijave.

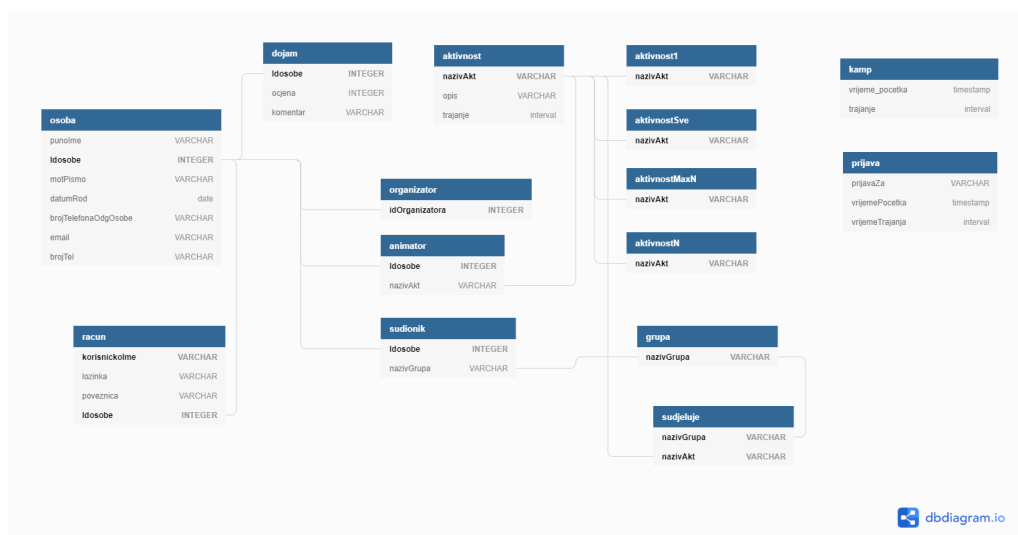
prijava		
prijavaZa	VARCHAR	prijava za sudionika/animatora
vrijemePocetka	TIMESTAMP	vrijeme početka prijave
vrijemeTrajanja	INTERVAL	trajanje prijave

kamp - ovaj entitet sadrži informacije o kampu za računarstvo za koji se izrađuje aplikacija. Sadrži attribute vrijemePocetka i vrijemeTrajanje.

kamp		
vrijemePocetka	TIMESTAMP	vrijeme početka kampa
vrijemeTrajanja	INTERVAL	trajanje kampa

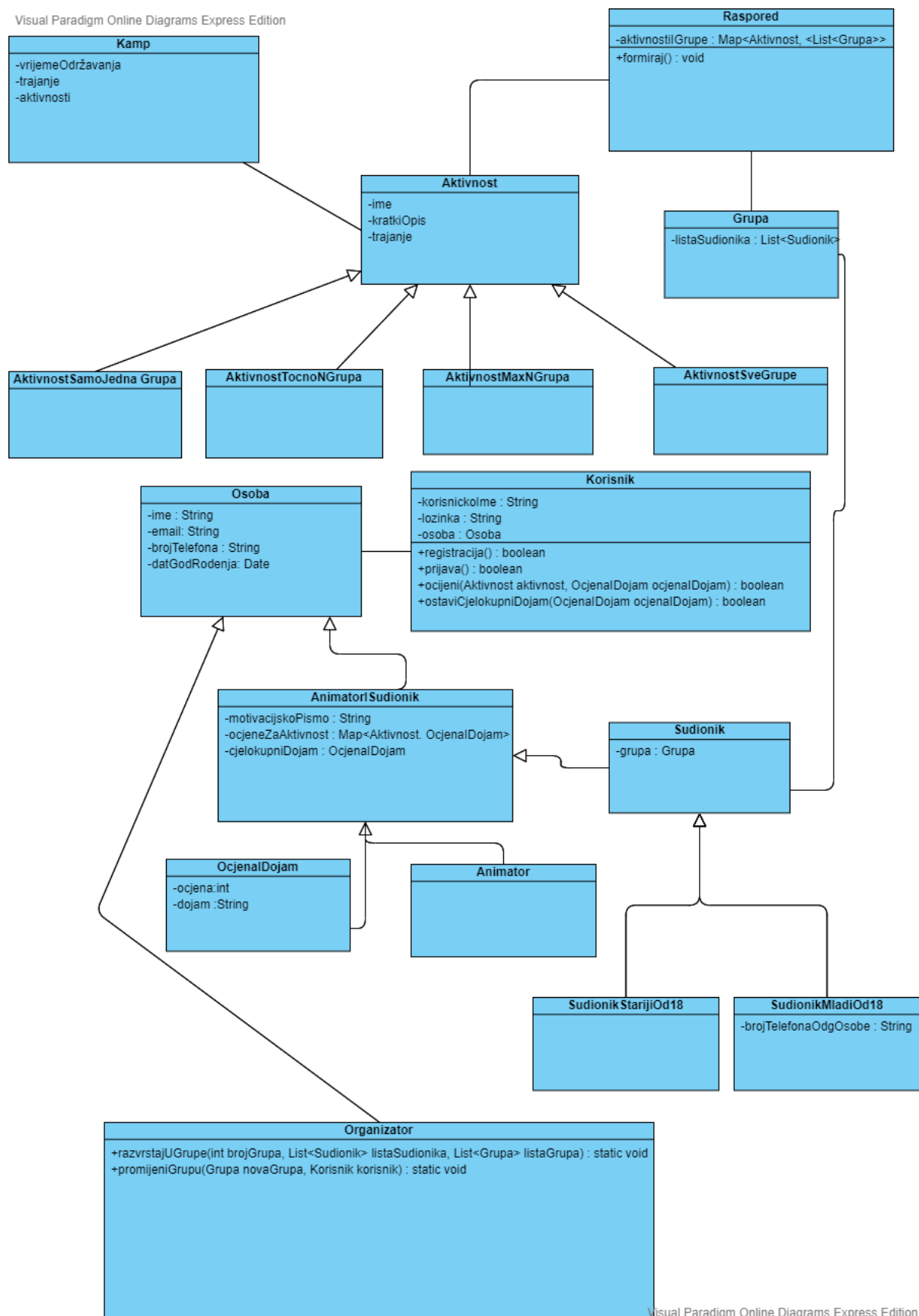
4.1.2 Dijagram baze podataka

Dijagram baze podataka



Slika 4.1: ER dijagram baze podataka

4.2 Dijagram razreda



dio 2. revizije

Prilikom druge predaje projekta dijagram razreda i opisi moraju odgovarati stvarnom stanju implementacije

4.3 Dijagram stanja

dio 2. revizije

*Potrebno je priložiti dijagram stanja i opisati ga. Dovoljan je jedan dijagram stanja koji prikazuje **značajan dio funkcionalnosti** sustava. Na primjer, stanja korisničkog sučelja i tijekom korištenja neke ključne funkcionalnosti jesu značajan dio sustava, a registracija i prijava nisu.*

4.4 Dijagram aktivnosti

dio 2. revizije

Potrebno je priložiti dijagram aktivnosti s pripadajućim opisom. Dijagram aktivnosti treba prikazivati značajan dio sustava.

4.5 Dijagram komponenti

dio 2. revizije

Potrebno je priložiti dijagram komponenti s pripadajućim opisom. Dijagram komponenti treba prikazivati strukturu cijele aplikacije.

5. Implementacija i korisničko sučelje

5.1 Korištene tehnologije i alati

dio 2. revizije

*Detaljno navesti sve tehnologije i alate koji su primijenjeni pri izradi dokumentacije i aplikacije. Ukratko ih opisati, te navesti njihovo značenje i mjesto primjene. Za svaki navedeni alat i tehnologiju je potrebno **navesti internet poveznicu** gdje se mogu preuzeti ili više saznati o njima.*

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

*Potrebno je provesti ispitivanje jedinica (engl. unit testing) nad razredima koji implementiraju temeljne funkcionalnosti. Razraditi **minimalno 6 ispitnih slučajeva** u kojima će se ispitati redovni slučajevi, rubni uvjeti te izazivanje pogreške (engl. exception throwing). Poželjno je stvoriti i ispitni slučaj koji koristi funkcionalnosti koje nisu implementirane. Potrebno je priložiti izvorni kôd svih ispitnih slučajeva te prikaz rezultata izvođenja ispita u razvojnom okruženju (prolaz/pad ispita).*

5.2.2 Ispitivanje sustava

*Potrebno je provesti i opisati ispitivanje sustava koristeći radni okvir Selenium¹. 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.

¹<https://www.seleniumhq.org/>

5.3 Dijagram razmještaja

dio 2. revizije

*Potrebno je umetnuti **specifikacijski** dijagram razmještaja i opisati ga. Moguće je umjesto specifikacijskog dijagrama razmještaja umetnuti dijagram razmještaja instanci, pod uvjetom da taj dijagram bolje opisuje neki važniji dio sustava.*

5.4 Upute za puštanje u pogon

dio 2. revizije

*U ovom poglavlju potrebno je dati upute za puštanje u pogon (engl. deployment) ostvarene aplikacije. Na primjer, za web aplikacije, opisati postupak kojim se od izvornog kôda dolazi do potpuno postavljene baze podataka i poslužitelja koji odgovara na upite korisnika. Za mobilnu aplikaciju, postupak kojim se aplikacija izgradi, te postavi na neku od trgovina. Za stolnu (engl. desktop) aplikaciju, postupak kojim se aplikacija instalira na računalo. Ukoliko mobilne i stolne aplikacije komuniciraju s poslužiteljem i/ili bazom podataka, opisati i postupak njihovog postavljanja. Pri izradi uputa preporučuje se **naglasiti korake instalacije uporabom natuknica** te koristiti što je više moguće **slike ekrana** (engl. screenshots) kako bi upute bile jasne i jednostavne za slijediti.*

Dovršenu aplikaciju potrebno je pokrenuti na javno dostupnom poslužitelju. Studentima se preporuča korištenje neke od sljedećih besplatnih usluga: Amazon AWS, Microsoft Azure ili Heroku. Mobilne aplikacije trebaju biti objavljene na F-Droid, Google Play ili Amazon App trgovini.

6. Zaključak i budući rad

dio 2. revizije

U ovom poglavlju potrebno je napisati osvrt na vrijeme izrade projektnog zadatka, koji su tehnički izazovi prepoznati, jesu li riješeni ili kako bi mogli biti riješeni, koja su znanja stečena pri izradi projekta, koja bi znanja bila posebno potrebna za brže i kvalitetnije ostvarenje projekta i koje bi bile perspektive za nastavak rada u projektnoj grupi.

Potrebno je točno popisati funkcionalnosti koje nisu implementirane u ostvarenoj aplikaciji.

Popis literature

Kontinuirano osvježavanje

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

1. Programsko inženjerstvo, FER ZEMRIS, <http://www.fer.hr/predmet/proinz>
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

Dodatak: Prikaz aktivnosti grupe

Dnevnik sastajanja

Kontinuirano osvježavanje

U ovom dijelu potrebno je redovito osvježavati dnevnik sastajanja prema predlošku.

1. sastanak

- Datum: u ovom formatu: 13. studenoga 2020.
- Prisustvovali: I.Prezime, I.Prezime
- Teme sastanka:
 - * opis prve teme
 - * opis druge teme

2. sastanak

- Datum: u ovom formatu: 13. studenoga 2020.
- Prisustvovali: I.Prezime, I.Prezime
- Teme sastanka:
 - * opis prve teme
 - * opis druge teme

Tablica aktivnosti

Kontinuirano osvježavanje

Napomena: Doprinosi u aktivnostima treba navesti u satima po članovima grupe po aktivnosti.

	Ime Prezime voditelja	Ime Prezime	Ime Prezime	Ime Prezime	Ime Prezime	Ime Prezime	Ime Prezime
Upravljanje projektom							
Opis projektnog zadatka							
Funkcionalni zahtjevi							
Opis pojedinih obrazaca							
Dijagram obrazaca							
Sekvencijski dijagrami							
Opis ostalih zahtjeva							
Arhitektura i dizajn sustava							
Baza podataka							
Dijagram razreda							
Dijagram stanja							
Dijagram aktivnosti							
Dijagram komponenti							
Korištene tehnologije i alati							
Ispitivanje programskog rješenja							
Dijagram razmještaja							
Upute za puštanje u pogon							
Dnevnik sastajanja							
Zaključak i budući rad							
Popis literature							
<i>Dodatne stavke kako ste podijelili izradu aplikacije</i>							

	Ime Prezime voditelja	Ime Prezime	Ime Prezime	Ime Prezime	Ime Prezime	Ime Prezime	Ime Prezime
<i>npr. izrada početne stranice</i>							
<i>izrada baze podataka</i>							
<i>spajanje s bazom podataka</i>							
<i>back end</i>							

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.