Oblikovanje programske potpore

Ak. god. 2017./2018.

*GuildBuild*

Dokumentacija, Rev. *1*

Grupa: *Amigosi*

Voditelj: *Karlo Poljanec*

Datum predaje:

Nastavnik: *Nikolina Frid*

Sadržaj

[1. Dnevnik promjena dokumentacije 3](#_Toc431806045)

[2. Opis projektnog zadatka 4](#_Toc431806046)

[3. Pojmovnik 6](#_Toc431806047)

[4. Funkcionalni zahtjevi 7](#_Toc431806048)

[5. Ostali zahtjevi 8](#_Toc431806049)

[6. Arhitektura i dizajn sustava 9](#_Toc431806050)

[6.1. Svrha, opći prioriteti i skica sustava 10](#_Toc431806051)

[6.2. Dijagram razreda s opisom 11](#_Toc431806052)

[6.3. Dijagram objekata 12](#_Toc431806053)

[6.4. Ostali UML dijagrami 13](#_Toc431806054)

[7. Implementacija i korisničko sučelje 14](#_Toc431806055)

[7.1. Dijagram razmještaja 14](#_Toc431806056)

[7.2. Korištene tehnologije i alati 15](#_Toc431806057)

[7.3. Isječak programskog koda vezan za temeljnu funkcionalnost sustava 16](#_Toc431806058)

[7.4. Ispitivanje programskog rješenja 17](#_Toc431806059)

[7.5. Upute za instalaciju 18](#_Toc431806060)

[7.6. Korisničke upute 19](#_Toc431806061)

[8. Zaključak i budući rad 20](#_Toc431806062)

[9. Popis literature 21](#_Toc431806063)

[Dodatak A: Indeks (slika, dijagrama, tablica, ispisa kôda) 22](#_Toc431806064)

[Dodatak B: Dnevnik sastajanja 23](#_Toc431806065)

[Dodatak C: Prikaz aktivnosti grupe 24](#_Toc431806066)

[Dodatak D: Plan rada / Pregled rada i stanje ostvarenja 26](#_Toc431806067)

# Dnevnik promjena dokumentacije

|  |  |  |  |
| --- | --- | --- | --- |
| **Rev.** | **Opis promjene/dodatka** | **Autor(i)** | **Datum** |
| 0.1 | Napravljen predložak i opis projekta | Poljanec | 17.10.2017 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Opis projektnog zadatka

Cilj projekta je razviti mobilnu aplikaciju GuildBuild koja će omogućavati zajednicama igrača MMO igara tzv. cehovima (eng. guild) bolju komunikaciju i organizaciju svoje zajednice kroz praćenje događaja, napretka, stečenih vještina koje posjeduju određeni članovi i sl.

Članovi svakog ceha organizirani su u slijedećoj hijerarhiji:

* Članovi
* Koordinatori
* Vođa ceha

Postoji samo jedan vođa ceha, a on može vodstvo prepustiti nekom drugom članu, npr. ako privremeno ili trajno napušta ceh. Broj članova i koordinatora je neograničen. Svaki član može biti unaprijeđen u koordinatora ili unazađen ovisno o zaslugama od bilo kojeg koordinatora ili vođe ceha. Vođu ceha mogu izabrati koordinatori većinskim glasovima. Inicijalni vođa ceha je onaj član koji je napravio ceh. Ukoliko vođa ceha nenadano obriše svoj profil, a nije prepustio vodstvo nekom drugom članu, svi koordinatori se kod prijave preusmjeravaju na glasanje o novom vođi te sve ostale funkcionalnosti (osim uređenja vlastitog profila) postaju nedostupne. To se ponavlja sve dok se ne izabere novi vođa. Pretpostavlja se da će uvijek biti koordinatora koji će izabrati novog vođu.

Aplikaciju mogu koristiti samo registrirani korisnici koji kod registracije moraju dati e-mail adresu i odgovarajući nadimak. Jedino drugi registrirani korisnici mogu pregledavati profile dok oni javno nisu dostupni. Registrirani korisnici mogu stvoriti svoj ceh ili se pridružiti nekom od postojećih. Ceh se stvara tako što se korisnik najprije registrira u aplikaciji, a zatim odabire ime ceha i MMO igru za koju ga stvara. Korisnik koji se želi pridružiti cehu popunjava obrazac koji je dostupan na javnoj stranici ceha u kojem kratko navodi zašto se želi pridružiti, a može biti prihvaćen od strane vođa ceha ili nekog od koordinatora. Na profilu korisnika je vidljivo kojem cehu pripada (ako pripada). Korisnik može biti registriran na više MMO igara, ali u zasebnoj igri može pripadati samo jednom cehu.

Članovima ceha je unutar aplikacije dostupan kalendar događaja u igri za koji se svaki član može prijaviti. Događaje u kalendaru stvaraju koordinatori ili vođa ceha, dok ostali članovi mogu samo označiti hoće li sudjelovati.

Aplikacija također mora omogućavati praćenje napredovanja ceha tako što vođa ceha navede ciljeve koje treba ispuniti, a koordinatori upisuju jesu li te ciljeve ispunili te navode koji su sve članovi u tome sudjelovali. Svaki cilj može imati podciljeve pa tako npr. za glavni cilj se može postaviti dovršetak neke operacije (eng. raid), a za podciljeve savladavanje nekog od protivnika u operaciji (eng. boss). Vođa ceha može odrediti da se neki od zadanih ciljeva (i stupanj ispunjenosti) vide na javnoj stranici ceha. Na profilu svakog člana je vidljiv popis svih ciljeva u čijem je ispunjenu sudjelovao.

Svaki član na svom profilu navodi nazive svih likova koje ima u igri te njihovu klasu i razinu (eng. level). Također navodi i sve vještine (eng. crafting skills) koje taj lik posjeduje i njihovu razinu. Dodatno svaki član može još na svom profilu napisati nešto o sebi te postaviti svoj avatar. Članovi si međusobno mogu slati poruke unutar aplikacije, a vođa ceha i vođe timova imaju mogućnost postavljanja obavijesti na stranicu ceha koja je vidljiva svim članovima. Poruke koje je neki član dobio i/ili poslao su vidljive isključivo njemu (nisu vidljive ostalim članovima).

U sustavu postoje i administratori koji mogu dodati ili obrisati korisnike te obrisati postojeći ceh. Brisanjem ceha svim članovima se sa profila uklanjaju svi podaci vezani za taj ceh. Administratori unose nazive igara za koje je moguće stvoriti ceh te svakoj igri pridružuju nazive klasa kojima mogu pripadati likovi članova ceha.

# Pojmovnik

**MMO -** skraćenica od MMORPG

**MMORPG –** engl. Massively Multiplayer Online Role-Playing Game, online igra sa puno igrača u kojem svaki igrač ima svoju ulogu

# Funkcionalni zahtjevi

*Funkcionalni zahtjevi su određeni samom specifikacijom sustava.*

*Nekoliko točaka je potrebno popuniti u ovom dijelu dokumentacije:*

* *Navesti koje su dionici koji imaju interes u ovom sustavu ili su nositelji odgovornosti. To su prije svega korisnici, ali i administratori sustava, sponzori, manageri.*
* *Navesti aktore koji izravno koriste ili komuniciraju sa sustavom. Oni mogu imati inicijatorsku ulogu, tj. započinju određene procese u sustavu ili samo sudioničku ulogu, tj. obavljaju određeni posao. Za svakoga od njih treba navesti ulogu i svrhu njihovog djelovanja u sustavu.*
* *Napraviti analizu obrazaca uporabe (engl.* Use Cases*):*
  + *Za sve obrasce koji su u planu za implementaciju napraviti detaljne opise (prema donjem predlošku).*
  + *Nacrtati dijagrame obrazaca uporabe (engl.* Use Case Diagrams*).*
* *Nacrtati sekvencijske dijagrame za sve obrasce uporabe sustava. Uz svaki dijagram dodati odgovarajući opis.*

# Ostali zahtjevi

*Ostali ili nefunkcionalni zahtjevi nisu izravno vezani za primarnu funkciju sustava. Govore o određenim mjerama kvalitete i svojstvima i ograničenjima platforme i procesa ostvarenja sustava.*

*Pod mjerama i zahtjevima na kvalitetu sustava misli se na nekoliko aspekata: neosjetljivost na pogreške, (is)koristivost, pouzdanost, performanse, (p)održivost.*

*Pod ograničenjima se uglavnom misli na svojstva tehnologije i metodologiju implementacije: ograničenja tehnologije, vrijeme izrade, (ne)usvojeni standardi, informacije o procesu razvoja i oblikovanja.*

*Svaki takav zahtjev potrebno je navesti u jednoj ili dvije rečenice.*

# Arhitektura i dizajn sustava

*Ovo je je jedno od bitnijih poglavlja jer se opisuju temelji implementacije sustava. Moguće je razlikovati više aspekata arhitekture i oblikovanja sustava od kojih je većinu potrebno dokumentirati već prema zahtjevima projektnog zadatka. Neki od tih su: stil arhitekture i identifikacija podsustava, preslikavanje na radnu platformu, spremišta podataka, mrežni protokoli, globalni upravljački tok, sklopovsko-programski zahtjevi.*

*U svakoj stavci navedenoj u nastavku potrebno je dati tekstovni opis. te ilustrirati odgovarajućim UML dijagramima. Projektna grupa odlučuje gdje i koje sve dijagrame treba prikazati u pojedinim stavkama, ali cijela točka 6. Arhitektura i dizajn sustava, mora sadržavati barem po jedan:*

*Dijagram razreda*

*Dijagram objekata*

*Dijagram stanja*

*Dijagram aktivnosti*

*Komunikacijski dijagram*

*Dijagram komponenti*

*Ostali UML dijagrami (npr. Sekvencijski za objekte) mogu se koristiti po potrebi.*

*Za crtanje UML dijagrama preporuka je koristiti alat Astah Community.*

*Preporuča se da ovo poglavlje podijelite po sljedećim potpoglavljima:*

*6.1. Svrha, opći prioriteti i skica sustava*

*6.2. Dijagram razreda s opisom*

*6.3. Objektni dijagram.*

*6.4. Ostali UML dijagrami*

## Svrha, opći prioriteti i skica sustava

*Potpoglavlje treba sadržavati sljedeće:*

*Koji sustav ili dio sustava ovaj dio dokumentacije opisuje, izbor arhitekture temeljem principa oblikovanja pokazanih na predavanjima (objasniti zašto ste baš odabrali takvu arhitekturu).*

*Organizacija sustava s najviše razine apstrakcije (npr. klijent-poslužitelj, baza podataka, datotečni sustav, grafičko sučelje (ako nije do kraja poznato, dopuniti u 2. ciklusu)).*

## Dijagram razreda s opisom

*Potpoglavlje treba sadržavati dijagram razreda, jedan ili više njih, s opisima šta koji razred znači. Ono što je potrebno imati u dokumentaciji prilikom prve predaje je: svi razredi specifični za vaš projekt, nazivi metoda i vrste pristupa metodama (npr. javni, zaštićeni), nazive atributa razreda, veze i odnosi između razreda, okviri (forme) grafičkog sučelja specifične za pojedini programski jezik, ako su vam poznate.*

*Primjer onoga što nije potrebno imati u dokumentaciji prilikom druge predaje su: tipovi parametara metoda razreda (npr. boolean, int), događaji (npr. klikovi miša).*

*Ako za neke stvari niste sigurni kako će biti implementirane, tada je bolje taj dio izostaviti prilikom prve predaje pa dopuniti na drugoj predaji. Prilikom druge predaje navest ćete sve one razrede sa svim detaljima koje direktno koristite u implementaciji sustava.*

## Dijagram objekata

*Prikazati stanje sustava u određenom trenutku. Dio je Revizije 1.*

## Ostali UML dijagrami

*Ovdje počinju sadržaji Revizije 2.*

*U ovom potpoglavlju potrebno je uvrstiti dodatna četiri dijagrama koji prikazuju arhitekturu sustava i to: komunikacijski dijagram, dijagram stanja, dijagam aktivnosti i dijagram komponenti. Potrebno je dati barem po jedan primjerak svakog od dijagrama. Potrebno je da svaki od dijagrama prikazuje (jedan) bitan dio funkcionalnosti sustava. Dijagram komponenti treba prikazivati sve komponente sustava. Prema potrebi, ovdje se mogu dodati i neki dodatni sekvencijski dijagrami za objekte.*

# Implementacija i korisničko sučelje

*U ovom poglavlju potrebno je:*

*dati dijagram razmještaja (deployment dijagram)*

*navesti koje su tehnologije i alati korišteni u razvoju sustava*

*dati isječak programskog koda koji implementira neku od temeljnih funkcionalnosti u sustavu*

*objasniti kako je ispitano implementirano rješenje i pokazati bar 4 ispitna scenarija*

*dati upute za instalaciju*

*dati upute za korištenje*

## Dijagram razmještaja

*Potrebno je umetnuti dijagram razmještaja i po potrebi ga opisati.*

## Korištene tehnologije i alati

*Navesti sve tehnologije i alate koji su primijenjeni pri izradi projekta te ih ukratko opisati; njihovo značenje i mjesto i način primjene.*

## Isječak programskog koda vezan za temeljnu funkcionalnost sustava

*U ovom poglavlju potrebno je prikazati isječak programa koji prema mišljenju studenta ostvaruje temeljnu funkcionalnost u sustavu (ili nekom modulu).*

## Ispitivanje programskog rješenja

*U ovom poglavlju je potrebno opisati provedbu ispitivanja implementiranih funkcionalnosti s prikazom odabira ispitnih slučajeva.*

*Prilikom prezentacije svojih Ispitnih scenarija (minimalno četiri) studenti trebaju ispitati temeljnu funkcionalnost i rubne uvjete. Poželjno je da se napravi i ispitni slučaj koji koristi funkcionalnosti koje nisu implementirane te da se vidi na koji način sustav reagira kada nešto nije u potpunosti ostvareno.*

*Različiti ulazi za ispitne scenarije trebaju pokrivati temeljnu funkcionalnost nekog modula i nekoliko rubnih uvjeta.*

## Upute za instalaciju

*U ovom poglavlju potrebno je dati upute za instalaciju ostvarenog prototipa.*

## Korisničke upute

*Korisničke upute ovisit će o količini implementirane funkcionalnosti. Očekuje se da upute budu na oko pet A4 stranica koje će dati potpuni opis funkcionalnosti sustava sa stajališta krajnjeg korisnika.*

# Zaključak i budući rad

*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.*

# Popis literature

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

1. Oblikovanje programske potpore, FER ZEMRIS, <http://www.fer.hr/predmet/opp>
2. Oblikovanje programske potpore, FER ZEMRIS, <http://www.zemris.fer.hr/predmeti/opp>

# Dodatak A: Indeks (slika, dijagrama, tablica, ispisa kôda)

*U ovom dijelu dodatka potrebno je ispisati (numerirane) liste slika, dijagrama, tablica, ispisa kôda i ostalih pomoćnih sadržaja iz teksta dokumentacije.*

# Dodatak B: Dnevnik sastajanja

*U ovom dijelu dodatka potrebno je prenijeti sadržaj dnevnika sastajanja, tj. datoteke 'dnevnik.txt' iz repozitorija grupe. Datoteku 'dnevnik.txt' potrebno je voditi kroz repozitorij, a ovdje će to biti preneseno kao dio konačne dokumentacije.*

# Dodatak C: Prikaz aktivnosti grupe

*U ovom dijelu dodatka potrebno je navesti*

* *popis članova grupe i njihovih zaduženja, tj. odrađenih poslova (u postocima ukupno odrađenog posla)*
* *prenijeti dijagram pregleda promjena nad datotekama projekta. Potrebno je u* BitBucket *sučelju instalirati dodatak* AwesomeGraph *koji daje prikaz grafički prikaz aktivnosti grupe, tj. '*commit' *akcija tijekom trajanja projekta, te na kraju projekta generirane grafove prenijeti u ovo poglavlje dokumentacije.*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Popis aktivnosti** | **Članovi grupe** (abecednim redom) | | | | | |
| Karlo Poljanec | Filip Kerman | Matija Krivošić | Kruno Lapat | Ivan Premužić | Ivan Varga |
| **Upravljanje projektom** |  |  |  |  |  |  |
| **Opis projektnog zadatka** |  |  |  |  |  |  |
| **Rječnik pojmova** |  |  |  |  |  |  |
| **Opis funkcionalnih zahtjeva** |  |  |  |  |  |  |
| **Opis ostalih zahtjeva** |  |  |  |  |  |  |
| **Arhitektura i dizajn sustava** |  | | | | | |
| Svrha, opći prioriteti i skica sustava |  |  |  |  |  |  |
| Dijagram razreda s opisom |  | | | | | |
| Dijagram objekata |  |  |  |  |  |  |
| Ostali UML dijagrami |  |  |  |  |  |  |
| **Implementacija i korisničko sučelje** |  | | | | | |
| Dijagram razmještaja |  |  |  |  |  |  |
| Korištene tehnologije i alati |  |  |  |  |  |  |
| Isječak programskog kôda |  |  |  |  |  |  |
| Ispitivanje programskog rješenja |  |  |  |  |  |  |
| Upute za instalaciju |  |  |  |  |  |  |
| Korisničke upute |  |  |  |  |  |  |
| **Plan rada** |  |  |  |  |  |  |
| **Pregled rada i stanje ostvarenja** |  |  |  |  |  |  |
| **Zaključak i budući rad** |  |  |  |  |  |  |
| **Popis literature** |  |  |  |  |  |  |
| **Dodaci** |  | | | | | |
| Indeks |  |  |  |  |  |  |
| Dnevnik sastajanja |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

*Napomena: Doprinose u aktivnostima treba navesti u postocima po članovima grupe. Zbroj postotaka u svakom retku treba biti 100%.*

Pregled pohrana kroz vrijeme trajanja projekta (primjer):



# Dodatak D: Plan rada / Pregled rada i stanje ostvarenja

*U ovom poglavlju potrebno je navesti:*

* *(u rev. 1) koji je plan rada za rev. 2,*
* *(u rev. 2) koji je status implementacije u odnosu na postavljene ciljeve, procjenu vremena dovršetka projekta (ako zadatak nije u potpunosti ispunjen), koje bi bile smjernice za daljnji rad kad bi se nastavilo s projektom te u čemu bi se sastojale buduće nadogradnje.*