

Mestna občina Celje
Komisija Mladi za Celje

BOUNCY WAR

Raziskovalna naloga

AVTORJI:

Mitja Brezovnik,
Patricij Jurgelj,
Niko Kolar

MENTOR:

mag. Boštjan Resinovič, univ. dipl. inž.

Celje, marec 2016

Šolski Center Celje,
Srednja šola za kemijo, elektrotehniko in računalništvo

BOUNCY WAR

Raziskovalna naloga

AVTORJI:

Mitja Brezovnik,
Patricij Jurgelj,
Niko Kolar

MENTOR:

mag. Boštjan Resinovič, univ. dipl.inž.,

Mestna občina Celje, Mladi za Celje

Celje, marec 2016

Kazalo vsebine

1	Uvod.....	6
1.1	Opredelitev problema	6
1.2	Hipoteze.....	6
1.3	Izbira raziskovalnih metod	7
2	Igre	7
2.1	Nasploh o igrah.....	7
2.2	Bouncy war.....	8
3	Praktični del.....	8
3.1	Delo z Unity.....	8
3.2	Spletna stran	13
3.3	Bluetooth in spletne storitve	17
3.3.1	Bluetooth	17
3.3.2	Spletne storitve	18
4	Analiza rezultatov	22
5	Zaključek.....	25
6	Zahvala	26
7	Literatura	26

Kazalo slik

Slika 1: Igra Flappy Bird	7
Slika 2: Prijavna stran	9
Slika 3: Glavni meni.....	9
Slika 4: Izbira načina igre.....	10
Slika 5: Povezava z nasprotnikom	10
Slika 6: Prikaz igre	11
Slika 7: Komponenta trčenje v 2D-okolju.....	11
Slika 8: Komponenta fizike v 2D-okolju	12
Slika 9: Komponenta žogice	12
Slika 10: Trgovina v igri	12
Slika 11: Ustvarjanje novega projekta.....	13
Slika 12: Začetek z praznim projektom.....	14
Slika 13: Vsebina začetnega projekta.....	14
Slika 14: Dodajanje vseh potrebnih komponent	14
Slika 15: Izgled domače strani po vizualni obdelavi s .css datotekami	15
Slika 16: Desni klik na Reference	15
Slika 17: Dodajanje spletne storitve.....	16
Slika 18: Koda za registracijo uporabnika	16
Slika 19: Logo Android Studia.....	17
Slika 20: Metoda, ki vključi bluetooth	17
Slika 21: Podatkovna baza	19
Slika 22: Spletna metoda, ki vpiše podatke v tabelo Igralec ter High_score (najboljši dosežek)	20
Slika 23: Objavljanje spletne aplikacije	20
Slika 24: Uporabnik v bazi.....	21
Slika 25: Dodeljevanje pravic	21
Slika 26: Osnovni podatki o anketirancih	22
Slika 27: Vprašanje 1	22
Slika 28: Vprašanje 2	22
Slika 29: Vprašanje 3	23
Slika 30: Vprašanje 4 in 5	23
Slika 31: Vprašanje 6	24
Slika 32: Vprašanje 7	24
Slika 33: Vprašanje 8 in 9	25

Povzetek

Ker smo želeli ustvariti igro z namenom, da ugotovimo, če naše šolsko znanje programiranja zadostuje za izdelavo le-te, smo se odločili, da poustvarimo, nam vsem poznano, klasično igro Pong, ki deluje na operacijskem sistemu Android. Igro smo ustvarili v nekoliko novejšem stilu in jo preimenovali v Bouncy War. V sklopu te naloge smo naredili igro v razvijateljskem okolju Unity in jo povezali z bazo v SQL Serverju. Povezavo smo naredili s pomočjo spletnega strežnika IIS, nanj pa smo nato povezali še spletno stran narejeno v ASP.NET spletnem okolju. Da smo lahko zadano nalogo opravili, smo se naučili in dogradili naše znanje v jezikih C#, CSS, Java in HTML. Tako smo ugotovili, da programersko znanje na šolski ravni ne zadovoljuje programerskih potreb, ki smo jih morali zadovoljiti za uspešno dokončanje projekta.

Abstract

Since we wanted to create a game, in order to determine if our school programming knowledge is enough to make it, we have decided to recreate, to all of us known, classic game of Pong, which runs on Android operating system. We created the game in slightly newer form and renamed it to Bouncy War. As part of the task, we have created a game in Unity and connected it to the database in SQL Server. The connection was made with server created by IIS (Internet Information Services) web server, on which we later also connected our official website made in ASP.NET web environment. In order to make things work we had to upgrade our knowledge of programming languages like C#, CSS, Java and HTML. Thus, we find, that our school programming knowledge does not meet the programming needs, we had to satisfy for the successful completion of the project.

Ključne besede

Pong, Android igra, Unity, baza, SQL Server, spletna stran, ASP.NET, C#, Java, HTML, CSS, bluetooth, spletna storitev

Keywords

Pong, Android game, Unity, database, SQL Server, website, ASP.NET, C#, Java, HTML, CSS, bluetooth, web service

1 Uvod

Z raziskovalno nalogo smo želeli trije učenci, željni znanja, ustvariti igro za Android, imenovano »Bouncy War«. Naše delo smo si porazdelili, tako da je vsak naredil del naloge in tako smo prišli do končnega rezultata.

Razlogov, zaradi katerih smo se odločili za takšno vrsto raziskovalne naloge, je veliko. Sprva smo želeli ustvariti le igro, namenjeno širšemu krogu uporabnikov, ki bi bila dostopna preko portala Google Play, nato pa smo si zadali cilj, da naredimo preprosto igro, zanjo pa nato ustvarimo še podatkovno bazo in jo uporabimo na naši uradni spletni strani. Igra naj bi prvotno potekala med 2 igralcema, ki se na igro povežeta preko bluetooth tehnologije in odigrata prijateljsko tekmo. V prihodnje pa načrtujemo, da bomo igro še razširjali, posodabljali in jo dopolnili še s kakšno novo funkcijo.

1.1 Opredelitev problema

V raziskovalni nalogi bomo poizkušali ugotoviti, ali lahko skupina treh angažiranih sošolcev naredi igro, ki bi bila prepoznavna širši javnosti. Na tem področju smo zelo neizkušeni, zato nam ta problem predstavlja velik izziv. Med samim ustvarjanjem projekta smo imeli kar precej težav, vendar smo jih s pomočjo interneta ter znanja profesorjev rešili.

1.2 Hipoteze

Na začetku te naloge smo si zadali naslednje hipoteze, ki jih bomo podrobneje analizirali. Na koncu bomo preverili njihovo veljavnost glede na dobljene rezultate raziskovanja.

1. **Android igra naj bi bila preprosta, igralcu naj bi predstavljala očiten izziv** (premagati igralca na nasprotni strani igrišča). Takšne igre naj bi bile bolj igrane in zanimive za večji spekter/krog uporabnikov.
2. **Na Android napravah manjka iger, kjer igrata med seboj igralca 1 proti 1.** Ta trditev izhaja iz dejstva, da smo po naših prvih ugotovitvah brez kakršne koli raziskave zaznali, da za Android naprave obstajajo pretežno samo enoigralske in večigralske igre, brez dejanske interakcije med igralci.
3. **Za dokončanje projekta zadostuje osnovno srednješolsko znanje.** Menimo, da bomo morali to hipotezo tekom raziskovalne naloge ovreči, saj smo si zadali kar težko nalogo in bo potrebna tudi uporaba novih znanj, ki jih bomo najbrž pridobili šele ob izdelovanju naše android igre.

1.3 Izbira raziskovalnih metod

Najpomembnejše delo naše raziskovalne naloge bo kreiranje igre »Bouncy War«, v razvijateljskem okolju Unity. Tukaj bo potrebno izboljšati naše predhodno znanje jezika C# in java. Za delovanje naše igre bomo potrebovali strežnik in bazo, naredili pa bomo tudi uradno spletno stran. Uporabljali bomo IIS spletni strežnik, zaradi katerega bomo imeli kasneje dostop do baze in spletne strani. Odločili smo se, da bomo spletno stran naredili v ASP.NET spletnem okolju, do katerega bomo dostopali preko Visual Studia. Tukaj bomo uporabili naše znanje HTML jezika in CSS-ja, prav tako pa bomo potrebovali manjše poznanstvo z C# jezikom. Za upravljanje s podatki bomo naredili bazo v SQL Server Management Studiu. Do te baze bomo dostopali preko spletnih metod (Web Methods), ustvarjenih v Visual Studiu, nato pa bomo te spletne metode uporabljali v naši igri in spletni strani, tako da se bodo vsi podatki o naših uporabnikih shranjevali v isto bazo.

2 Igre

2.1 Nasploh o igrah

Marsikdo bi si želel ustvarjati igre ter se s tem preživljati. Verjetno ima velika večina teh ljudi potrebno znanje, da naredi takšno igro. Vendar dejstvo, da zna posameznik izdelati igro, v večini primerov ni dovolj. Po našem mnenju je na prvem mestu ideja, ki bo igri dodala poseben čar oziroma jo bo naredila edinstveno ter zabavno za igranje.

Menimo, da imajo vse popularnejše igre skupno lastnost. To je preprostost same igre, ki jo naredi toliko bolj zanimivo, zato posledično zajame večji spekter uporabnikov. Na spodnji sliki je igra, ki je pred približno dvema letoma podirala vse rekorde, in sicer tako v višini ocen, prenosov ter časa igranja.



Slika 1-Igra Flappy Bird

Slika najdena na naslovu: <http://unblockedgamez.com/wp-content/uploads/thumbs/custom/F/flappybird.jpg>

2.2 Bouncy war

Glede na to, da v Sloveniji prevladujejo mobilne naprave z operacijskim sistemom Android, smo tudi mi prilagodili igro, namenjeno Androidu. Ob razmišljanju o ideji smo si zadali nekaj smernic, ki jih bomo upoštevali. Te smernice smo določili na podlagi tega, kaj je vseč vsakemu posamezniku pri igrah. Pri nekaterih je to napredovanje po lestvici, nekaterim je cilj premagovati nasprotnike, nekaterim je cilj, da lahko igra s prijateljem oziroma z več prijatelji in še mnogo drugih, vendar vseh želja nismo mogli upoštevati.

Odločili smo se, da bomo preoblikovali eno najstarejših računalniških igric in jo poizkusili narediti ter predstaviti v moderni obliki. Najprej bomo naredili »enoigralski« način, kjer bo uporabnik dobil točko vsakič, ko bo spravil žogico čez zgornjo mejo ekrana. Tukaj bo lahko uporabnik tudi vadil za naslednje, spodaj omenjene načine, kjer bo igral proti resničnemu igralcu.

Eden od načinov igranja bi bil tudi način igranja prek bluetootha, kjer bo uporabnik lahko nekoga izzval na dvoboj. Zmagovalec bo postal tisti, ki bo poslal žogico na drugo stran, nasprotnik pa mu je ne bo uspel vrniti.

Igrico bomo tudi posodabljali. Kasneje bomo dodali še možnost igranja preko interneta. S tem bi nato še zadostili želji, kjer bi dodali možnost napredovanja po lestvici.

V igrici Bouncy war smo implementirali trgovino, kjer bo igralec lahko kupil drugo obliko ploščka in bi mogoče znal presenetiti nasprotnega igralca, saj se bo žogica odbila pod drugačnim kotom, tako da bo težko predvidevati pozicijo žogice. Denar bo mogoče služiti na vseh načinih igranja.

3 Praktični del

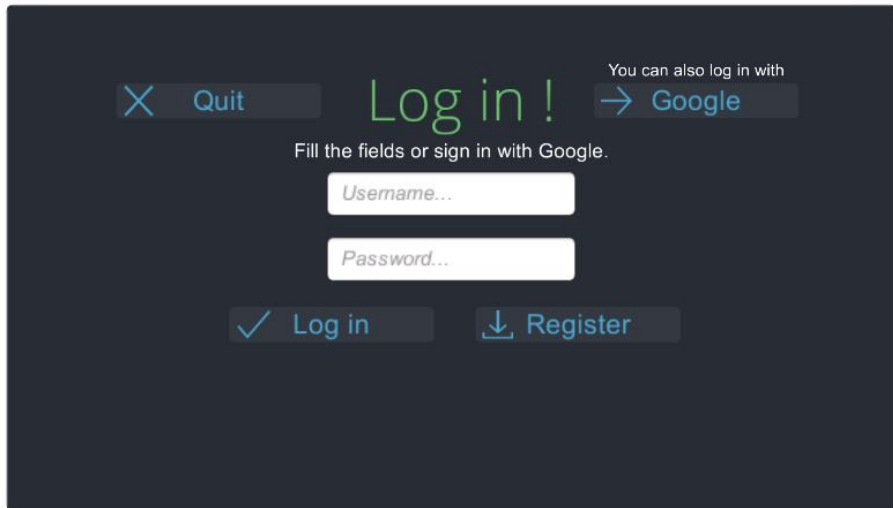
3.1 Delo z Unity

Igro smo sprva nameravali izdelati s programom, imenovanim Android Studio, vendar smo kaj kmalu ugotovili, da bo izvedba veliko težja v Android Studiu, saj ta ne vsebuje tri- ali dvodimenzionalnega okolja, ki je ključen za našo izvedbo samega projekta. Zaradi tega smo se odločili, da aplikacijo za operacijski sistem Android izvedemo v okolju Unity, ki podpira tri- in dvodimenzionalna okolja, prav tako pa ima vključeno tudi računanje fizike. To pomeni, da so gravitacija, trčenja, rotacije, odboji v dvodimenzionalnem in trodimenzionalnem okolju že vključena. Programska koda pa je bila ustvarjena s pomočjo programskega okolja Visual Studio.

Najprej smo izdelali vhodno stran, imenovano "Login" (Prijava). Dodali smo gumbe in vnosna polja, da lahko uporabnik vnese svoje podatke in se prijavi ali registrira kot nov uporabnik. Vpis in registriranje uporabnika se povežejo z našo spletno storitvijo, preko katere

dobi zahtevane podatke in prav tako doda podatke v podatkovno bazo. Kljub dodajanju in vrnitvi podatkov se pred tem izvrši še preverjanje, če uporabnik že obstaja oz. če je trenutno prijavljen. V tem primeru se mu prijava v račun zavrne, saj lahko en račun uporablja le en uporabnik naenkrat.

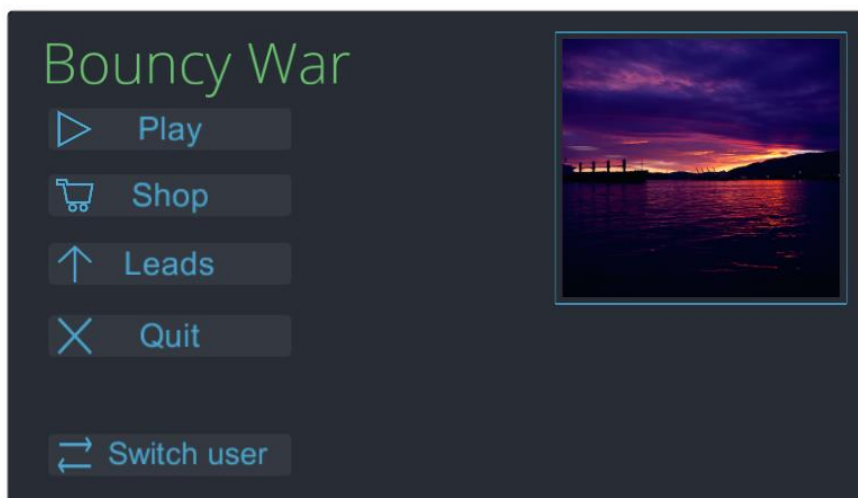
V tej sceni smo prav tako dodali, da se lahko uporabnik vpiše tudi z računom Google.



Slika 2: Prijavna stran

Ko se uporabnik vpiše v svoj račun, se naloži nova scena, in sicer "MainMenu", ki predstavlja glavni meni. V glavnem meniju lahko uporabnik dostopa do drugih menijev v igri z gumbi na zaslonu, kot so:

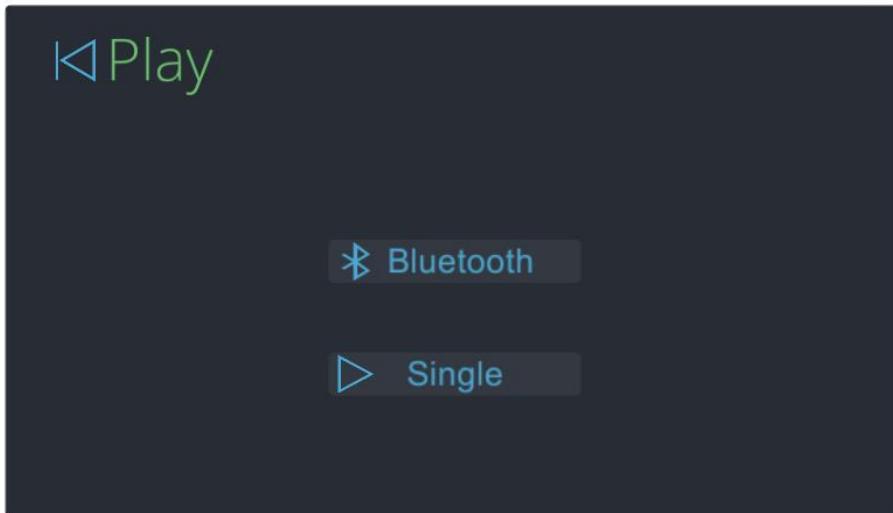
- Play (igraj)
- Shop (trgovina)
- Leads (lestvica najboljših)
- Switch user (zamenjaj uporabnika)



Slika 3: Glavni meni

Menije smo kar se da poenostavili in razumno zložili, tako da so za končne uporabnike čimbolj razumljivi in preprosti.

Uporabnik lahko igro prične s klikom na gumb Play, ki prične nalagati novo sceno. Zatem lahko uporabnik izbere med Single (1 igralec igra) in pa Bluetooth (en igralec proti drugemu – »večigralni« način).



Slika 4: Izbira načina igre

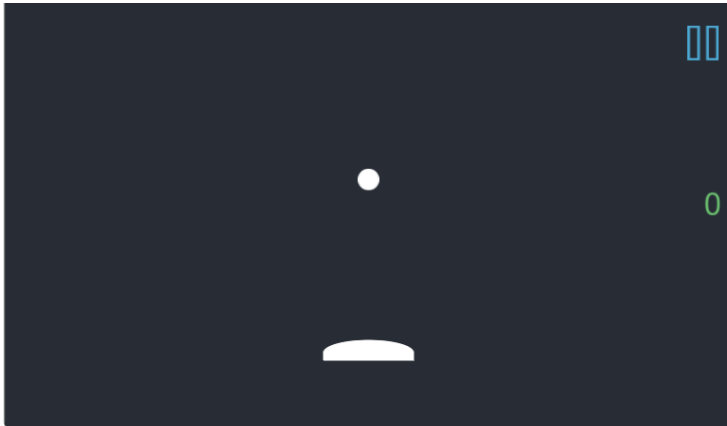
Za oba Single in Bluetooth se naložita novi sceni, kjer lahko uporabnik igra igro, in sicer v Single načinu lahko uporabnik nemudoma prične z igro, medtem ko se mora pri Bluetooth uporabnik najprej povezati z drugim uporabnikom z Bluetooth povezavo. To stori tako, da se mu na začetku scene prikaže diapozitiv, na katerem lahko počaka na povezavo z drugega pametnega telefona ali pa pritisne na Connect (poveži) gumb in se prikaže zaslon, na katerem se izpišejo vse najdene Bluetooth naprave v bližini. S klikom na eno od najdenih naprav se napravi lahko povežeta, če druga naprava to prošnjo sprejme.



Slika 5: Povezava z nasprotnikom

Ko se napravi končno povežeta, se igra prične. Igralec lahko igro zmaga tako, da zbere 10 točk. Točko pa dobi, če na nasprotno stran igralne površine (igrišča) odbije žogico in se ta prebije do točke za drugim igralcem.

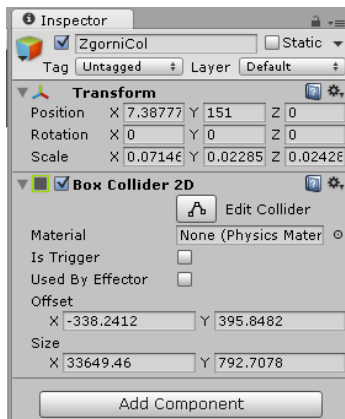
V Single načinu pa je to malo drugače, saj tam nimamo še enega igralca. V tem načinu igralec dobi točko, ko jo spravi čez vrh zaslona. Nato žogica dobi naključen naklonski kot in hitrost, tako da je igralcu zanimivo in ne preveč enostavno.



Slika 6: Prikaz igre

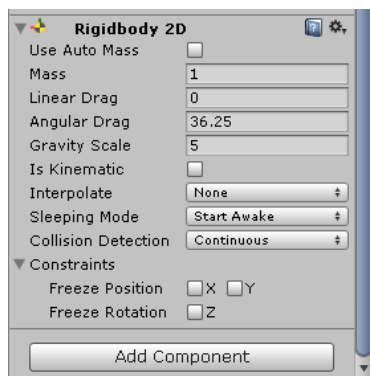
Sama fizika v teh dveh načinih je narejena iz več delov, in sicer kot prvo stvar smo dodali vsa območja, do koder lahko žogica gre oz. označba igralne površine.

Tem območjem smo dodali komponento, ki se imenuje “Collider2D”, kar pomeni, da je objekt trdnega stanja in je procesiran v 2D-okolju.



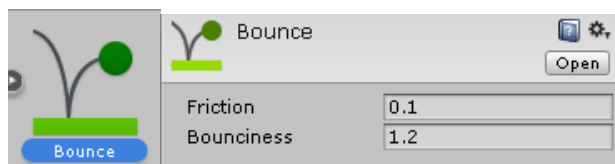
Slika 7: Komponenta trčenje v 2D-okolju

Igralec se mora tudi premikati, zato smo dodali prazen objekt, ki ga lahko igralec premika z nagibanjem. Prav tako smo mu dodali komponento “Rigidbody2D”, ki služi, da se naš objekt popolnoma odziva na trčenja s trdnimi objekti. Nato smo na ta prazen objekt dodali še različne oblike, kakršne lahko uporabnik kupi v zaslonu “Shop”.



Slika 8: Komponenta fizike v 2D-okolju

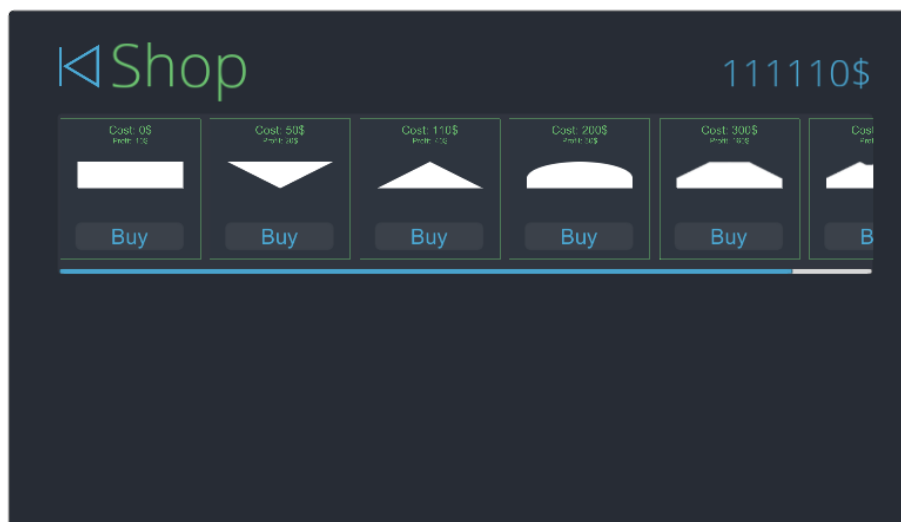
Nato smo dodali še žogico, ki je prav tako dobila komponento “Rigidbody2D”, služi pa za enake namene kot pri premikanju igralca, vendar ta objekt se mora samodejno odbijati in imeti neko hitrost že na začetku. Tako smo s programsko skripto omogočili začetno hitrost oz. jo nastavili. Prav tako smo dodali nov material, ki označuje njeno odbijanje. To odbijanje smo nastavili tako, da se odbije z večjo hitrostjo, kot je bila začetna.



Slika 9: Komponenta žogice

V zaslonu “Shop” lahko uporabnik kupuje nove oblike igralčeve ploskve, s katero odbija žogico.

Oblike smo nastavili tako, da dražja ploskev prinese več dobička, kar načeloma pripelje do tega, da na nek način privabimo uporabnike k večjemu igranju igre.



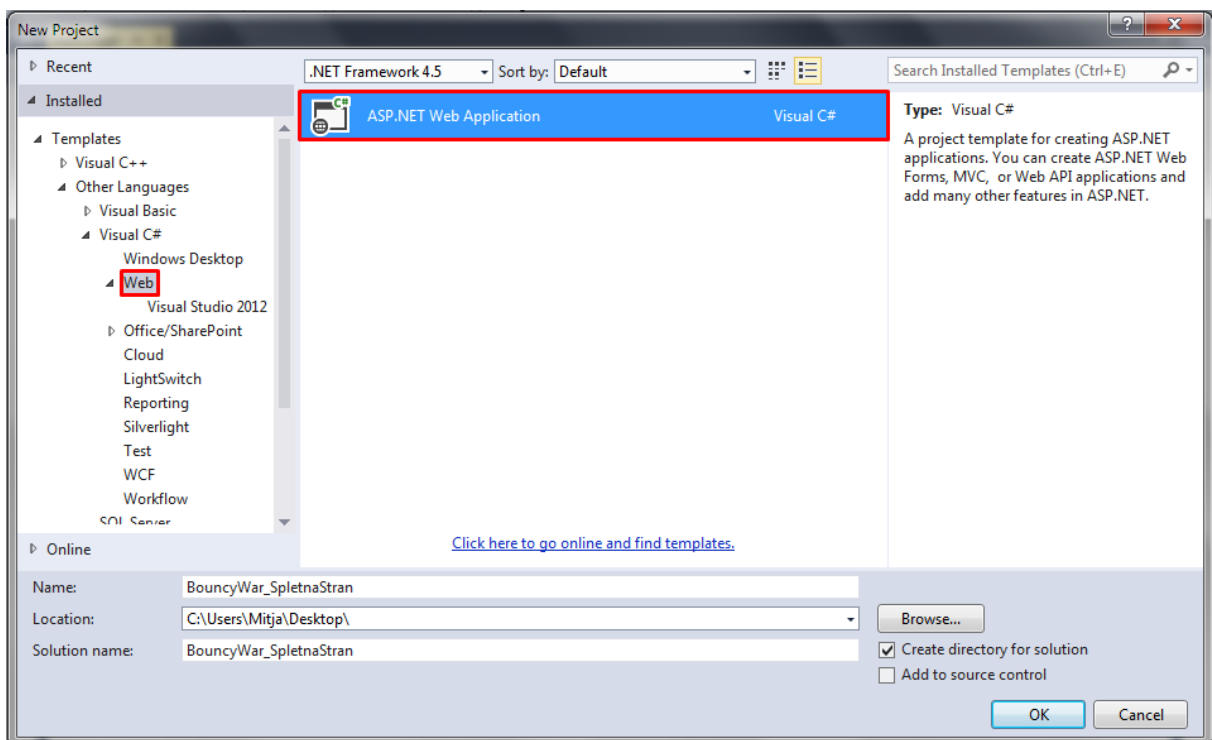
Slika 10: Trgovina v igri

3.2 Spletna stran

Odločili smo se, da bomo spletno stran za našo aplikacijo naredili preko Visual Studia v programskem okolju ASP.NET. To je orodje, ki nam omogoča oblikovanje spletnih strani z uporabo HTML, CSS, C# in JavaScript jezikov. Glavni razlog za takšno odločitev je, da smo se z ASP.NET na kratko seznanili že v tretjem letniku in nam bo to znanje odlično izhodišče za nadaljnje delo.

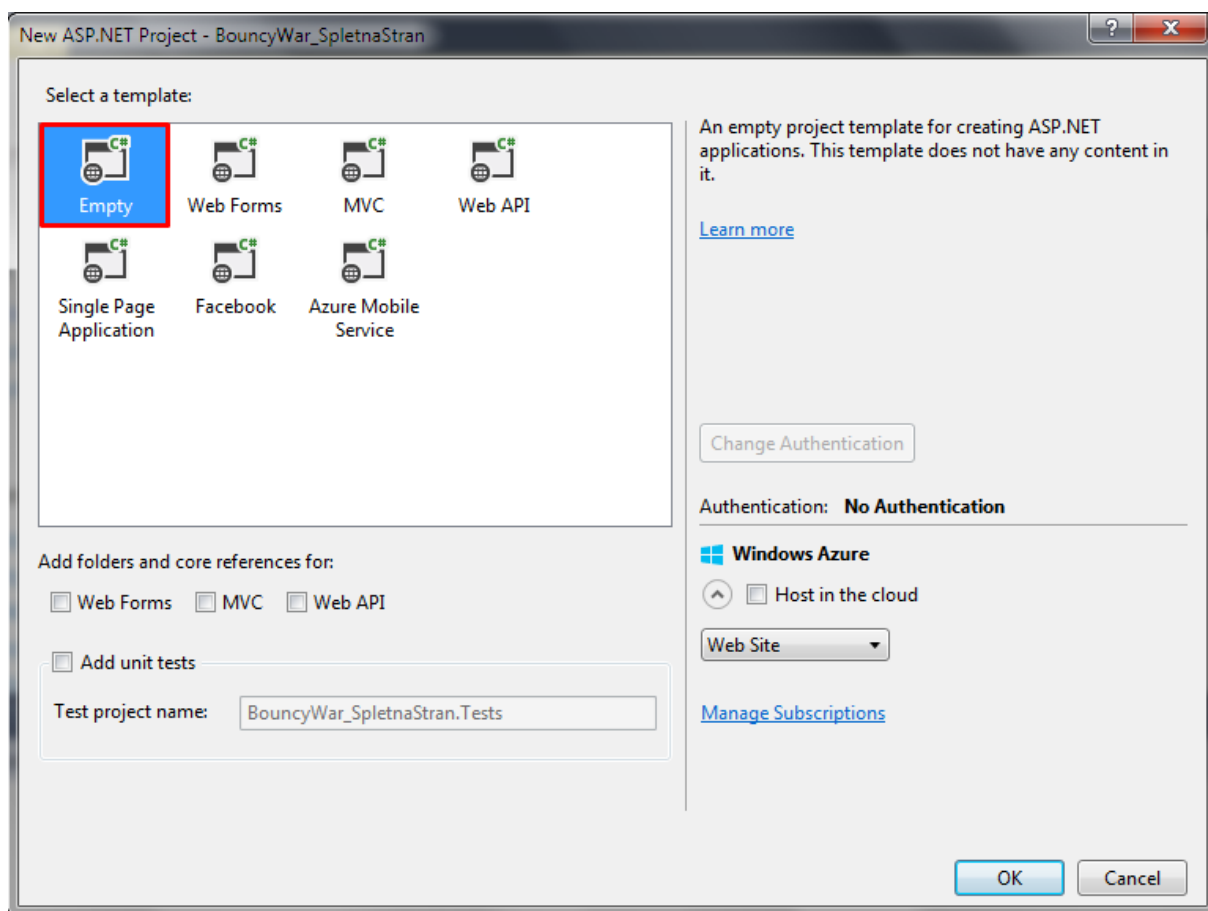
Namen te spletne strani je promovirati našo igro in omogočiti uporabnikom prijavo v spletno stran, kjer imajo vpogled v lestvico najboljših na svetu.

Ker smo imeli Visual Studio nameščen na računalnik že prej, saj smo ga uporabljali že za šolske naloge, smo lahko kar takoj začeli z delom. Ob zagonu smo ustvarili nov projekt in označili ASP.NET Web Application predlogo (ang. template).



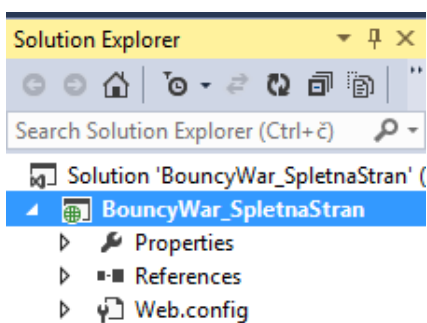
Slika 11: Ustvarjanje novega projekta

Nato smo določili specifično predlogo, s katero bomo zagnali naš projekt. Odločili smo se, da bomo začeli čisto od začetka, zato smo izbrali prazno predlogo, ki na začetku ne vsebuje praktično ničesar.

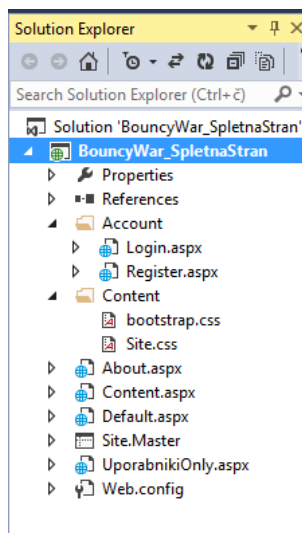


Slika 12: Začetek s praznim projektom

Ko je projekt bil ustvarjen, smo začeli razmišljati o vsebini in tako dodali najpomembnejši del spletne strani, in sicer »Master Page« ter na njo vezali vse nadaljnje strani. Tako smo dodali še strani »Home« (osnovna spletna stran), »About« (o nas), »Contact« (kontakti) in »Leaderboard« (lestvica najboljših). Ker pa moramo naše uporabnike registrirati in hraniti v bazo, smo naredili še strani »Register« (registracija) in »Login« (prijava).

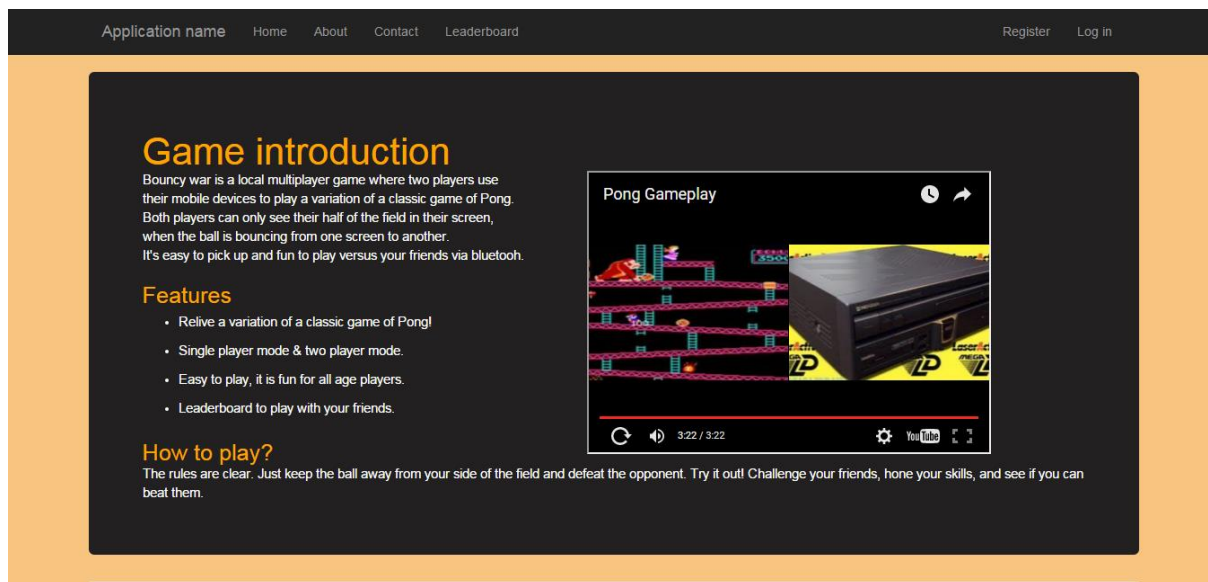


Slika 13: Vsebina začetnega projekta



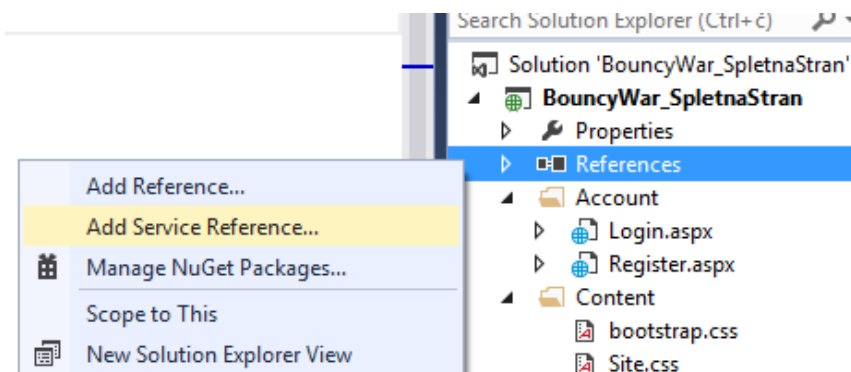
Slika 14-Dodajanje vseh potrebnih komponent

Sledilo je dodajanje tekstovne vsebine našim spletnim stranem in ko je bila vsa vsebina na svojem mestu, smo morali naše strani malo polepšati. Tako smo dodali v naš projekt še Site.css in bootstrap.css datoteki, s katerima smo z uporabo jezika CSS vizualno oblikovali strani. CSS je tako imenovani slogovni jezik, ki opisuje, kako naj bodo elementi na strani postavljeni. In tako smo končali z vsebinskim ter estetskim delom naših spletnih strani.

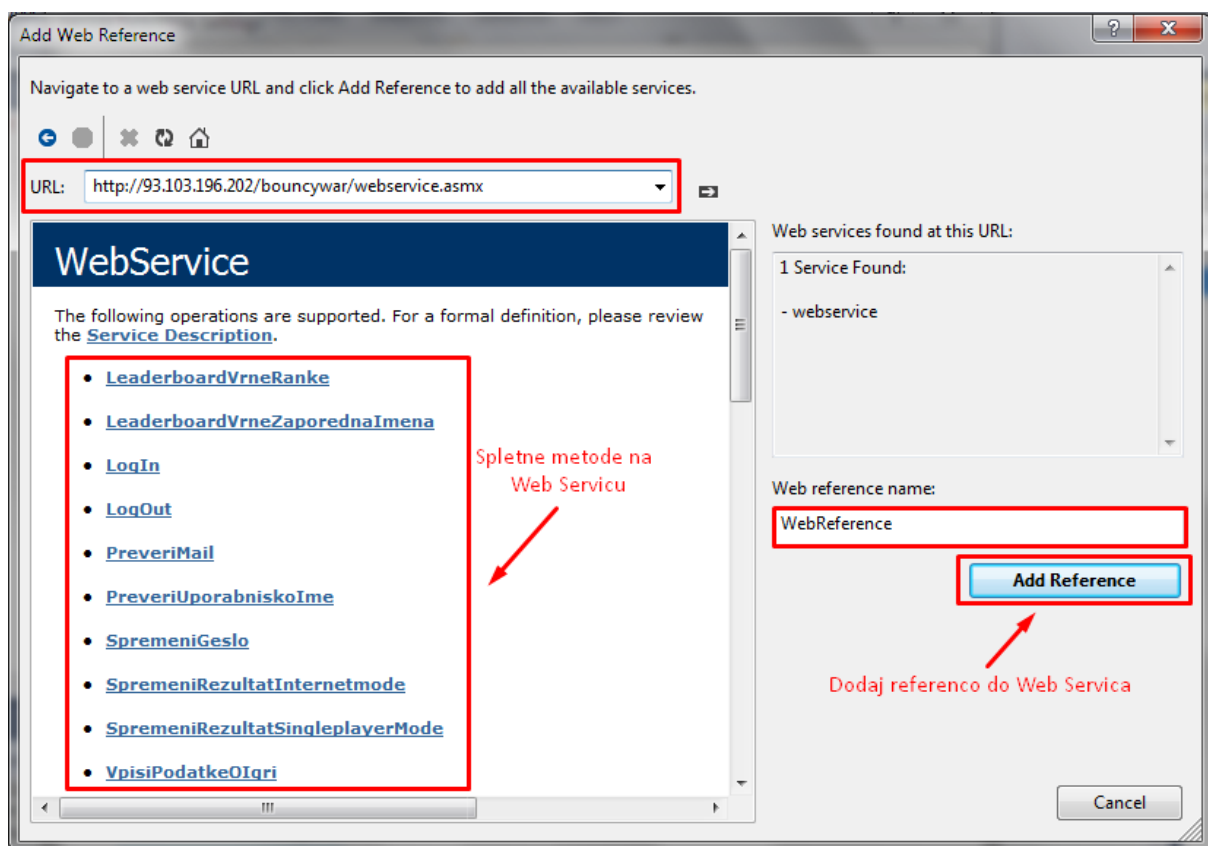


Slika 15: Izgled domače strani po vizualni obdelavi s .css datotekami

Sledilo je dodajanje kode v ozadje spletne strani. Pri tem se nanašamo predvsem na kodo za prijavo in registracijo uporabnika. Ker pa smo morali podatke uporabnika pri registraciji in posredno tudi pri prijavi shraniti v bazo, smo morali našo spletno stran povezati s spletno storitvijo (ang. Web Service). To smo storili tako, da smo kliknili zavihek »Dodaj spletno storitev«, šli pod napredne nastavitve (ang. Advance) in tam pritisnili na gumb »Dodaj spletno referenco«. Odprlo se nam je okno, kjer smo morali določiti URL naše spletne storitve in njeno ime. S pritiskom na gumb je bila ustvarjena referenca. Tako smo dodali referenco na konkretno spletno storitev v IIS spletnem strežniku.



Slika 16-Desni klik na Reference



Slika 17: Dodajanje spletne storitve

Sedaj smo lahko uporabljali spletne metode (ang. Web Methods), s katerimi smo lahko dostopali do podatkov v bazi, jih uporabljali in spreminjali. S pomočjo teh metod smo naredili registracijo uporabnika in njegovo prijavo. Pri tem delu pa smo uporabljali še tako imenovane seje (angl. Sessions). Te seje se uporabljajo za shranjevanje in pridobivanje podatkov uporabnika, ko se sprehaja po naših spletnih straneh. Seja se začne, ko se uporabnik prijavi na našo spletno stran (Log in) in konča šele, ko se odjavi iz nje (Log out) ali pa ko uporabniku poteče časovna omejitev (ang. timeout). Za naše seje smo uporabljali način **InProc**, saj je tako dostop do podatkov zelo hiter in lahek. Na spodnji sliki lahko vidimo kodo, ki se izvrši ob kliku na gumb »Register«. To je seveda le delček kode, uporabljene na naši spletni strani.

```
protected void CreateUser_Click(object sender, EventArgs e)
{
    BouncyWarService.WebService WS = new BouncyWarService.WebService();

    if (WS.PreveriUporabniskoIme(Name.Text) == true)
    {
        Napaka.Text = "Username " + Name.Text + " is taken, please choose another one.";
        Napaka.Visible = true;
    }

    if (WS.PreveriMail(Email.Text) == true)
    {
        Napaka2.Text = "Email " + Email.Text + " is taken, please choose another one.";
        Napaka2.Visible = true;
    }

    if (WS.PreveriUporabniskoIme(Name.Text) == false && WS.PreveriMail(Email.Text) == false)
    {
        WS.registrirajIgralca(Name.Text, Password.Text, Email.Text, "", 1, 1);

        Session["Name"] = Name.Text;

        Response.Redirect("~/Default.aspx");
    }
}
```

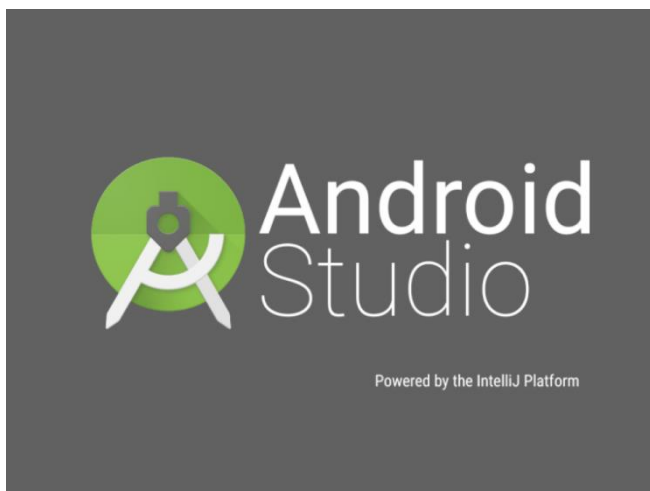
Slika 18: Koda za registracijo uporabnika

Naša spletna stran pa ima tudi stran, ki je dostopna le prijavljenim uporabnikom. In sicer je to stran Leaderboard (lestvica najboljših). Če želimo do te strani dostopati kot anonimen uporabnik, nas stran preusmeri na stran za prijavo. Navsezadnje, ko je bilo vse to že za nami, smo stran objavili (ang. Publish) na IIS spletni strežnik (angl. Web Server).

3.3 Bluetooth in spletne storitve

3.3.1 Bluetooth

Na začetku smo imeli probleme, saj si nismo znali porazdeliti dela. Ker je v tem primeru povezava med telefonoma tudi zelo pomembna, je nekdo moral narediti tudi to. Na tem področju smo bili popolnoma brez izkušenj. Najprej smo začeli z razvijanjem povezave preko bluetootha. To smo delali v programskem okolju Android Studio s programskim jezikom Java.



Slika 19: Logo Android Studia

Sprva je bilo to težko, saj se na ta jezik nismo spoznali dobro. Zelo smo si tudi pomagali z internetom ter znanjem naših profesorjev.

Začeli smo tako, da smo naredili metodo, ki lahko vključi bluetooth.

```
91 |  
92 | }  
93 | private void turnOnBT() {  
94 |     // TODO Auto-generated method stub  
95 |     Intent intent =new Intent(BluetoothAdapter.ACTION_REQUEST_ENABLE);  
96 |     startActivityForResult(intent, 1);  
97 | }
```

Slika 20: Metoda, ki vključi bluetooth

Vidimo tudi izgled kode v Android Studiu, ki pa se nam ne zdi ravno najboljši, saj smo naučeni drugačne sintakse. To se vidi že po položajih zavrtih oklepajev.

Tukaj se nam je rahlo zapletlo, saj nismo točno razumeli delovanja komunikacije preko bluetooth povezave. Po dolgotrajnem brskanju smo nekako prišli do ugotovitve, da je napravo najprej treba najti, napravi se morata spoznati in šele nato povezati. Ko je vse to narejeno, si lahko napravi pošiljata sporočila.

V projekt smo dodali Android komponento, imenovano ListView, v kateri smo prikazovali vse najdene naprave. To smo naredili tako, da smo naredili metodo, ki smo jo klicali za vsako, ko je uporabnik pritisnil gumb Refresh.

Bližnje naprave s prižganim bluetoothom smo našli s posebno komponento, ki jo ima vsak telefon z Bluetooth tehnologijo. Reče se ji Bluetooth adapter, ki je pomagala tudi pri vključevanju ter izključevanju bluetootha. Ta komponenta je prisotna v vseh telefonih z Bluetooth tehnologijo.

Naši nadaljnji postopki so bili rešeni s pomočjo interneta. Spoznavanje je zelo pomembnega pomena, saj je potrebno za povezovanje med napravami. Šele ko sta napravi povezani, si lahko pošiljata podatke med seboj.

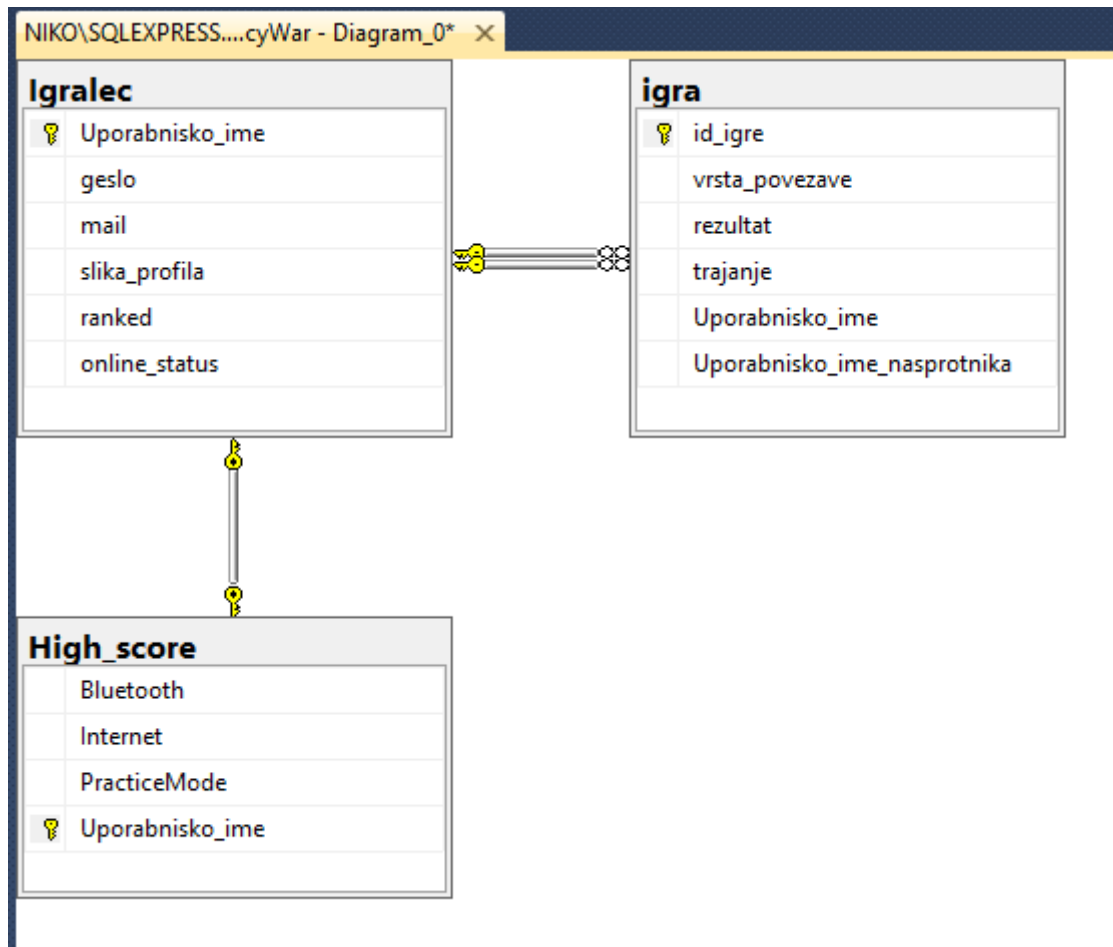
Ko so bile naprave najdene, je bil narejen že kar precej del. Dodali smo še kodo, ki je izpisala že vse seznanjene oziroma če rečemo v računalniškem žargonu "paired". S temi napravami potem ni bilo potrebno s kodo seznanjati naprav, ampak smo lahko šli kar na povezovanje oziroma angleško "connect".

Glede pošiljanja podatkov pa smo na uradni strani za razvijanje glede bluetootha našli kodo, ki sporočilo pretvori v bajte in ga nato pošlje. Nato je to druga naprava sprejela. Ta program je nato uporabil sošolec v igri BouncyWar.

3.3.2 Spletne storitve

Spletne storitve so storitve, ki so dostopne preko spleta. Z njihovo pomočjo bomo lahko vpisovali v podatkovno bazo ter pridobivali podatke iz nje. Med izdelovanjem spletnih storitev smo naleteli na mnogo težav. Že na samem začetku se je videlo, da nam primanjkuje znanja o tej temi. Najprej smo torej morali ugotoviti, kako spletne storitve delujejo v teoriji. Ko smo si to razjasnili, smo na internetu našli, kako deluje še v praksi. Med raziskovanjem tega smo zasledili, da obstaja več vrst spletnih storitev. Na primer asmx ter WCF spletna storitev, za katero smo ugotovili, da je kar precej težja od asmx, zato se zanjo raje nismo odločili. Asmx je spletna storitev, ki smo jo dodali v programu Visual studio. Tukaj smo pisali spletne metode, ki jih potem kliče administrator spletne strani ter izdelovalec igrice.

Prva stvar, ki smo jo naredili, je bila podatkovna baza. Že tukaj so se videli nekateri problemi, ki jih nismo predvideli, tako da smo morali bazo večkrat popravljati, ko bi že davno morala delati.



Slika 21: Podatkovna baza

Baza za našo igro ni pretirano kompleksna, vendar vseeno ni šlo brez problemov. Šli smo po načinu top-down (od zgoraj navzdol), saj smo si najprej določili entitete, ki jih bomo hranili. Nato smo počasi dodajali attribute k vsaki entiteti. Baza je nato dobila podobo, kot jo vidite na zgornji sliki. Zelo velik problem nam je predstavljala entiteta igra, kjer smo morali povezati dva igralca, ki sta igrala drug proti drugemu, saj smo morali združiti tuja ključa. Iz te slike se da razbrati še, da je za vsakega igralca shranjen še najboljši rezultat.

Ko je bila baza končana, smo se lotili pisanja poizvedb v programu Visual Studio. Uporabljali smo sintakso jezika C#, ki ga spoznavamo že 3 leta, zato ni bilo pretiranih problemov, pomagali pa smo si tudi z jezikom SQL, ki ga prav tako tudi že dobro obvladamo.

Poizvedba izgleda tako:

```
[webMethod]
0 references
public bool registrirajIgralca(string uporabniko_ime, string geslo, string mail, string potdoSlike, int ranked, int online_status)
{
    SqlCommand cmd2;
    SqlConnection con;

    con = new SqlConnection(@"Data Source=.\sqlexpress;Initial Catalog=BouncyWar;Integrated Security=True");
    string query = "Select geslo from Igralec where uporabniko_ime like '"+uporabniko_ime+"'";
    con.Open();
    cmd2 = new SqlCommand(query, con);

    string x = Convert.ToString(cmd2.ExecuteScalar());

    if (x==String.Empty)
    {
        SqlCommand cmd;
        cmd = new SqlCommand("INSERT INTO Igralec (Uporabniko_ime,geslo, mail, ranked, online_status) VALUES(@Uporabniko_ime,@geslo, @mail,@ranked,@online_status)", con);

        cmd.Parameters.AddWithValue("@Uporabniko_ime", uporabniko_ime);
        cmd.Parameters.AddWithValue("@geslo", geslo);
        cmd.Parameters.AddWithValue("@mail", mail);
        cmd.Parameters.AddWithValue("@ranked", ranked);
        cmd.Parameters.AddWithValue("@online_status", online_status);

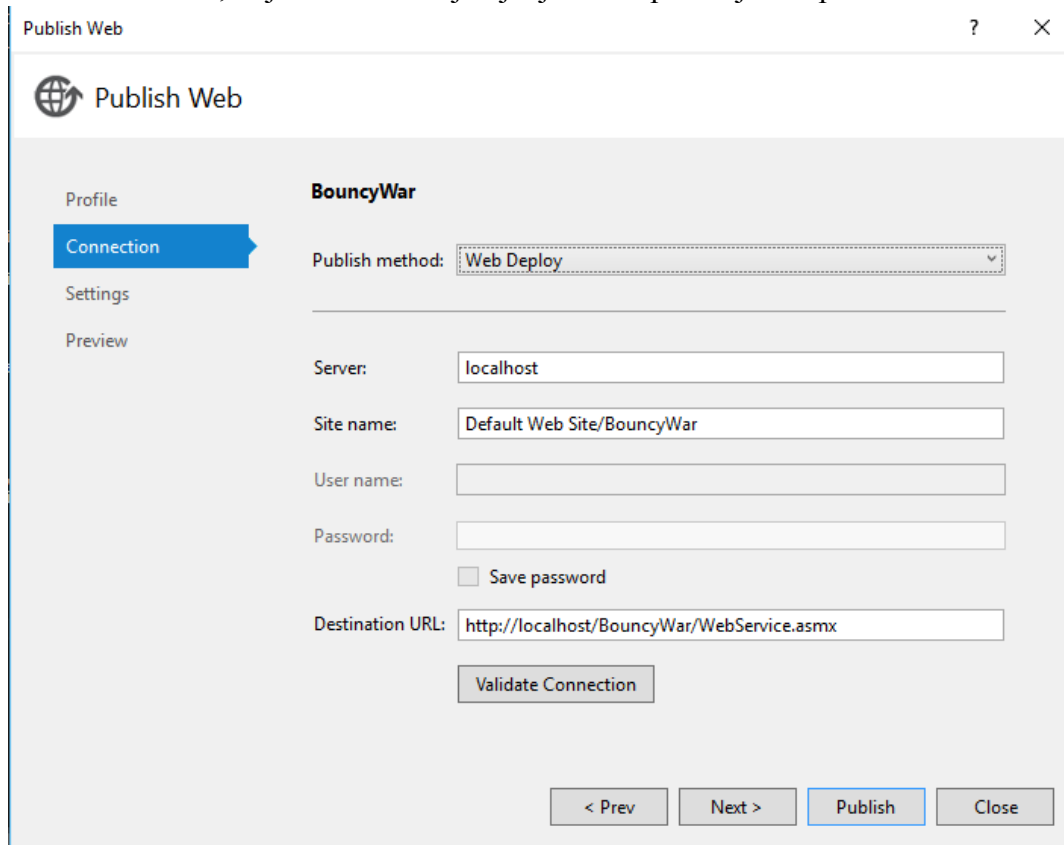
        cmd.ExecuteNonQuery();

        int krneki = 0;
        cmd = new SqlCommand("insert into high_score (@Bluetooth, @Internet, @PracticeMode,@Uporabniko_ime) values (@Bluetooth, @Internet, @PracticeMode, @Uporabniko_ime)", con);
        cmd.Parameters.AddWithValue("@Bluetooth", krneki);
        cmd.Parameters.AddWithValue("@Internet", krneki);
        cmd.Parameters.AddWithValue("@PracticeMode", krneki);

        cmd.ExecuteNonQuery();
        con.Close();
        return true;
    }
    else
    {
        con.Close();
        return false;
    }
} //Registrira uporabnika
```

Slika 22: Spletna metoda, ki vpiše podatke v tabelo Igralec ter High_score (Najboljši dosežek)

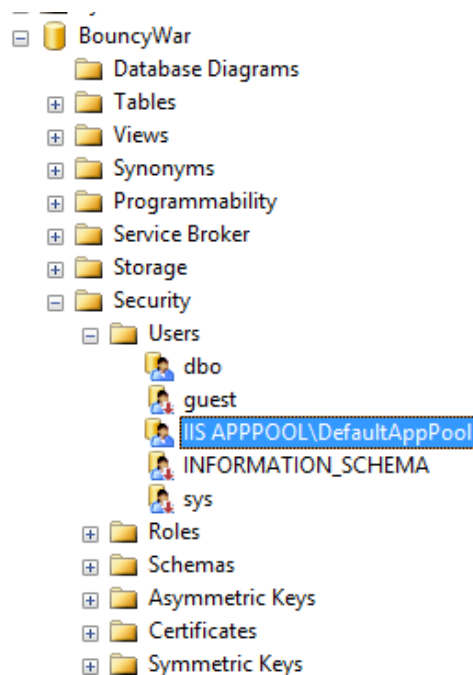
Napisali smo potrebne metode, nato pa smo se lotili še objavljanja tega. Najprej smo določili, kam želimo spraviti našo spletno aplikacijo. To smo storili v programu Visual studio, kjer smo naredili račun, ki je skrbel za objavljanje naše aplikacije na spletu.



Slika 23: Objavljanje spletne aplikacije

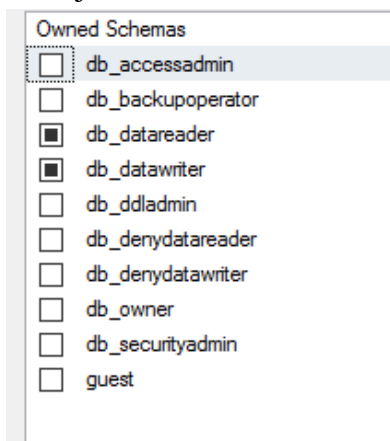
Tukaj je na vrsto prišel program IIS (Internet Information Services), ki je že od vsega začetka nameščen na operacijskem sistemu Windows. Ta program gosti našo spletno storitev. Za delovanje tega smo morali nastaviti »port forwarding« ali preusmeritev vrat na našem routerju, tako da lahko uporabnik od zunaj dostopa do našega lokalnega omrežja.

Ko smo prišli do tega koraka, so se pojavile nove težave. Bile so težave s podatkovno bazo, ki se obnaša drugače kot na lokalnem omrežju. Najprej smo dobili težave, da uporabnik IIS APPPOOL/DefaultAppPool nima dostopa do baze. Najprej nismo razumeli, kaj ta težava predstavlja, kasneje pa smo raziskali, da se vsak uporabnik, ki uporablja to spletno aplikacijo, predstavi kot uporabnik IIS APPPOOL/DefaultAppPool. To težavo smo odpravili tako, da smo naši bazi dodali uporabnika s točno takšnim imenom.



Slika 24: Uporabnik v bazi

Naslednji problem je bil, da ta uporabnik ni imel nastavljenih pravic, kaj lahko počne in česa ne sme (tako smo mu dovolili izvajati samo SELECT, UPDATE, INSERT INTO stavke).



Slika 25-Dodeljevanje pravic

Naš poslednji problem je predstavljalo to, da ta uporabnik ni imel določenih nalog oziroma funkcij oziroma tudi vlog na naši podatkovni bazi. Dali smo mu samo možnost vpisovanja ter branja iz baze, da ne bi prišlo do kakšnih težav v primeru, če bi se kdo izmed neželenih uporabnikov znašel v aplikaciji ter hotel izbrisati bazo.

4 Analiza rezultatov

Naredili smo raziskavo na temo igrice. Raziskavo smo naredili preko spleta. Postavili smo 9 vprašanj. Povratne informacije smo dobili od 35 ljudi, ki so v povprečju stari 18.9 let. Od tega je bilo 25 moških ter 10 žensk.

Vsi udeleženci:	35
Moški:	25
Ženske:	10
Povprečna starost:	18.9 leta

Slika 26: Osnovni podatki o anketirancih

Naše prvo vprašanje je bilo povezano z igranjem igrice nasploh. Bili smo kar presenečeni nad izidom, saj smo menili, da jih bo več izbralo odgovor "da".

1. Ali na vašem telefonu redno igrate igrice?			
da	19		57.6%
ne	14		42.4%
Vsi rezultati	33		

Slika 27: Vprašanje 1

Naslednje vprašanje je bilo takšnega tipa, da če je anketiranec na prvo vprašanje odgovoril z "da", mu ga je program zastavil, v nasprotnem primeru pa to vprašanje ni bilo zastavljeno. Želeli smo namreč izvedeti, koliko časa uporabniki v povprečju na dan porabijo za igranje igrice. Večina uporabnikov jih igra približno 30 minut oziroma manj, nekateri pa tudi več kot 1 uro.

2. Koliko časa dnevno porabite za igranje igrice?			
Manj kot 15 minut	6		31.6%
Od 15 do 30 minut	7		36.8%
Od 30 minut do 1 ure	3		15.8%
Od 1 ure do 2 uri	2		10.5%
Več kot 2 uri	1		5.3%
Vsi rezultati	19		

Slika 28: Vprašanje 2

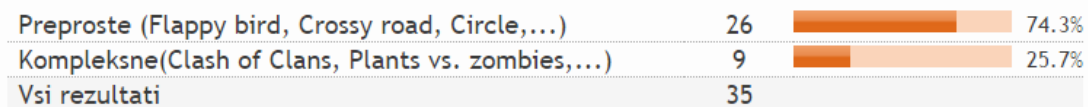
Nadaljevali smo z vprašanjem, s katerim smo želeli izvedeti, kakšne igre najraje igrajo anketiranci. Ugotovili smo, da prevladujejo miselne. Menimo, da je to zato, ker prisilijo igralca k razmišljanju. Takšne igrice so primerne za daljši čas. Miselnim igram sledijo zabavne, katerih namen je, da uporabniku predvsem krajšajo čas.



Slika 29: Vprašanje 3

Hipotezo, da naj bi igra bila preprosta in igralcu predstavljala očiten izziv, smo na podlagi četrtega in petega vprašanja potrdili. Na spodnjih dveh slikah je razvidno, da je večina ljudi takšnega mišljenja.

4. Katere igre so po vašem mnenju najbolj igrane?



5. Ali radi igrate mobilne igre, pri katerih tekmujete proti svojemu prijatelju?



Slika 30: Vprašanje 4 in 5

Z našim šestim vprašanjem smo želeli samo okvirno izvedeti, katere igre anketiranci najraje igrajo. Želeli smo izvedeti, ali bi bilo mogoče, če bi se čez čas pojavila igrice Bouncy War na tem seznamu.

6. Katera je vaša najljubša mobilna igra(napiši ime)?

- 1) matjažek je gujdeka klau hahaha:D
- 2) Besedovnjak
- 3) Dual
- 4) fun run
- 5) Fun run
- 6) Najbolj mi je v zadnjem času pri srcu fun run, ki jo lahko igram skupaj s prijatelji.
- 7) Fun run
- 9) Veliko iger je zanimivih in zabavnih za zapravljanje časa, to vprašanje je dvoumno, najljubša kot v smislu med šolo, ali najljubša kot igranjem doma? no kakorkoli nekaj fajnih: 8Ball Pool, 2 cars, Fruit Ninja
- 10) clash royale
- 11) Nimam najljubše.
- 12) Quiz up
- 13) Piano tiles 2

Slika 31: Vprašanje 6

Pod vprašanjem številka sedem smo izvedeli, zakaj so si uporabniki izbrali te igre. V večini primerov vidimo, da jim je ta igra všeč, ker lahko igrajo s prijateljem oziroma jim predstavlja nek izziv. Tukaj lahko potrdimo našo hipotezo, da ljudje radi igrajo igre 1 proti 1 ter si upamo trditi, da takšnih iger nikoli ne bo preveč.

7. Napišite razloge, zaradi katerih vam je ta igra najljubša.

- 1) kolar je lep:3
- 2) lizziv
- 3) Ker uživam ob tem, ko premagujem prijatelje.
- 4) tekmuješ s prijatelji
- 5) Igranje s prijatelji
- 6) Igra mi je všeč predvsem zaradi njene preprostosti, ter lahko jo igram skupaj s prijatelji, ki igro še bolj popestri.
- 7) Lahko jo igram istočasno s prijatelji.
- 8) ker klikaš stvari po ekrani pa je epic stari ful
- 9) Ker je enostavna in zabavna
- 10) kompleksna , miselna
- 11) /
- 12) V kvizu lahko s svojim znanjem tekmuješ s prijatelji ali neznanci po svetu. Igra je v angleščini in mi omogoča dodatno pridobivanje znanja angleščine. Vprašanja pri kvizu so iz vseh področij (general knowledge, chemistry, memes, computer logos, math,...)
- 13) všeč mi je glasba
- 14) nuoro
- 15) Tak dobra je
- 16) Zanimiva, igranje proti resničnim osebam
- 17) Podobna je drugim popularnim mobilnim igram s tem, da je ne moraš igrati dolgo časa, saj so postavljene omejitve npr. gradnja struktur... Ampak dovolj drugačna in zabavna, da se je ne naveličam. Kot ostale popularne igre je tudi socialni aspekt te igre privlačen saj lahko igraš s prijatelji oz. proti njim, in napadaš igralce, ki jih ne poznaš, poleg tega pa se lahko pridružiš clanu s katerim nato sodeluješ in se zabavaš.
- 18) Plants vs zombies je bil že zelo zanimiv kot igra na računalniku. Ker se je pozneje portala na android in ios jo lahko igram na vlaku/avtobusu in bilo kje drugje, kar prvotno nisem mogel.

Slika 32: Vprašanje 7

Z 8 in 9 vprašanjem smo hoteli dobiti vtis, kaj ljudje menijo o igrah, ki za svoje delovanje potrebujejo kakšno dodatno stvar, kot sta internet in bluetooth. Menimo, da bosta slikovna prikaza rahlo drugačna, predvsem drugi, potem ko bodo poizkusili igro Bouncy War.



Slika 33: Vprašanje 8 in 9

5 Zaključek

Ko smo končali naš, na videz mali projekt, in se ozrli nazaj, smo se šele zavedli, koliko znanja in truda smo vložili v to raziskovalno nalogo, da smo jo uspešno izpeljali. Z veseljem smo lahko rekli, da je to naše delo.

Klasična igra Pong je bila svoj čas uspešnica zaradi svoje preprosti in ker trg takrat še ni ponujal tako širokega spektra iger kot danes. Pravzaprav igra Pong velja za prvo športno arkadno igro, zato je bil njen uspeh tako rekoč že zagotovljen. Po vseh teh letih, ko je razvijanje računalniških iger že toliko napredovalo in se je grafična podoba iger nešteto krat izboljšala, uporabniki pametnih telefonov še vedno raje posegajo po preprostejših igrah. Tako smo hipotezo: »Android igra naj bi bila preprosta in naj bi igralcu predstavljala očiten izziv.« odločno potrdili.

Hipotezo: »Na Android napravah manjka iger, kjer igrata med seboj igralca 1 proti 1.« smo potrdili, saj smo ugotovili, da v mobilnih trgovinah, kot sta Google Play in App Store primanjkuje tovrstnih iger, uporabniki pa bi bili več kot zadovoljni, če bi na trg prišla kakšna igra več s tovrstno tematiko.

Ne nazadnje pa smo pod drobnogled vzeli hipotezo: »Za dokončanje projekta zadostuje osnovno srednješolsko znanje«, ki smo jo takoj po dokončanju naše raziskovalne naloge ovrgli. Zadali smo si precej težko nalogo, ki je zahtevala, da smo se hočeš nočeš naučili veliko novega, če smo želeli, da bi nam stvar uspela. Poglavitni vir znanja smo črpali iz naših profesorjev, ki so nam z veseljem pomagali, če pa smo naleteli na kakšno težavo doma, smo si pomagali z internetom.

Med izdelavo raziskovalne naloge smo nadgradili znanje vrsto programerskih jezikov, se naučili postaviti spletni strežnik, uporabljati spletne storitve in najpomembnejše, naučili smo se ekipnega dela. Verjamemo, da nam bo vso pridobljeno znanje v veliko pomoč pri nadaljnjem izobraževanju, zato te izkušnje ne bi zamenjali za nič na svetu.

6 Zahvala

Zahvaljujemo se našemu mentorju mag. Boštjanu Resinoviču, ki nam je z razlagami pomagal pri razrešitvi problemov, ki so nas pestili tekom dela.

Zahvaljujemo se tudi profesorju Dušanu Fugini, ki nam je večkrat priskočil na pomoč ter si vzel čas za nas.

7 Literatura

Tom Dykstra, 2013, ASP.NET Web Deployment using Visual Studio: Deploying to Test. [online] (23.3.2015). Dostopno na: <http://www.asp.net/mvc/overview/deployment/visual-studio-web-deployment/deploying-to-iis>, 16.2.2016.

ProgrammingKnowledge, 2013, ASP.NET Tutorial 1- Introduction and Creating Your First ASP.NET Web Site [online] (30.7.2014). Dostopno na: https://www.youtube.com/playlist?list=PLS1QulWo1RIaM8-S7kTHgWd_pGNu-CyQS, 20.12.2015.

Abhijit Jana, 2009, Exploring Session in ASP.NET [online] (25.1.2009). Dostopno na: <http://www.codeproject.com/Articles/32545/Exploring-Session-in-ASP-Net#12>, 15.1.2016.

Mudassar Khan, 2012, TextBox: Minimum and Maximum Character Length Validation using ASP.Net RegularExpression Validators [online] (unknown). Dostopno na: <http://www.aspsnippets.com/Articles/TextBox-Minimum-and-Maximum-Character-Length-Validation-using-ASPNet-RegularExpression-Validators.aspx>, 23.1.2016.

Unknown, 1999, W3Schools online Web Tutorials [online] (unknown). Dostopno na: <http://www.w3schools.com/>, 20.12.2015.

Profgustin, 2012, Android - Discovering Bluetooth Devices [online] (unknown). Dostopno na: <https://www.youtube.com/watch?v=X-Z4TiJTtY>, 25.11.2015.

Stack Overflow, 2013, How to get the bluetooth devices as a list [online] (13.8.2015). Dostopno na: <http://stackoverflow.com/questions/10795424/how-to-get-the-bluetooth-devices-as-a-list>, 25.11.2015.

Yuval Joseph, 2012, Android Bluetooth tutorial [online] (5.12.2013). Dostopno na: <https://www.youtube.com/playlist?list=PLQrQKDQmvSfxEmYOugNkYLSEs5oLxs5u6>, 26.11.2015.

Stack Overflow, 2014, How to pair Bluetooth device programmatically Android [online] (6.3.2016). Dostopno na: <http://stackoverflow.com/questions/17168263/how-to-pair-bluetooth-device-programmatically-android>, 27.11.2015.

Developers, unknown, Bluetooth [online] (unknown). Dostopno na: <http://developer.android.com/intl/ja/guide/topics/connectivity/bluetooth.html>, 18.11.2015.

Unity, unknown, Unity Manual [online] (unknown). Dostopno na: <http://docs.unity3d.com/Manual/index.html>, 14.12.2015.

Unity, unknown, Unity Manual [online] (unknown). Dostopno na: <http://docs.unity3d.com/Manual/index.html>, 14.12.2015.

Unity, unknown, Learn with Unity [online] (unknown). Dostopno na: <http://unity3d.com/learn>, 10.11.2015.

IZJAVA*

Mentor (-ica) BOSTJAN RESINČIČ, v skladu z 2. in 17. členom Pravilnika raziskovalne dejavnosti »Mladi za Celje« Mestne občine Celje, zagotavljam, da je v raziskovalni nalogi naslovom

BOUNCY WAR

katere avtorji (-ice) so PATRICK JUREČIČ, MIKO KOLAR, TIJA BREZOVNIK

- besedilo v tiskani in elektronski obliki istovetno,
- pri raziskovanju uporabljeno gradivo navedeno v seznamu uporabljene literature,
- da je za objavo fotografij v nalogi pridobljeno avtorjevo (-ičino) dovoljenje in je hranjeno v šolskem arhivu,
- da sme Osrednja knjižnica Celje objaviti raziskovalno nalogo v polnem besedilu na knjižničnih portalih z navedbo, da je raziskovalna naloga nastala v okviru projekta Mladi za Celje,
- da je raziskovalno nalogo dovoljeno uporabiti za izobraževalne in raziskovalne namene s povzemanjem misli, idej, konceptov oziroma besedil iz naloge ob upoštevanju avtorstva in korektnem citiranju,
- da smo seznanjeni z razpisni pogoji projekta Mladi za Celje

Celje, 9. 3. 16



žig šole

Podpis mentorja(-ice)

Podpis odgovorne osebe

*

POJASNILO

V skladu z 2. in 17. členom Pravilnika raziskovalne dejavnosti »Mladi za Celje« Mestne občine Celje je potrebno podpisano izjavo mentorja(-ice) in odgovorne osebe šole vključiti v izvod za knjižnico, dovoljenje za objavo avtorja(-ice) fotografskega gradiva, katerega ni avtor(-ica) raziskovalne naloge, pa hrani šola v svojem arhivu.