SVEUČILIŠTE U ZAGREBU

FAKULTET ORGANIZACIJE I INFORMATIKE

V A R A Ž D I N

Dario Horvat

Tea Jarčov

Anabel Li Kečkeš

Matija Popijač

In4maticsQuiz

projektni rad IZ KOLEGIJA ANALIZA I RAZVOJ PROGRAMA

Varaždin, 2015.

SVEUČILIŠTE U ZAGREBU

FAKULTET ORGANIZACIJE I INFORMATIKE

V A R A Ž D I N

Dario Horvat, 44497/15-R

Smjer: Organizacija poslovnih sustava

Tea Jarčov 44442/15-R

Smjer: Organizacija poslovnih sustava

Anabel Li Kečkeš 44448/15-R

Smjer: Informacijsko i programsko inženjerstvo

Matija Popijač, 44441/15-R

Smjer: Informacijsko i programsko inženjerstvo

Diplomski studij

In4maticsQuiz

PROJEKTNI RAD IZ KOLEGIJA ANALIZA I RAZVOJ PROGRAMA

Mentor:

Doc. dr. sc. Zlatko Stapić

Varaždin, listopad 2015.

**Sadržaj**

[1. Uvod 1](#_Toc434510966)

[2. Korisnički zahtjevi 2](#_Toc434510967)

[2.2. Pismo namjere 3](#_Toc434510968)

[3. Specifikacija funkcionalnih zahtjeva 5](#_Toc434510969)

[4. Projektni plan 6](#_Toc434510970)

[4.1. Proračun projekta 7](#_Toc434510971)

[5. Metodologija razvoja 9](#_Toc434510972)

[5.1. Definiranje tima 10](#_Toc434510973)

[5.2. Praćenje izrade projekta kroz Quick Scrum alat 11](#_Toc434510974)

[5.2.1. Sprint 1 – detaljna specifikacija 14](#_Toc434510975)

[6. Dijagram slučajeva korištenja 16](#_Toc434510976)

[7. Arhitektura sustava 17](#_Toc434510977)

[8. Dijagram klasa 18](#_Toc434510978)

[9. ERA model 19](#_Toc434510979)

[10. Mockup – korisničko sučelje 20](#_Toc434510980)

[11. Webservis 20](#_Toc434510981)

# Uvod

Kroz kolegij Analiza i razvoj programa bavit ćemo se razvojem aplikacije koja je prvenstveno namijenjena I. osnovnoj školi Bjelovar, konkretno učenicima informatike. Aplikacija In4maticsQuiz bit će javno dostupna kako bi svi učenici imali pristup istoj. Ideja za aplikaciju nastala je na temelju komunikacije jednog od člana tima sa profesorom informatike. Ideja je olakšati učenje informatike kroz zabavan kviz uz povratne informacije o stečenom znanju u obliku bodovanja i rang listi. Aplikaciju ćemo razvijati za Android platformu jer su pametni telefoni android platforme jeftiniji i pristupačniji našoj ciljanoj grupi korisnika te je njihov broj veći u odnosu na druge platforme.

Ciljana grupa korisnika kojoj je namijenjena ova aplikacija je profesor informatike I. osnovne škole Bjelovar te njegovi učenici. Aplikaciju bi koristili učenici koji bi svoje znanje provjeravali kroz natjecanje i mogli bi se uspoređivati s drugima. Profesor bi aplikaciju koristio za dodavanje, ažuriranje i brisanje pitanja. Mogao bi pregledavati rang liste odigranih testova te sam odigrati kviz.

Zadana je agilna metodika razvoja, tj. metodologija Scrum po kojoj ćemo razvijati ovaj projekt.



# Korisnički zahtjevi

Da bi bili u mogućnosti aplikaciju kreirati sukladno potrebama korisnika, najvažnije je jasno i detaljno odrediti što korisnik aplikacije želi da mu bude omogućeno pri korištenju aplikacije. Definiranje korisničkih zahtjeva u većini metodika razvoja predstavlja jedan od temeljnih koraka u razvoju programskog proizvoda pa tako i u Scrum metodologiji razvoja. Bitno je upoznati se sa svim zahtjevima korisnika kako bi se mogli kreirati zadaci pomoću kojih se realiziraju funkcionalnosti programskog rješenja. Definiranjem korisničkih zahtjeva opisujemo što će aplikacija raditi, način na koji će raditi te kako će to pomoći krajnjim korisnicima u njihovom radu.

Korisnički zahtjevi za In4maticsQuiz mobilnu aplikaciju nastali su na temelju razgovora s učiteljem informatike i želje da uvede neke promijene u nastavu. Mobilna aplikacija bi učeniku trebala omogućiti pregled pitanja za gradivo koje se obradilo na nastavi i tako mu učinilo ponavljanje zanimljivijim. Isto tako bi aplikacija služila da učitelj ima neke povratne informacije od učenika jer bi tu aplikaciju mogao iskoristiti na satu i vidjet koliko je učenika shvatilo gradivo jer bi on, kao administrator, mogao kreirati, ažurirati i brisati već postojeća pitanja.

**Tablica 1. Popis korisničkih zahtjeva**

|  |  |  |
| --- | --- | --- |
| Zahtjev ID | Naziv | Opis zahtjeva |
| KZ 1 | Prijava u sustav | Učenik se treba prijaviti u sustav kako bi se mogao nalaziti na rang listama s odabranim nadimkom i kako bi se mogao potvrditi njegov identitet prema imenu i prezimenu. Prilikom prijave učitelja prepoznaje se da je on administrator i otvaraju mu se drugačije mogućnosti od onih što su ponuđene učeniku. |
| KZ 2 | Odabir razreda | Učeniku se nudi mogućnost prijave za koji razred želi odgovarati na pitanja. |
| KZ 3 | Odgovaranje na pitanja | Korisniku se postavljaju pitanja na koja on može odabrati jedan odgovor, više točnih odgovora ili upisati odgovor riječima. |
| KZ 4 | Pregled rang lista | Korisnici mogu pregledati gdje se nalaze na listi na temelju svojih odgovora. Rang liste koje se brišu nakon svakog dodavanja novih pitanja. |
| KZ 5 | Dodavanje novih pitanja | Administratoru je omogućeno dodavanje novih pitanja u aplikaciju |
| KZ 6 | Brisanje pitanja | Administratoru je omogućeno brisanje već postojećih pitanja u aplikaciji. |
| KZ 7 | Ažuriranje pitanja | Administratoru je omogućeno ažuriranje već postojećih pitanja u aplikaciji. |

## 2.2. Pismo namjere

Na sljedećoj slici nalazi se pismo namjere, tj. potvrda škole da je spremna surađivati s nama na ovom projektu.



**Slika 1. Pismo namjere**

# Specifikacija funkcionalnih zahtjeva

Osnovna funkcionalnost ove aplikacije je rješavanje zadataka. Nakon što učenik riješi zadatak biti će smješten na rang listu ovisno o točnosti i brzini rješavanja zadataka. Na rang listi će se naći sa svojim imenom koje je upisao prilikom registracije, a kasnije prilikom prijave. Učenici će birati za koji razred žele odgovarati na pitanja i rang liste će biti posebne za svaki razred. Administrator, to jest profesor, će moći kreirati nova pitanja, ažurirati ih i brisati već postojeća pitanja.

Aplikacija In4maticsQuiz će imati tri modula. Moduli će se razlikovati u načinu odgovora na postavljena pitanja, to jest na različite tipove pitanja:

1) pitanje u kojem učenik odgovara upisom točnog pojma

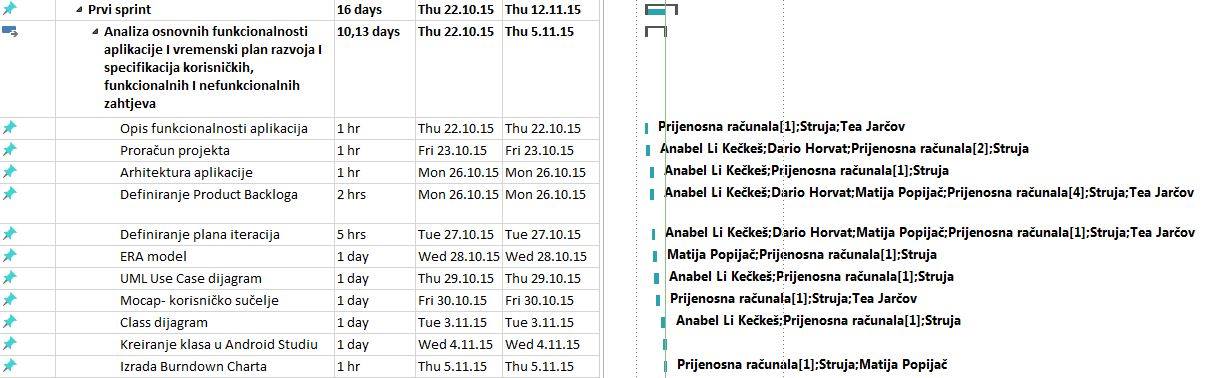
2) pitanje u kojem učenik odgovara odabirom jedne od ponuđenih opcija

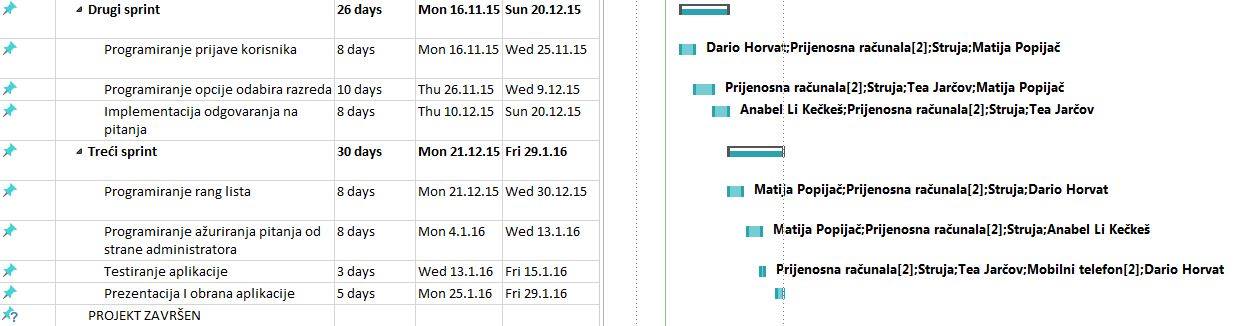
3) pitanje u kojem učenik označuje jedan ili više točnih odgovora

# Projektni plan

Projektni plan napravljen je u MS Project alatu.







**Slika 2. Projektni plan iz MS Projecta**

## 4.1. Proračun projekta

Prema podacima iz državnog zavoda za statistiku (2014.godina)  **plaća za programera** iznosi 9000kn (informacijska uslužna djelatnost). Nakon što smo taj iznos podijelili sa 20 (broj radnih dana u mjesecu) i sa 8 (broj sati u jednom radnom danu) dobili smo iznos od 56.25 kn po satu. No taj iznos ćemo modificirati tako da Anabel Li Kečkeš koja je Scrum master (brine se da je sve u skladu sa Scrum metodologijom) u ovom projektu dobiva **60kn** po satu, a ostali članovi tima (Tea Jarčov, Dario Horvat i Matija Popijač) dobivaju **50kn** po satu.

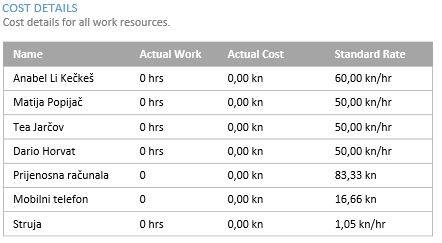
izvor: http://www.dzs.hr/Hrv\_Eng/publication/2014/09-01-01\_12\_2014.htm

Cijena **električne energije** po satu je 0,84kn bez PDV-a. Sa PDV-om je **1,05kn** po satu.

izvor: http://vijesti.hrt.hr/163266/tocna-cijena-kilovatsata

Ako svaki član tima ima **mobilni telefon** za 1000,00kn. Kada taj iznos podijelimo sa 20 (broj radnih dana u mjesecu ) i brojem mjeseci (3 mjeseca radimo aplikaciju)iznos će biti **16.66kn** po danu. Kako trebamo telefonirati ta cijena će biti 20,00kn po danu sa PDV-om.

Svaki član tima ima **prijenosno računalo** koje dođe 5000,00kn. Kada taj iznos podijelimo sa 20 (broj radnih dana u mjesecu) i sa brojem mjeseci (3 mjeseca radimo aplikaciju) iznos će biti **83,33kn** po danu sa PDV-om.



**Slika 3. Troškovi**

Slijedi slika grafa koji prikazuje našu procjenu troškova projekta. Navedena cijena je bez PDV-a.



**Slika 4. Slika troškova iz MS Projecta**

Cijena sa PDV-om (25%) je: 47.835,22kn + 11.958,81kn = 59.794,03kn

Troškovi zarade iznose 40 000,00kn pa je Ukupna cijena : 99.794,03kn.

Cijenu ćemo zaokružiti te će biti **100 000, 00kn**

# Metodologija razvoja

Kao što smo na početku naveli, metodologija razvoja našeg programskog proizvoda je Scrum metodologija. To je agilna metodologija za upravljanje razvojem softverskih proizvoda. Scrum je široko korištena metodologija koja se koristi za razvoj softverskih proizvoda, ali pogodna je za manje projekte. Ona podrazumijeva da se razvoj softverskog proizvoda podjeli na dijelove (sprintove) tijekom kojih se programira određena funkcionalnost. Sprintovi su vremenski periodi koji traju od 1 do 4 tjedna. Oni zadaci čije je izvršavanje planirano za određeni sprint, a nisu dovršeni u tom istom sprintu, prenose se u sljedeći sprint te se nastoje izvršiti u sljedećem sprintu. Svaki sprint sastoji se od sastanka za planiranje sprinta, dnevnog Scruma, posla razvoja, revizije sprinta te retrospektive sprinta.

Temeljni dio Scrum metodologije su Product Backlog i Sprint Backlog. Product Backlog predstavlja listu svih korisničkih priča, tj. ideja koje moraju biti implementirane u skladu s korisnikovim potrebama. U početku Product Backlog sadrži samo one zahtjeve koji su inicijalno poznati i razumljivi, te se daljnje razvija sa razvojem proizvoda. Sadrži listu svih mogućnosti, funkcionalnosti, zahtjeva, unaprjeđenja i popravaka koji zajedno čine promjene koje će se izvršiti nad proizvodom u budućnosti. Osim implementiranih ideja, Product Backlog sadrži i neke odbačene ideje koje ne odgovaraju potrebama korisnika. Product Backlog se inače sortira prema vrijednosti, nužnosti i prioritetu, a član tima koji je odgovoran za Product Backlog je vlasnik proizvoda (engl. Product Owner). Sprint Backlog obuhvaća korisničke priče definirane u Product Backlogu koje su odabrane za određeni sprint. Korisničkim pričama koje su odabrane za određeni sprint dodjeljuju se zadaće kako bi se korisničke priče lakše ostvarile. Svakoj zadaći dodjeljuje se određena težina koja predstavlja težinu realizacije te zadaće. Sprint Backlog se koristi kako bi razvojni tim mogao napraviti procjenu koje će se funkcionalnosti implementirati u sljedećem vremenskom razdoblju i koliki će posao biti potreban za realizaciju tih funkcionalnosti.

## 5.1. Definiranje tima

Da bi razvoj aplikacije bio što lakši, potrebno ju je razvijati unutar tima. U Scrum metodologiji razvoja svakom članu tima mora biti pridružena određena uloga kako bi se podijelili zadaci i odgovornosti pojedinih članova tima. Uloge Scrum metodologije razvoja koje su podijeljene unutar tima su vlasnik proizvoda (engl. Product Owner), razvojni tim te Scrum master.

Vlasnik proizvoda je osoba koja je zatražila izradu aplikacije i kome će na kraju ona biti isporučena. Razvojni tim mora raditi prema uputama vlasnika proizvoda te svi moraju poštovati njegove odluke.

Scrum master je osoba koja je odgovorna da je Scrum razumljiv svima u timu, da se sve odvija prema planu i da se koristi pravilno.

Razvojni tim je skupina osoba koje rade konkretan posao te na kraju svakog sprinta isporučuju funkcionalni dio proizvoda.

Tim se sastoji od četiri člana, a podjela uloga u timu je sljedeća:

1. Scrum master – Anabel Li Kečkeš
2. Vlasnik proizvoda – I. osnovna škola Bjelovar (profesor informatike)
3. Razvojni tim:
   * Dario Horvat
   * Tea Jarčov
   * Anabel Li Kečkeš
   * Matija Popijač

## 5.2. Praćenje izrade projekta kroz Quick Scrum alat

Quick Scrum alat je besplatni scrum alat koji koristimo prilikom razvoja aplikacije i dokumentacije. U njemu su definirani svi zadaci koji su potrebni kako bi se projekt uspješno izvršio. U Product backlogu definirane su sljedeće korisničke priče :

**US#1:** Analiza osnovnih funkcionalnosti aplikacije i vremenski plan razvoja i specifikacija korisničkih, funkcionalnih i nefunkcionalnih zahtjeva

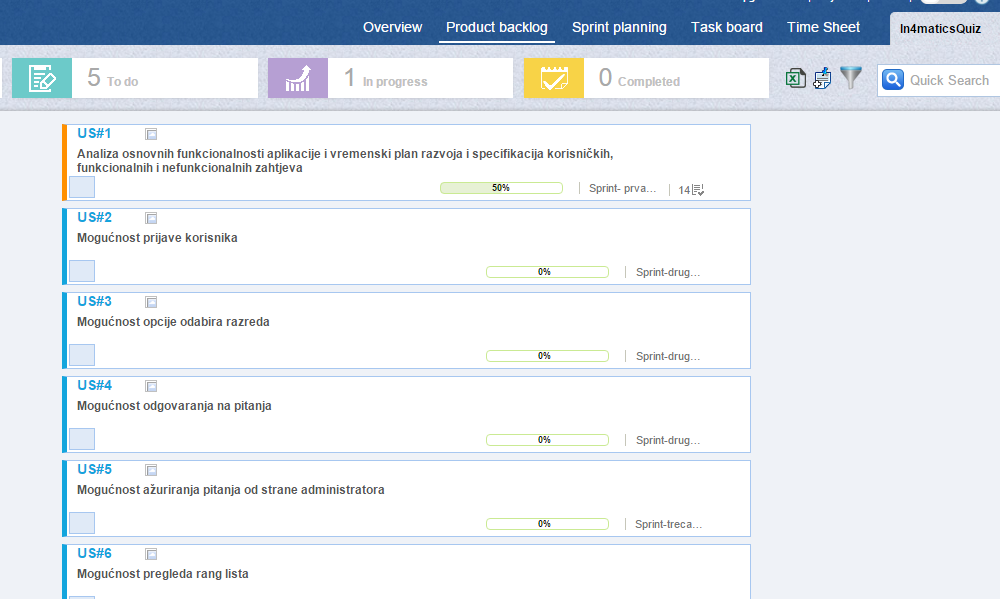
**US#2:** Mogućnost prijave korisnika

**US#3**: Mogućnost opcije odabira razreda

**US#4:** Mogućnost odgovaranja na pitanja

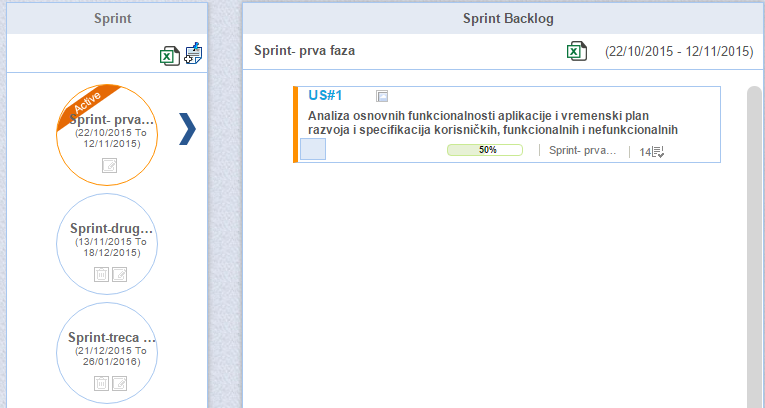
**US#5:** Mogućnost ažuriranja pitanja od strane administratora

**US#6:** Mogućnost pregleda rang lista

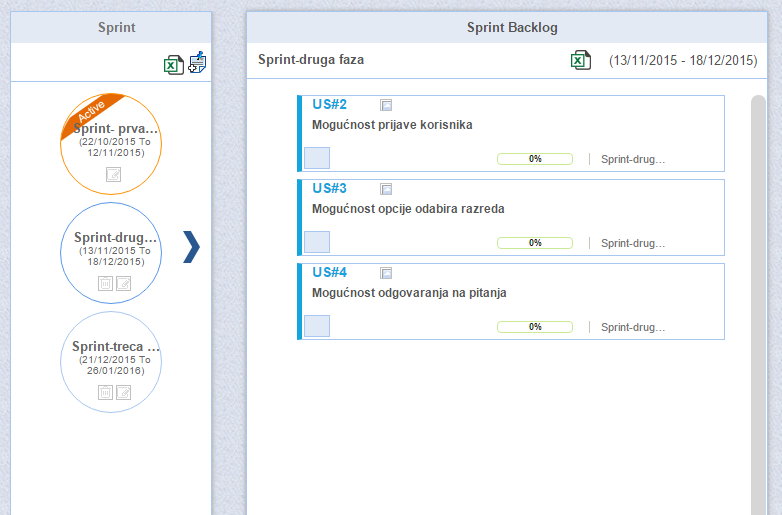


**Slika 5. Product backlog iz Quick Scrum-a**

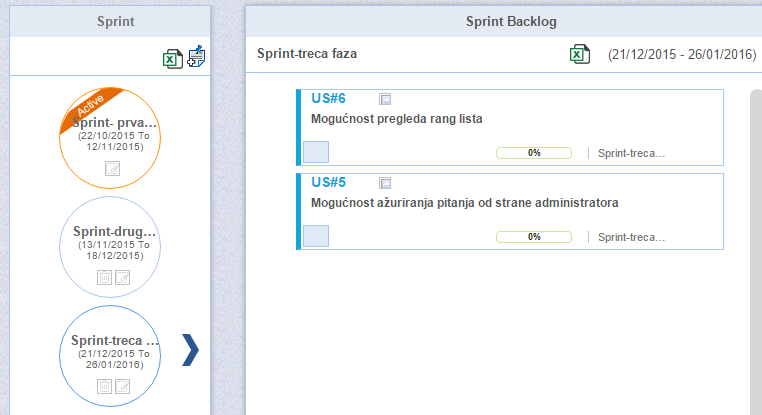
Sve zadatke koje moramo izvršiti razvrstali smo u tri sprinta. U svaki sprint dodali smo korisničke priče koje ćemo raditi u tom sprintu.



**Slika 6. Sprint Backlog iz Quick Scrum-a (prva faza)**

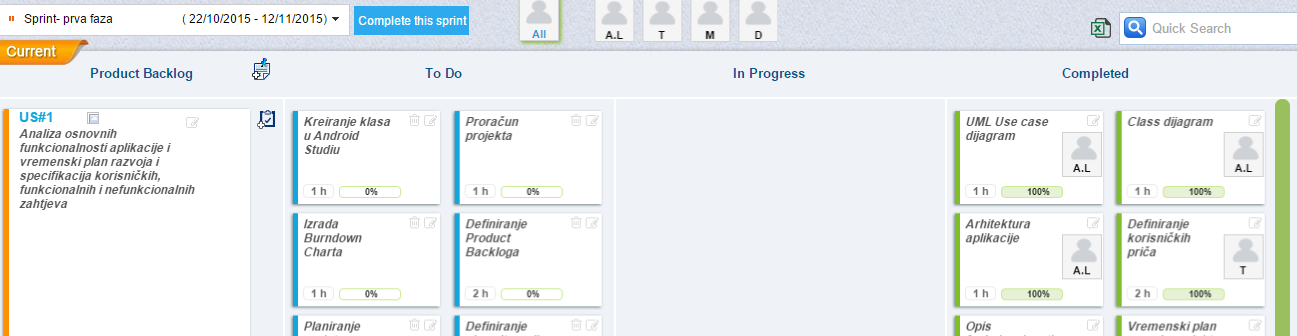


**Slika 7. Sprint Backlog iz Quick Scrum-a (druga faza)**

****

**Slika 8. Sprint Backlog iz Quick Scrum-a (treća faza)**

Svaka korisnička priča ima svoje zadatke koji su dodijeljeni pojedinim članovima tima.



**Slika 9. Task Board iz Quick Scrum-a**

### 5.2.1. Sprint 1 – detaljna specifikacija

U prvom sprintu napravljena je prva korisnička priča koja traje od 22/10/2015 do 12/11/2015:

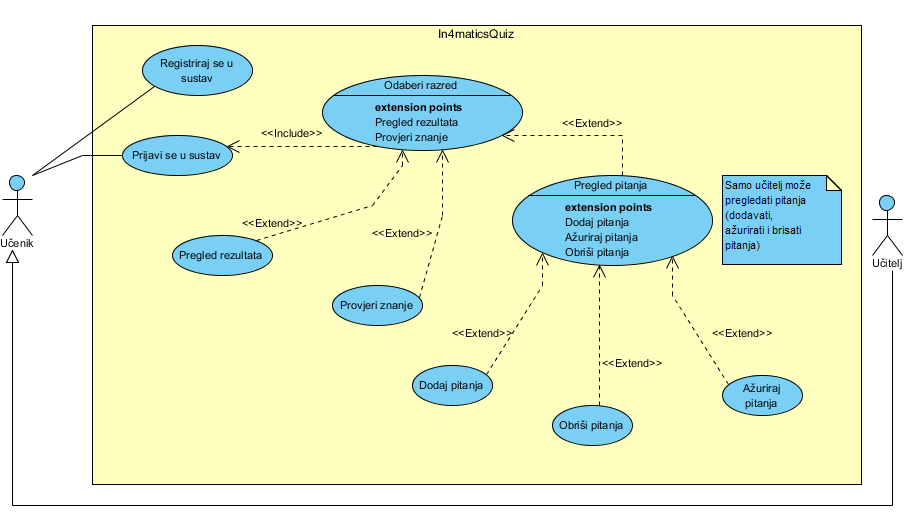
**US#1:** Analiza osnovnih funkcionalnosti aplikacije i vremenski plan razvoja i specifikacija korisničkih, funkcionalnih i nefunkcionalnih zahtjeva

**Tablica 2. Zadaci koji se nalaze u US#1**

|  |  |  |
| --- | --- | --- |
| Naziv zadatka | Trajanje zadatka | Zadatak obavio |
| Proračun projekta | 1 h | Anabel Li Kečkeš |
| Dijagram slučajeva korištenja (Use case dijagram) | 1 h | Anabel Li Kečkeš |
| Class dijagram | 1 h | Anabel Li Kečkeš |
| Arhitektura aplikacije | 1 h | Anabel Li Kečkeš |
| Planiranje sprintova  (trajanje, zadaci) | 5 h | Dario Horvat |
| Vremenski plan razvoja projekta | 3 h | Dario Horvat |
| Definiranje korisničkih priča | 2 h | Tea Jarčov |
| Opis funkcionalnosti aplikacije | 1 h | Tea Jarčov |
| Mockup – korisničko sučelje | 5 h | Tea Jarčov |
| ERA model | 2 h | Matija Popijač |
| Kreiranje klasa u Android Studiu | 2 h | Matija Popijač |

**Slika 10. BurnDown chart iz Quick Scrum-a**

# Dijagram slučajeva korištenja



**Slika 11. Dijagram slučajeva korištenja**

# Arhitektura sustava

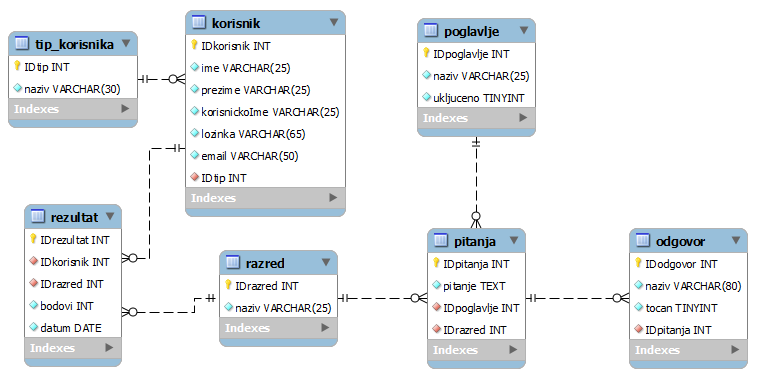


**Slika 12. Arhitektura sustava**

# Dijagram klasa

**Slika 13. Dijagram klasa**

# ERA model



**Slika 14. ERA model**

# Mockup – korisničko sučelje

# Webservis