**TEHNIČKO VELEUČILIŠTE U ZAGREBU**

**STRUČNI STUDIJ INFORMATIKE**



**Seminarski rad - Objektno orijentirano programiranje II**

**Multiplayer RPG igrica temeljena na textualnom prikazu** (Napisan u C++ jeziku i u C++ Builderu)

Rad u paru: Filip Vasiljević i Leon Paolović

**Zagreb, svibanj 2020.**

Prijavnica – tema seminara iz

Objektno orijentiranog programiranja II

|  |  |
| --- | --- |
| **Ime i prezime:** | *Filip Vasiljević i Leon Paolović* |
| **JMBAG:** | *0035217569 i 0246082510* |
| **Naziv teme/aplikacije:** | *Multiplayer RPG igrica temeljena na textualnom prikazu* |
| **Opis teme/aplikacije:** | *Trenutno bezimena igrica bi bila izvedena kao VCL aplikacija napisana C++ builderu unutar Embarcaderovog RAD (Rapid Application Development) alata C++ builder.*  *Igra bi pratila linearnu 'Dungeon Crawl' strukturu sa on screen displayom u kojoj se prostoriji na mapi igrači nalaze. Save game bi se mogao izvesti kao pohrana XML dokumenta. Sama čudovišta i igrače se mogu izvesti uporabom polimorfizma (isto vrijedi za predmete unutar irge), a igrači bi kontrolirali svoje likove sa terminalom (poput CMD-a, ali s našim homebrew komandana za igru). Igrači bi također kroz taj isti terminal mogli međusobno komunicirati i unos komandi bi se odvijao na intervalu reda (Turn based system). Zajedno igranje bi se izvelo preko TCP protokola u klijent server modelu. Sama pravila igre bi pratila pojednostavljenu verziju 5e 'Dungeons and Dragons' sustava.* |
| **Pokrivene laboratorijske teme:** | *Multiplayer RPG text based igra pokriva sljedeće laboratorijske teme:*  *1.* ***C++builder i klase*** *sa polimorfizmom i nasljeđivanjem (Igrac, Neprijatelji, Oklop, Pozicija u prostoru – Grid sustav). Klase i metode ćemo koristiti u našoj aplikaciji.*  *2.* ***XML*** *– Pohrana stanja igre (singleplayer ili mutliplayer, život i energija, pohrana predmeta i oklopa u invetory-u), Učitavanje save state-a za nastavak igre, Učitavanje i ispis opisa prostora, predmeta i likova (igrica bazirana na tekstu), Chat log pohrana i naknadno učitavanje*  *4.* ***Jednostavno grafičko sučelje***  *Minimalno 3 dijaloga (forme), lokalizacija na engleski (uz hrvatski kao glavni)* ***Primjer forma:***  *(Singleplayer / Multiplayer) – (Glavna forma igre) – (Inventory) – (Stats) – (New Game – Load Save) – (Character creation)*  *6.* ***DLL biblioteke*** *– Funkcije za svaku pojedinačnu 'tekst komandu' koju korisnik može unjeti za svog lika, npr. Napad, Opis, Pokupi, Iskoristi, Pričaj itd. Aplikacija će pozivati funkcije iz DLL datoteke.*  *8.* ***TCP/IP klijent-server***  *Zajednička igra između dvije instance igre (multiplayer), host računalo igraču 2 dijeli stanje 'svijeta', drugom igraču daje do znanja kad je njegov potez za unjet komandu i chat opcija između korisnika. Login/registracija sa zaštitom lozinke pomoću hash funckija.* |

Contents

[1. Uvod 4](#_Toc41853976)

[2. Arhitektura sustava 5](#_Toc41853977)

[3. Redoslijed pozivanja formi i srodne funkcionalnosti: 15](#_Toc41853978)

[3.1. Login – Glavni izbornik 15](#_Toc41853979)

[3.2. Character creation – Kreacija lika 16](#_Toc41853980)

[3.3. Text Dungeon – mainForm 17](#_Toc41853981)

[3.4. Side menu – Sporedni izbornik 21](#_Toc41853982)

[3.4.1. Stats and Items – Podaci i Oprema 21](#_Toc41853983)

[3.4.2. Connect – Spoji me 22](#_Toc41853984)

[3.4.3. Save Game i Load Game 24](#_Toc41853985)

[3.4.4. Quit – Izlaz 25](#_Toc41853986)

[4. Zaključak 26](#_Toc41853987)

[5. Literature 27](#_Toc41853988)

# **Uvod**

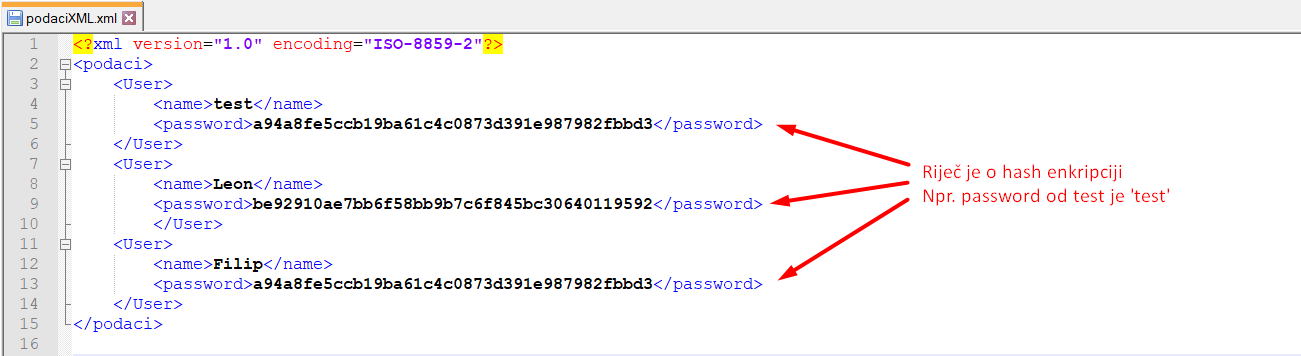
Multiplayer RPG igrica temeljena na tekstualnom prikazu, koju smo naknadno tokom developmenta imenovali „Text Dungeon“ je simulacija igre i služi kao dokaz koncepta da se takvo što može izvesti kao VCL Forms aplikacija. Igra sama po sebi puno prije demonstrira mogučnosti realizacije takvog projekta nego predstavlja pravu igru. Da smo nastavili developati ovo bi lagano postala prava igrica, u trenutnom stanju samo simboličo iskorištava sve funkcionalnost s kojim bi se prava igrica izradila, a one su:

* Kretanje po „grid“ sutavu od 20 polja u prostoru na karti pomoću strelica na tipkovnici
* Multiplayer movement izveden preko TCP/IP server-klijent veze
* Chat sustav dopisivanja izveden preko TCP/IP server-klijent veze
* Combat sustav izveden u obliku sustava poteza tokom kojeg se mogu uporabiti naše vlastorucno izrađene tekstualne naredbe
* Tekstualne naredbe koje korisnik može pozivati tokom sustava Combat-a, ali izvana, svaka pojedinacna naredba se sortira kroz DLL functionSort funkciju
* Ispis rezultata tih naredbi, ali i opisa soba prilikom kretanja kroz spomenuti grid sustav, se prikazuje u TListView objektu kojeg smo imenovali Display
* Character creation i Login pri poćetku rada aplikacije uporabljuju prosljeđivanje podataka između formi (extern tag), ali i hash inkripciju šifre
* Podaci o kreiranom liku, Login informacije i Opisi soba (oboje na hrv i eng) su pohranjene kao XML datoteke koje se pozivaju po potrebi
* Save Game i Load Game je realizirano kroz čitanje i kreaciju XML dokumenta
* Custom klase Entity i Item koji imaju klase potomke koje se s polimorfizmom pozivaju i koriste na mnoštvu mjesta
* Lokalizacija na 2 jezika
* 6 formi koje međusobno komuniciraju i dijele varijable po potrebi

# Arhitektura sustava

1 Flowchart aplikacije

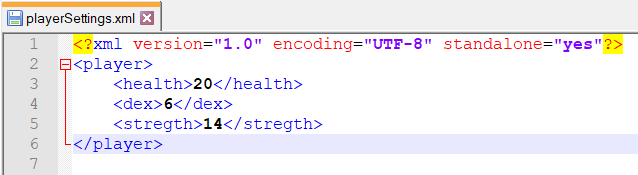
Sustav prilikom svakog paljenja traži korisnički login, informacije se čitaju (ili unose ako je riječ o registraciji novog korisnika) u XML datoteku



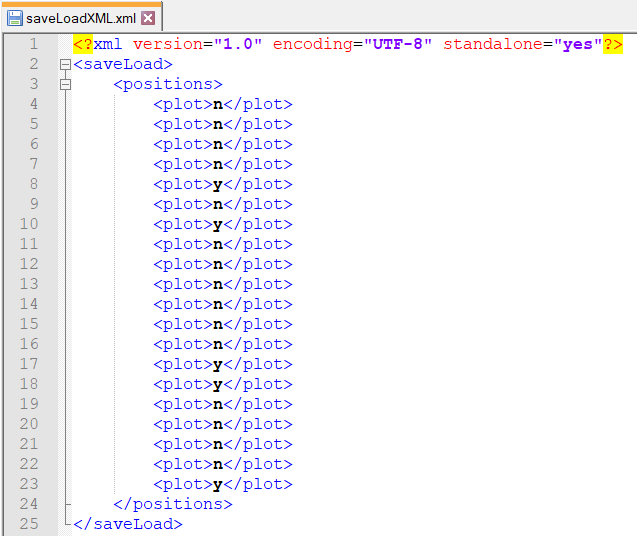
2 XML login podataka

Cijela aplikacija radi na principu da dijeli bitne varijable međusobno putem externalnog deklariranja varijabli i pozivanje srodnih .h datoteka unutar header prostora. Podaci korisnika, podaci likova, stanje igre i pohranjene poruke se spremaju u XML datoteke i iz njih čitaju prilikom naknadnog paljenja aplikacije.

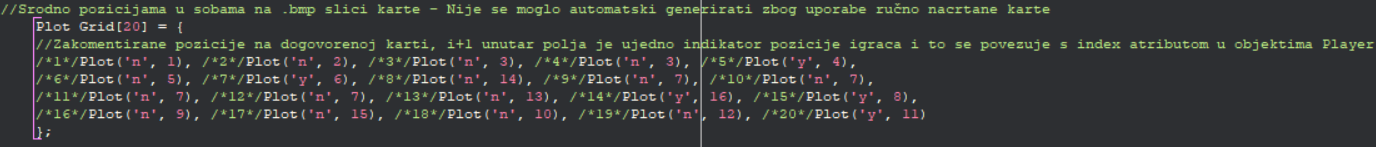
Podaci o stvorenom liku:



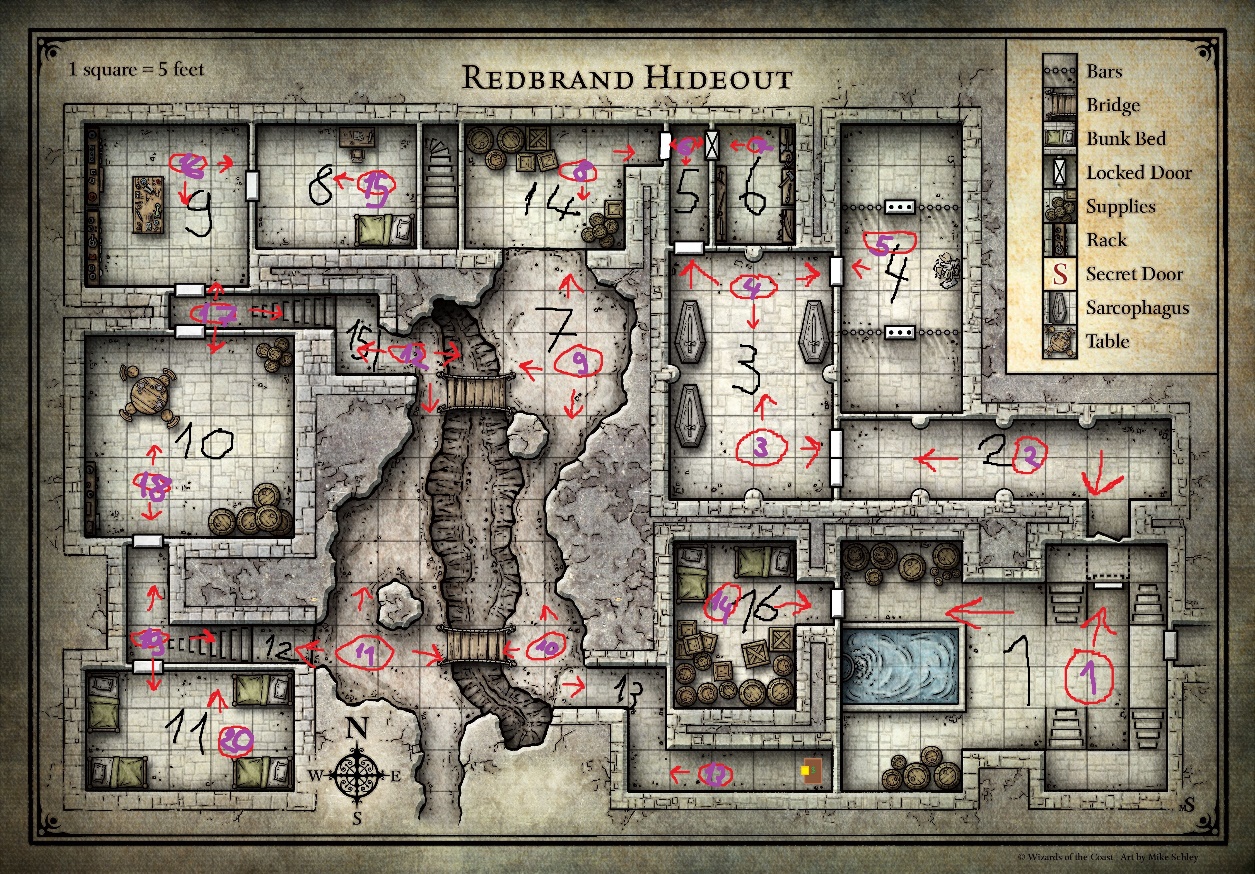
3 XML kreiranog lika

Podaci o stanju svijeta: 

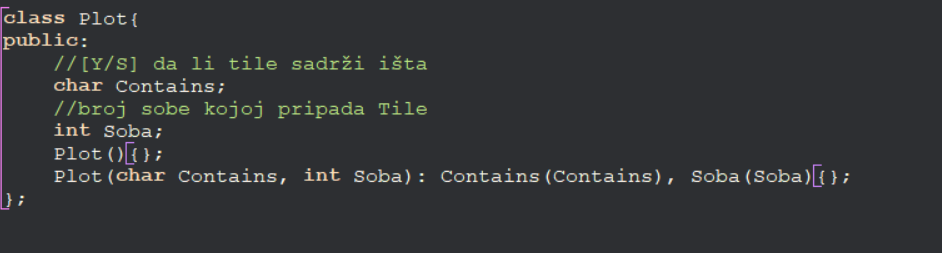
4 XML stanja u svijetu

Svaki plot u XML-u je srodan jednoj poziciji u plot gridu:

5 Grid Plot

Grid[20] je jednodimenzionalno polje čije svaka pozicija reprezentira jedno mjesto na kojem igraći mogu stajati na dogovorenoj karti:

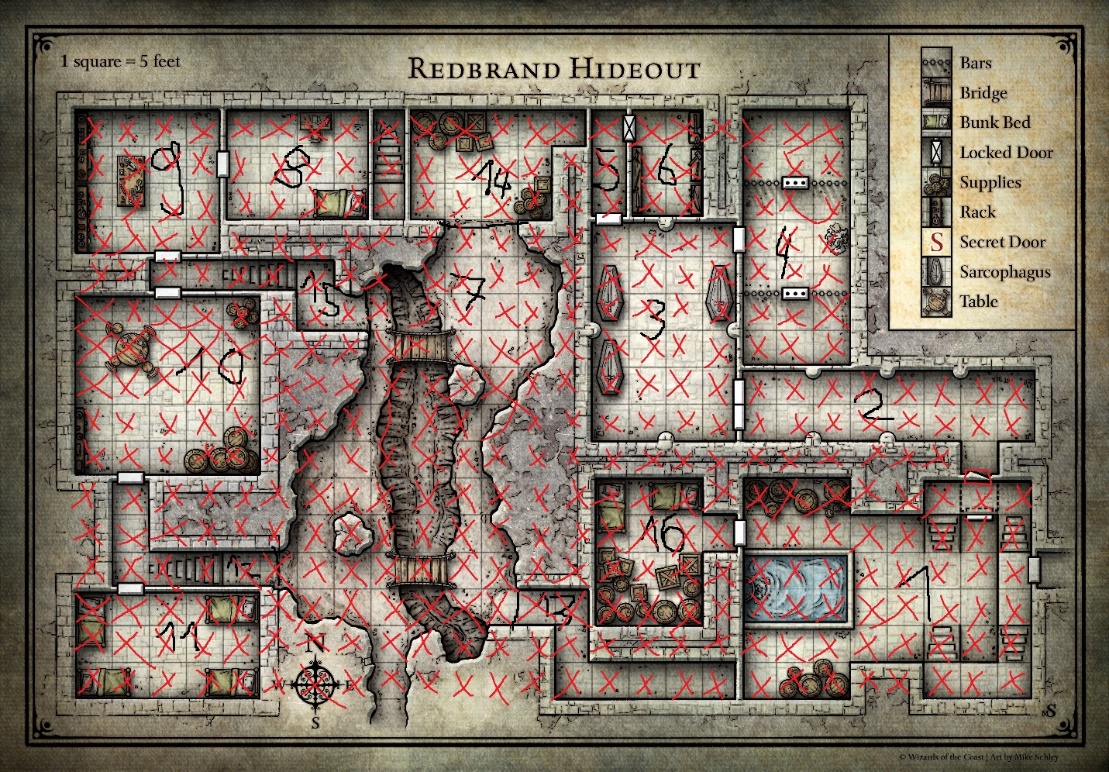
6 Dogovoreni layout pozicija

Dok je Plot naša klasa koja sadrži informacije o tome što se nalazi na svakoj pojedinačnoj poziciji:

7 Klasa Plot

Contains [Y/N] Ovisno o ovome se poziva Combat funkcija ako se na njemu nalazi neprijatelj  
Soba – Indikator u kojoj se sobi nalazi pozicija, neke sobe imaju više pozicija

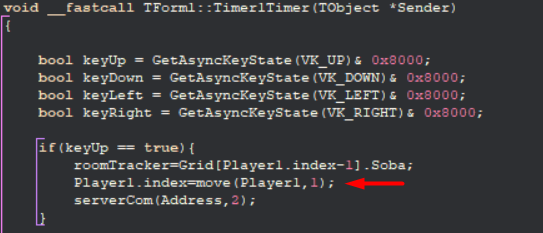
Ovaj pristup je zapravo downgrade od orginalno zamišljenog pristupa, prvih 10ak dana smo pokušavali osposobiti da je svaki kvadratić zasebna pozicija:



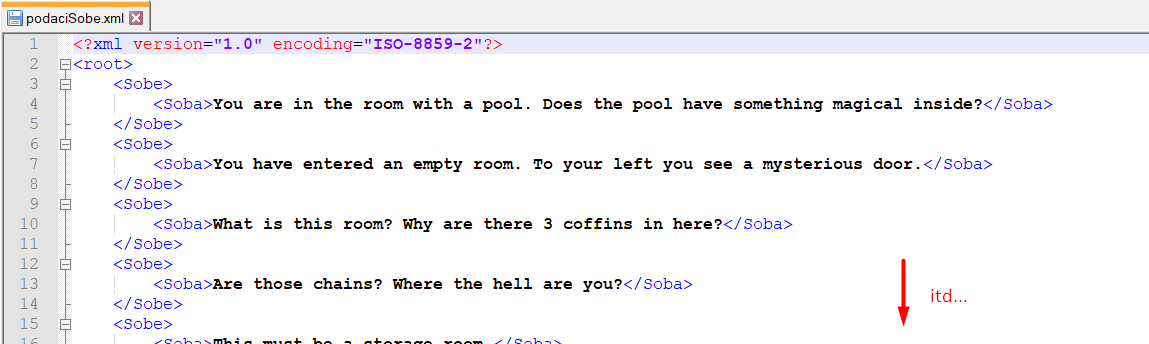
8 Stari Dogovoreni Layout Pozicija

Orginalni Grid je bio dvodimenzionalno polje i čak smo uspjeli sve ispravno iskodirat da se takvo kretanje omogući, ali smo od toga odustali kada smo vidjeli kolike daljnje implikacije na ostatak koda takva odluka nosi. Da je ovo punopravna igra išli bi s time pristupom, ali u svrhu prezentacije koncepta smo se odlučili za pristup sa jednodimenzionalnim poljem (oba pristupa su u skladu sa prijavnicom).

Samo kretanje se izvodi prilikom pritiska strelica na tipkovnici, pozivanjem naše vlastite move funkcije i komunikacijom sa tcp serverom da javi našem partneru da se naš srodni PlayerDot pomaknuo:

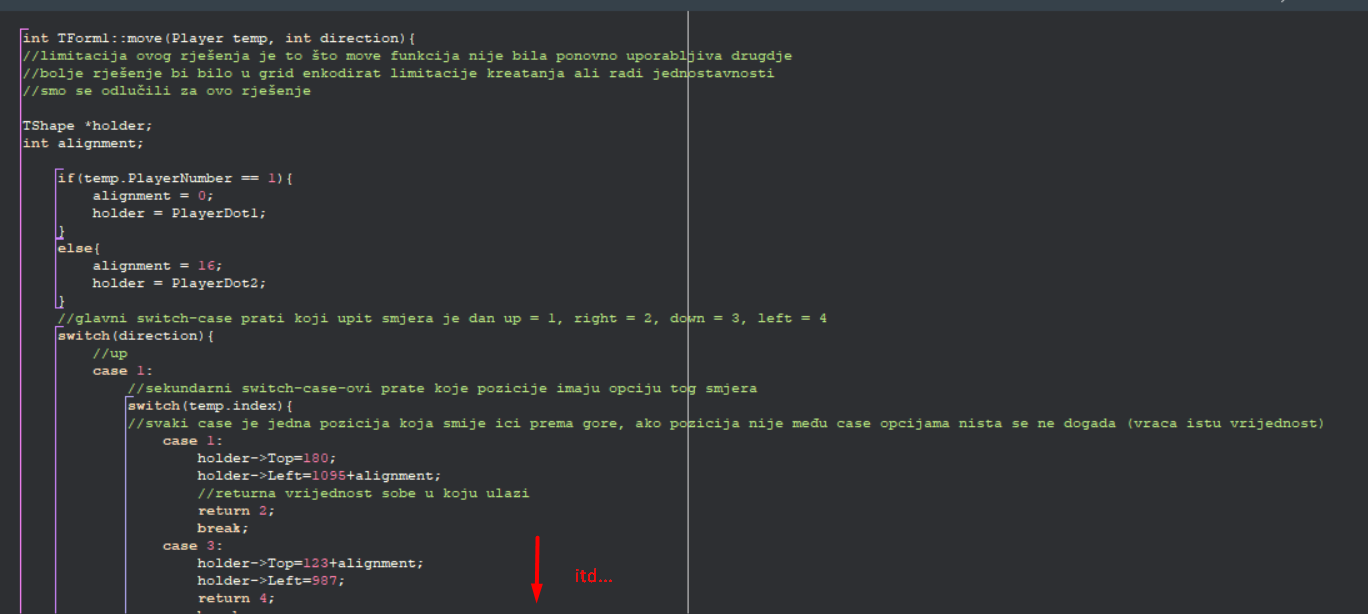


9 Dohvačanje strelica na tipkovnici - Posebne zahvale profesoru Kovačeviću na pomoći s ovime

U ovom segmentu se odvija mnogo stvari. **Prvo**, unutar Timer objekta aplikacija gleda je li korisnik pritisnuo jedno od 4 strelica smjera na tipkovnici. **Drugo**, roomTracker varijabla gleda je li korisnik svojim kretanjem promijenio sobu kojoj se nalazi i ako je poziva iz sljedećeg XML-a srodni opis sobe (u slučaju lokalizacije na hrv povlaci iz drugog XML-a): 

10 XML podatak opisa soba

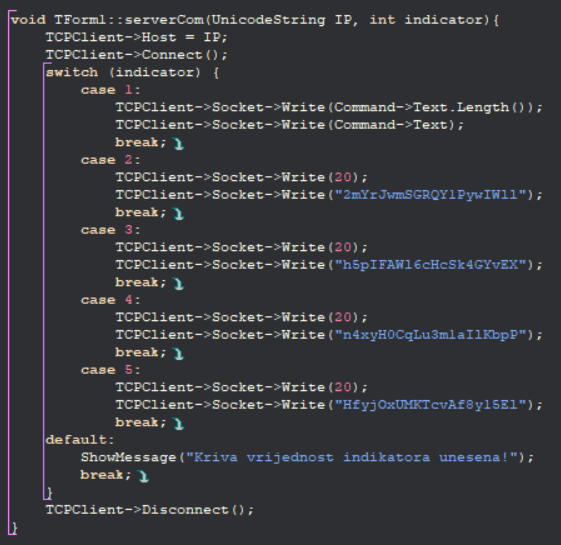
**Treće**, Player1.index = move(Player1,1); Poziva se move funkcija koja vraća poziciju novoga indexa i istovremeno na karti pomiće poziciju točke koja reprezentira igrača.



11 Move funkcija

Move funkcija prima objekt player i ovisno jel player 1 ili 2 seta pokazivać Tshape \*holder kao PlayerDot1 ili PlayerDot2 (s time smo reducirali kod za duplo) i tom pokazivaću mijenjali srodne pozicije. Move funkcija gleda prvo koji je direction unesen u njega (1 = gore, 2 = desno, 3 = dolje, 4 = lijevo) i onda gleda iz kojih pozicija se smije u tome smjeru kretati i ako ne smije kao default ispisuje „Sudarili ste se sa zidom“. Ovo je izvedeno preko 2 ugnježdena switch case-a. Na samome kraju vraća vrijednost nove pozicije.

**Četvrto** što radi if(keyUp == true) je poziva serverCom() funkciju, to smo također mi složili i s time osiguravamo da prilikom kretanja se naš PlayerDot pomakne i kod kolege s kojim smo spojeni putem TCP/IP vezom.



12 ServerCom funkcija

O čemu je riječ, funkcija prima int indikator što mi radimo. Ako je indikator = 1, riječ je o regularnom slanju chat poruke. Ako je 2-5 riječ je o enkodiranjom ponašanju, šaljemo string od 20 (prethodno nasumično generiranih) znakova, koje server na drugoj strani hvata:  i srodno tome pomiče PlayerDot2. Bitno je primjetiti da player 2 se pomiče na serveru zato što je korisnik sam sebi uvijek player 1 dok je on sam kod spojenog partnera player 2. Ovako osiguramo da oba igrača vide ispravno kretanje svojeg i tuđeg lika.



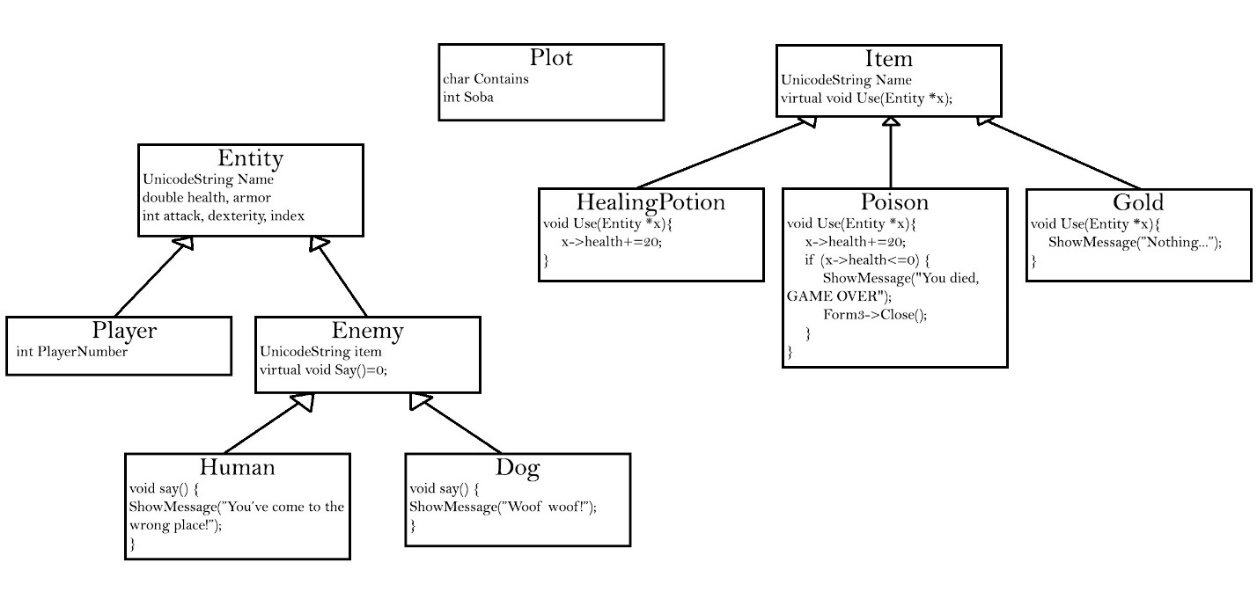
13 Random.Org

Ovo je pozivanje player objekta:



Player 2 ime smo mi kao Easter Egg stavili naš potpis zato što se ta informacija nikad ne uporabljuje u samoj aplikaciji.

A ovo je UML dijagram svih klasa i nasljeđivanja u pitanju:

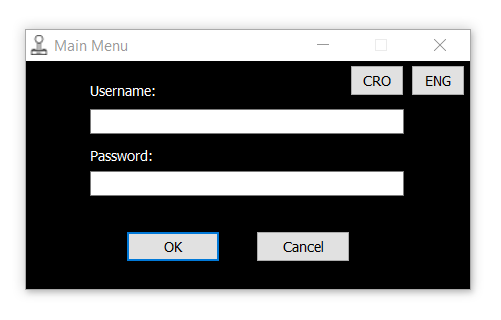


14 UML dijagram svih klasa

Preko polimofizma smo izbjegli solidnu količinu redundantnog koda, u situacijama gdje se poziva npr. Combat funkcija ne moramo naglasiti da se specifično tuče igrač protiv psa, nego je rijec o entity protiv entity. Sva 3 objekta u igri također se referenciraju kao Item->Use() umjesto tri različita poziva.

# 3. Redoslijed pozivanja formi i srodne funkcionalnosti:

## 3.1. Login – Glavni izbornik



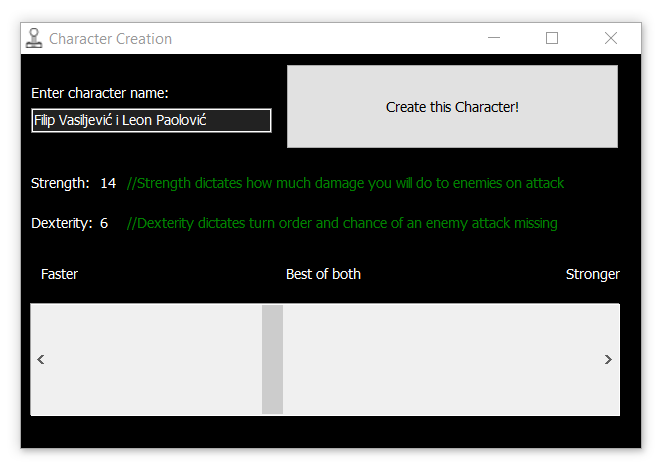
15 Glavni Izbornik

Prilikom paljenja aplikacije korisniku se prikaže glavni izbornik koji služi kao login. Na ovoj formi može odabrati lokalizaciju jezika na hrvatski ili engleski. Ovisno o unesenim podacima se može dogoditi jedno od 3 stvari:

* Korisnik nije ispunio neko polje -> Poruka greške
* Korisnik unese svoje ispravne korisničke podatke -> Nastavak u aplikaciju
* Korisnik unese nove podatke -> Registracija novof korisnika i nastavak u aplikaciju

Razumno, Cancel gumb gasi aplikaciju.

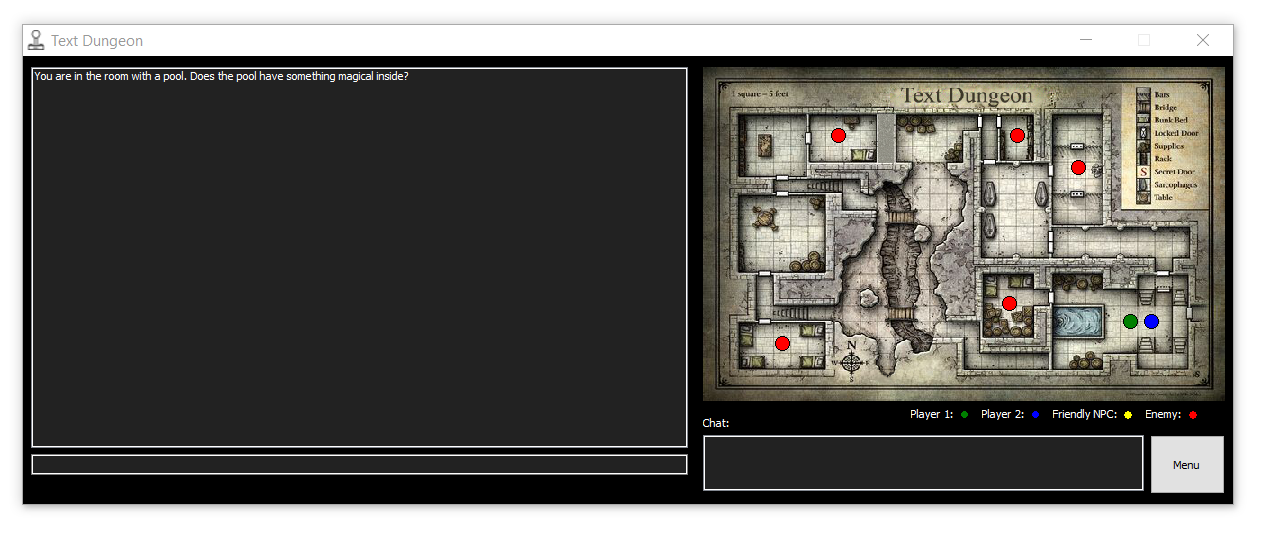
## 3.2. Character creation – Kreacija lika



16 Kreacija lika

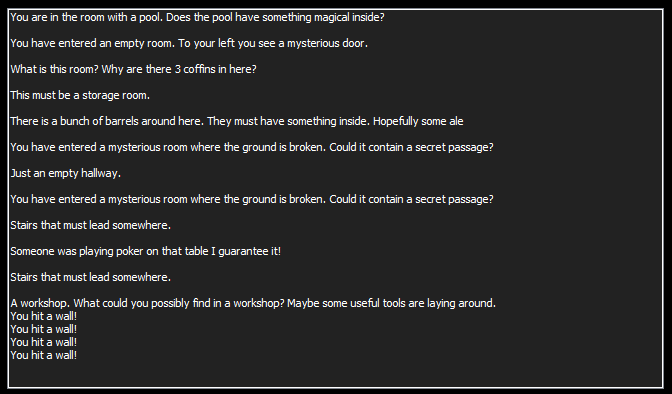
Ova forma je veoma jednostavna, na dnu ima jedan slider s kojim korisnik bira brzinu i snagu svojeg lika. Logika u pozadini je također jednostavna. Snaga se računa kao pozicija na slideru + 10 (balancing da korisnik pobijedi svaki combat) dok je brzina (dexterity) uvijek 10-snaga. Ove varijable se pohranjuju u prikladne varijable i onda se njima dostupa iz druge forme preko taga extern. Create this Character gumb otvara glavnu formu igre.

## 3.3. Text Dungeon – mainForm



**17** Glavna Forma

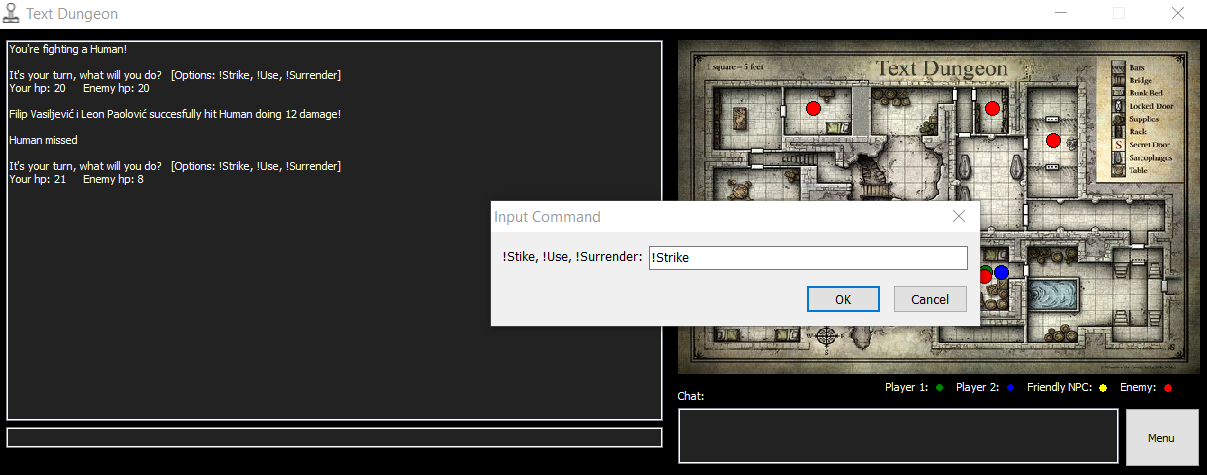
Ovo je prikaz glavne forme, gornji lijevi okvir jer TListView koji smo nazvali Display i u njega ispisujemo opis sobe prilikom kretanja kroz Plot-Grid:



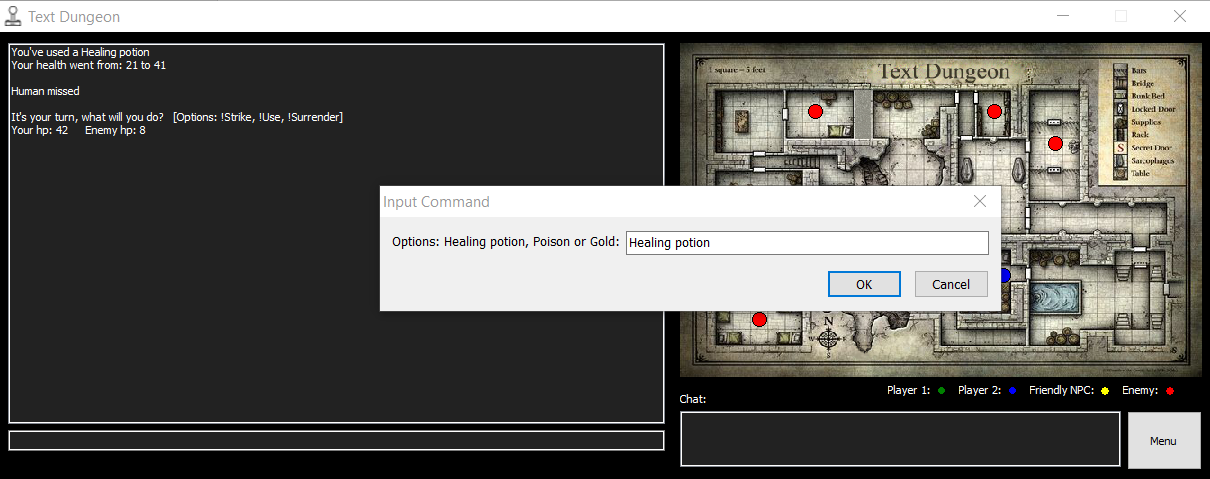
18 Display

Također postoji clearCounter u programu koji broji koliko je puta nešto bilo upisano u Display te nakon 14 upisa se resetira natrag na 0 i radi Display->Items->Clear(); što briše sve iz Displaya.

Ispod displaya je prazni TEdit, kojeg smo imenovali Command, može primiti ili normalni string ili našu tekstualnu naredbu. Sve tekstualne naredbe počinju sa ! i time se poziva functionSort iz DLL-a. Function sort testira o kojem je stringu riječ, ako je neka naša komanda !Strike, !Use, !Me ili !Napadni, !Iskoristi, !Opis će vratiti srodni int koji se main formi dalje koristi u switch case (ovo rješenje je optimalno jer je prosljeđivanje reference na TListView element u funkciju iz DLL datoteke nepogodni način rukovanja s memorijom te može doći do neočekivanih ispisa (ako upće dođe do ispisa)). Iz tog razloga smo što sadrži ispis u Diplay ostavili u .cpp dok je logičko rukovanje koji je koji case unutar DLL-a.



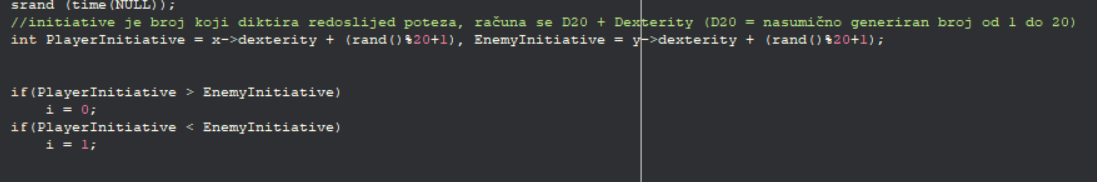
19 Combat



20 Iskoristi

Slike reprezentiraju poziv Combat funkcije kada igraci stanu na poziciju gdje se također nalazi i nerpijatelj.

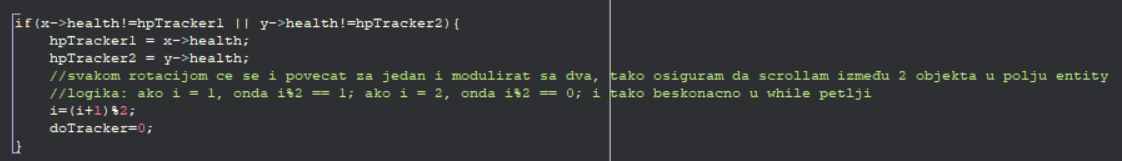
Ta specifična Combat funckija se vrti u while petlji koja prati dok su oboje igrač i neprijatelj živi (HP > 0) i izmjenjuje redoslijed poteza između igrača i neprijatelja. Redoslijed poteza se određuje tako što se na dexterity (brzinu) od oboje igrača i neprijatelja zbroji nasumično generiran broj 0-20 (initianitve roll preuzet iz 5e Dungeons and Dragons pravila) i gleda se čiji je initiative veći.



21 Logika poteza 1

Aplikacija je relativno jednostavna, ako je neprijateljev potez će uvijek napadati, a ako je igračev potez on može birati između napada, korištenja predmeta ili predaje (predaja daje Game over i gasi aplikaciju). U oba slučaja napad ima šansu da promaši, dexterity + dice roll 0-20 iz funkcije dodgeCalculator iz DLL-a.

Nakon svakog poteza se izvršava sljedeća logika:

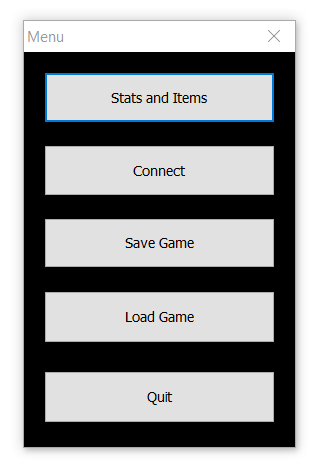


22 Logika poteza 2

Ako dođe do promjene u hp-u (dodge regenerira hp za 1) i se postavlja u 1 ili u 0. Pošto je riječ o while petlji doTracker pazi da se sve korisniku ispiše samo jednom. (Tokom developanja je ovo bilo problematično jer bi bez toga se aplikacija zaledila u beskonačnoj petlji ispisa gdje korisnik nije mogao stići unjeti naredbu).

Menu gumb otvara sideMenu (sporedni izbornik) formu u kojoj se nalaze ostale opcije.

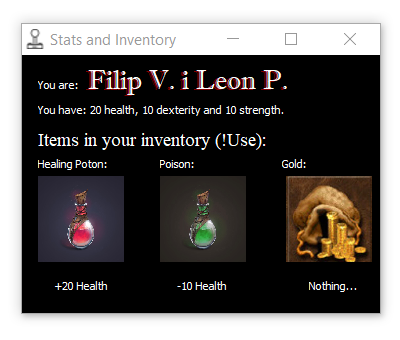
## 3.4. Side menu – Sporedni izbornik



23 Sporedni izbornik

## 3.4.1. Stats and Items – Podaci i Oprema

Otvara novu formu u kojoj se ispisuje trenutno ime lika, životni bodovi, brzina i snaga (srodno kreaciji lika, extern varijabla iz forme kreacije lika). Također prikazuje neinteraktivni izbornik svih predmeta dostupnih sa !Use ili !Iskoristi te opis šta koji pojedini predmet radi:

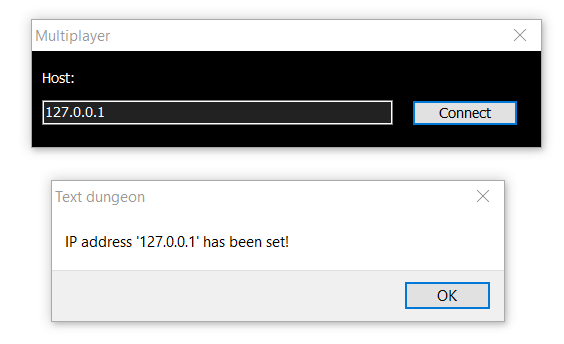


24 Podaci i Oprema

Efekt crvenog sijenčanja iza slova imena igrača je postignut sa 2 TLabel objekta koji se razlikuju za 1 piksel u poziciji na formi.

## 3.4.2. Connect – Spoji me

Otvara malu formu, koja sa extern varijablom u mainForm prosljeđuje UnicodeString vrijednost, koja se dalje prosljeđuje našoj prethodno opisanoj serverCom() funkciji.

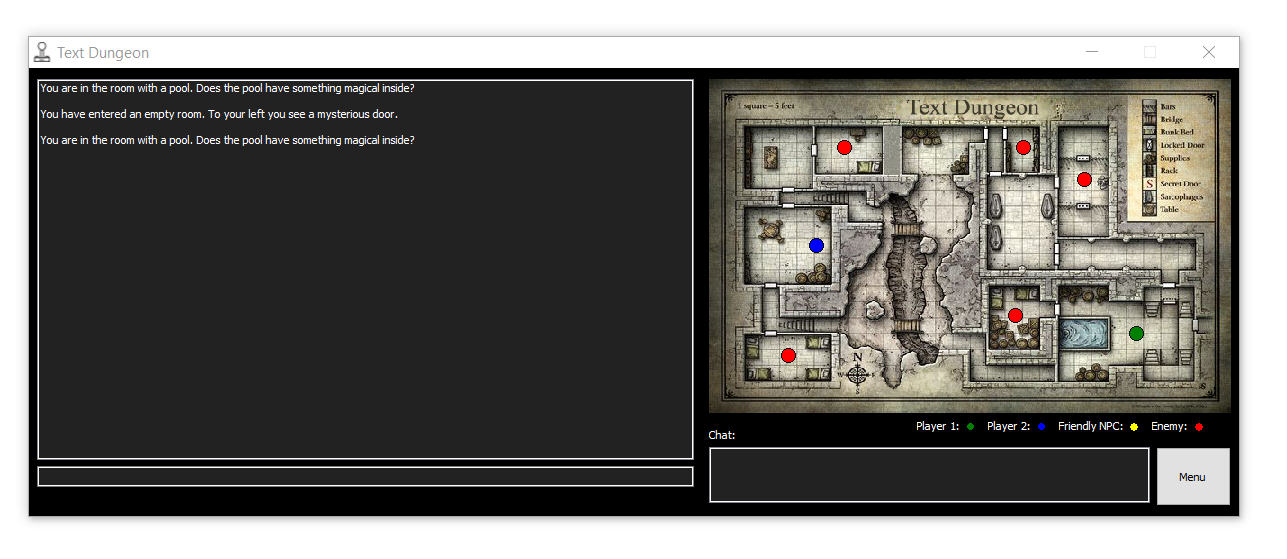


25 Spajanje na IP adrese

Singleplayer je izveden tako što je lokalna adresa od 127.0.0.1 po defaultu postavljena, a multiplayer se izvodi tako što korisnik unese ip adresa od kolege s kojim igrate. Napomena, prvobitno je potrebno ugasiti firewal i napraviti iznimku u routeru ili u našem slučaju kreirati virtualni lan pa se sa time spajati. Mi smo virtualni lan, po dogovoru sa proferosom, realizirali preko alata hamachi:



26 LogMeIn Hamachi



27 Prikaz rada multiplayer-a (plavu točku je kolega pomicao)

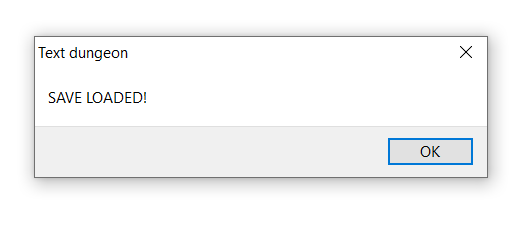
## 3.4.3. Save Game i Load Game

Ovi gumbi ne otvaraju novu formu, jednostavno ili čitaju i podatke iz xml-a pohranjuju u .cpp ili podatke iz .cpp spremaju u novi istoimeni XML (podaci i liku i podaci o svijetu). Jedino što korisnik dobije je poruku potvrde:



28 Pohrana igre

ili



29 Učitavanje igre

## 3.4.4. Quit – Izlaz

Osnovni gumb za izlaz, isto kao i X znak u gornjem desnom kutu gasi aplikaciju.

# 4. Zaključak

Text Dungeon aplikacija služi više kao dokaz koncepta (proof of koncept) i kao igra je dosta skromna (pogotovo u usporedbi s onim što je prisutno na tržištu igri poput steam-a itd.). U trenutnom stanju samo je implementacija svih funkcionalnosti obavljena i kad bi se išla prava igra iz toga raditi bi se veoma lagano samo nadodavalo.

Text Dungeon je potom samo funkcionalni kostu koji služi samo za prikaz mogučnosti svih implementiranih funkcionalnosti.

Svim silama smo se trudili minimizirati repeticiju koda i to smo izveli pomoću polimorfizma u funkcijama akcija. Također smo isto postigli uporabom pokazivača kada kontekstualno treba baratat s istim objektima (PlayerDot1 i PlayerDot2). Kao rezultat toga program je veoma fleksibilan i mogu se lagano dodavati novi neprijatelji u obliku klasa potomka Enemy klase. Isto vrijedi za objekte kao potomke klase Item.

Jedino rješenje koje punopravno priznajemo da smo mogli bolje je lokalizacija, funkcionalna je ali nije optimalno u načinu koji smo mi izveli.

Seminar izradili: Filip Vasiljević i Leon Paolović

# 5. Literature

[1] NAPREDNE TEHNIKE PROGRAMIRANJA U C++ BUILDERU - Željko Kovačević, struč.spec.ing.techn.inf. - Link: <https://www.dropbox.com/s/k82ffr580vzqx1t/Napredne%20tehnike%20programiranja%20u%20C%2B%2B%20Builderu.pdf?dl=1>

[2] VCL Examples – You Tube – Link: <https://www.youtube.com/user/vclexamples>

[3] Stack Overflow – Razno – Link: <https://stackoverflow.com/>

[4] Dungeons and Dragons (karta preuzeta iz Lost mines of Phandelver) – Wizards of the coast – Link: <https://company.wizards.com/>

[5] Dungeons and Dragons 5e pravila (računanje initiativa, dodge i genralna inspiracija iza rada) – Wizards of the coast 5e basic rules – Link: <https://media.wizards.com/2018/dnd/downloads/DnD_BasicRules_2018.pdf>

[6] LogMeIn Hamachi – Link: https://www.vpn.net/

[7] Random.org – Link: https://www.random.org/