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

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:

Dr.Sc. Zlatko Stapić

Varaždin, listopad 2015.

**Sadržaj**

[1. Uvod 1](#_Toc433313740)

[2. Korisnički zahtjevi 2](#_Toc433313741)

[2. Projektni plan 4](#_Toc433313742)

[2.1. Metodologija razvoja 4](#_Toc433313743)

[2.2. Definiranje tima 5](#_Toc433313744)

[2.3. Funkcionalnosti aplikacije 5](#_Toc433313745)

[2.4. Dijagram slučajeva korištenja 6](#_Toc433313746)

# 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 namjeravamo razvijati za Android platformu, s obzirom da su pametni telefoni android platforme jeftiniji te pristupačniji 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. metoda 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 metodi 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.

|  |  |  |
| --- | --- | --- |
| 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. |

# Projektni plan

## 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 2 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 metode 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.

## 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

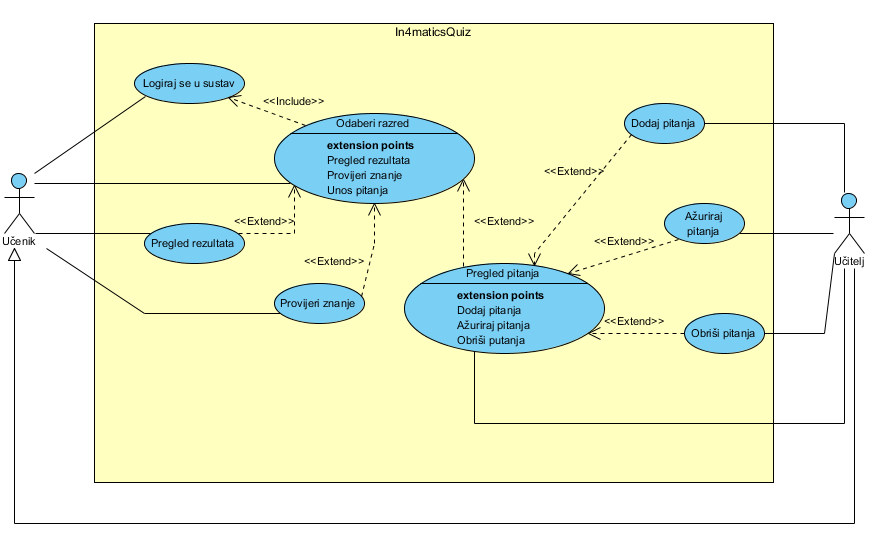
## Funkcionalnosti aplikacije

Osnovna funkcionalnost ove aplikacije je rješavanje zadataka. Nakon što učenik riješi zadatak bit 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

## 

## Dijagram slučajeva korištenja



## Arhitektura sustava

