Sveučilište Jurja Dobrile u Puli

Fakultet informatike u Puli

**TOMISLAV IVANDA**

**RAZVOJ RAČUNALNE IGRE U OKRUŽENJU COCOS2D-X**

Završni rad

Pula, rujan, 2019.

Sveučilište Jurja Dobrile u Puli

Fakultet informatike u Puli

**TOMISLAV IVANDA**

**RAZVOJ RAČUNALNE IGRE U OKRUŽENJU COCOS2D-X**

Završni rad

**JMBAG: 0303069498, redoviti student**

**Studijski smjer: Informatika**

**Predmet: Programsko inženjerstvo**

**Znanstveno područje: Društvene znanosti**

**Znanstveno polje: Informacijske i komunikacijske znanosti**

**Znanstvena grana: Informacijski sustavi i informatologija**

**Mentor: doc. dr. sc. Tihomir Orehovački**

Pula, rujan, 2019.



**IZJAVA O AKADEMSKOJ ČESTITOSTI**

Ja, dolje potpisani \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_, kandidat za prvostupnika \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ovime izjavljujem da je ovaj Završni rad rezultat isključivo mojega vlastitog rada, da se temelji na mojim istraživanjima te da se oslanja na objavljenu literaturu kao što to pokazuju korištene bilješke i bibliografija. Izjavljujem da niti jedan dio Završnog rada nije napisan na nedozvoljen način, odnosno da je prepisan iz kojega necitiranog rada, te da ikoji dio rada krši bilo čija autorska prava. Izjavljujem, također, da nijedan dio rada nije iskorišten za koji drugi rad pri bilo kojoj drugoj visokoškolskoj, znanstvenoj ili radnoj ustanovi.

Student

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

U Puli, \_\_\_\_\_\_\_\_\_, \_\_\_\_\_\_\_\_ godine



**IZJAVA**

**o korištenju autorskog djela**

Ja, \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ dajem odobrenje Sveučilištu Jurja Dobrile u Puli, kao nositelju prava iskorištavanja, da moj završni rad pod nazivom \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ koristi na način da gore navedeno autorsko djelo, kao cjeloviti tekst trajno objavi u javnoj internetskoj bazi Sveučilišne knjižnice Sveučilišta Jurja Dobrile u Puli te kopira u javnu internetsku bazu završnih radova Nacionalne i sveučilišne knjižnice (stavljanje na raspolaganje javnosti), sve u skladu s Zakonom o autorskom pravu i drugim srodnim pravima i dobrom akademskom praksom, a radi promicanja otvorenoga, slobodnoga pristupa znanstvenim informacijama.

Za korištenje autorskog djela na gore navedeni način ne potražujem naknadu.

U Puli, \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (datum)

Potpis

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**SAŽETAK**

Igra GokuFly vrača igrače u vremena kada je igrica „Flappy Birds“ bila aktualna. Igrica se sastoji od lika kojim upravlja igrač stiskanjem tipke enter i cijevi koje se nasumično pojavljuju nebi li svojim zahtjevnim pozicijama što prije omele lika i tako mu nanijele poraz. Cilj ove igre je pokušati proći kroz što više cijevi ne dirajući ih i tako postavljati osobne rekorde. Kretanje samog lika nije kao u običajnoj prvobitnoj igrici da ga gravitacija vuče prema dolje nego je ona izmijenjena i vuče ga prema gore što još više otežava samu igru

**KLJUČNE RIJEČI:** *Cocos2dx, GokuFly, 2d igra, Notepad++,Html5, JS*

ABSTRACT

GokuFly brings players back to the time when Flappy Birds was current. The game consists of a character controlled by the player by pressing the enter key and pipes that appear randomly in order to distract the character as quickly as possible and cause him defeat. The goal of this game is to try to get through as many tubes as possible without touching them and thus setting personal records. The movement of the character itself is not like in the original original game that gravity pulls it down but it is altered and pulls it up which further complicates the game itself

**KEYWORDS:** *Cocos2dx***,** *GokuFly, 2d game , Notepad++, Html5, JS*

**SADRŽAJ**

[1.UVOD 7](#_Toc19381943)

[2. PREZENTACIJA IGRICE 8](#_Toc19381944)

[2.1 POČETNI ZASLON SA UPUTAMA 8](#_Toc19381945)

[2.2 POČETAK IGRICE 9](#_Toc19381946)

[2.3 IGRICA U TIJEKU 10](#_Toc19381947)

[2.4 KRAJ IGRICE 11](#_Toc19381948)

[3. PRIKAZ STVARANJA IGRICE 12](#_Toc19381949)

[3.1 DODAVANJE I NAMJEŠTANJE POZADINE 12](#_Toc19381950)

[3.2 DETEKCIJA DODIRA NA EKRANU 13](#_Toc19381951)

[3.3 DODAVANJE LIKA 15](#_Toc19381952)

[3.4 DODAVANJE FUNKCIJE SKAKANJA 16](#_Toc19381953)

[3.5 DODAVANJE KRETANJA SPOREDNIH ELEMENATA 17](#_Toc19381954)

[3.6 DODAVANJE TUBA I NJIHOVOG NAČINA RADA 19](#_Toc19381955)

[3.7 STVARANJE TUBI NA EKRANU 21](#_Toc19381956)

[3.8 DODAVANJE COLLISIONA 22](#_Toc19381957)

[3.9 DODAVANJE TEKSTA 23](#_Toc19381958)

[4. ZAKLJUČAK 24](#_Toc19381959)

[5. LITERATURA 25](#_Toc19381960)

[6. POPIS SLIKA 26](#_Toc19381961)

# 1.UVOD

GokuFly je igrica napravljena po uzoru na prijašnju igricu „Flappy Birds. Ovdje se radi o 2d igrici koja se sastoji od jednog lika kojemu je ime Goku koji je ime dobio po liku iz crtane serije Zmajeva kugla i koji ima cilj proći kroz što više tubi da ih pritom ne dira. Tube se stvaraju i nikad se ne zna gdje će se stvoriti što daje samoj igri dodatnu čar i napetost .Sama igrica zahtjeva maksimalnu koncentriranost igrača da što bolje procjeni padanje lika i vrijeme kada mora stisnuti tipku za novi skok .

Ono što daje drugu dimenziju i novitet igrici je to što lik ne skače prema gore i gravitacija ga ne vuče prema gore , nego ga gravitacija vuče prema dolje i tako pritiskom tipke on pokušava ostat što bliže zemlji.

Za stvaranje ove aplikacije korišteno je razvojno okruženje Cocos2d-x.Ovo razvojno okruženje pruža više načina za stvaranje igrica kao što je Cocos Creator i HTML5Lite. Cocos2d-x je 2D platforma za razvoj mobilnih igara sa otvorenom i cross platformom otvorenog koda, poznat po brzini, stabilnosti i lakoći uporabe .

# 2. PREZENTACIJA IGRICE

## 2.1 POČETNI ZASLON SA UPUTAMA

Početni zaslon sastoji se od imena same igrice i opisa što igrač mora raditi da bi ostao što duže na životu. Ispod toga nalazi se naputak da pritiskom bilo gdje na ekranu igrač pokreče samu igricu .

****

*Slika 1. Početni zaslon sa uputama*

## 2.2 POČETAK IGRICE

Na ovoj slici vidimo lika te okruženje u kojem se on nalazi . Tube ne vidimo jer se one ne pojavljuju odmah . Uz lika vidimo još stabla i oblake koji su u stalnoj kretnji kako bi dali doživljaj stvarne kretnje lika . U gornjem lijevom kutu vide se dva broja . Prvi gornji broj označava trenutni rezultat koji je ostvario lik , a broj ispod njega označava najbolji postignuti rezultat koji je ostvario sami lik.

**

*Slika 2. Početak igrice*

## 2.3 IGRICA U TIJEKU

Kada je igrica u tijeku pojavljuju nam se tube koje se nasumično pojavljuju te nikad se ne zna kako će i gleda biti sljedeća postavljena. Gore u kutu vidimo kako su se brojevi promijenili i kako prvi broj označava da je lik prošao do sad tri tube , a da mu je dosadašnji skor bio dva .



*Slika 3. Igrica u tijeku*

## 2.4 KRAJ IGRICE

Na samome kraju igrice kada igrač dodirne tubu ili nebo igra prestaje. Pred igračem se pojavljuje natpis na ekranu da je igra gotova , a ispod toga stoji natpis vezan uz kreatora igrice ,njegovog mentora i ustanovu. U gornjem lijevom kutu se vidi da je trenutni rezultat izbrisan pošto je igra stala , a da se novi najbolji rezultat upisao dolje. Na slici se vidi zadnja pozicija lika prije nego što je igrač pogriješio i ta scena traje 4 sekunde dok se ponovno ne učita početni ekran gdje igrač može ponovno započeti igru.

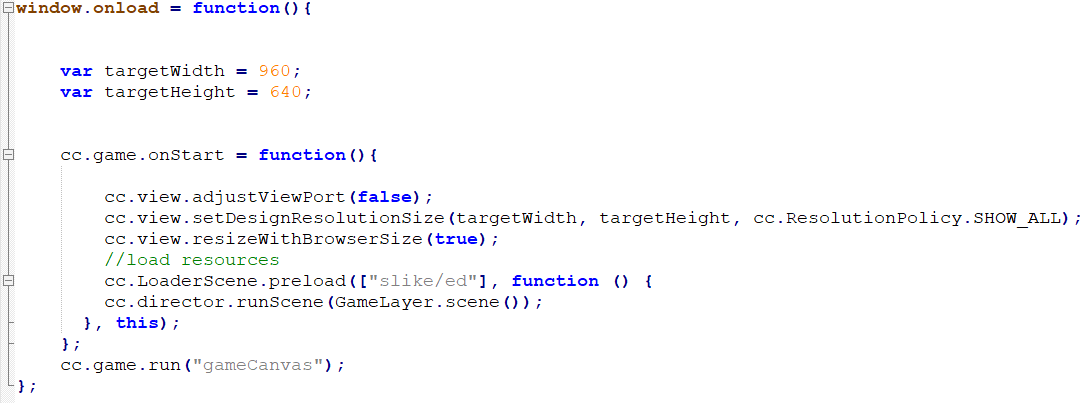


*Slika 4. Kraj igrice i prikaz credita*

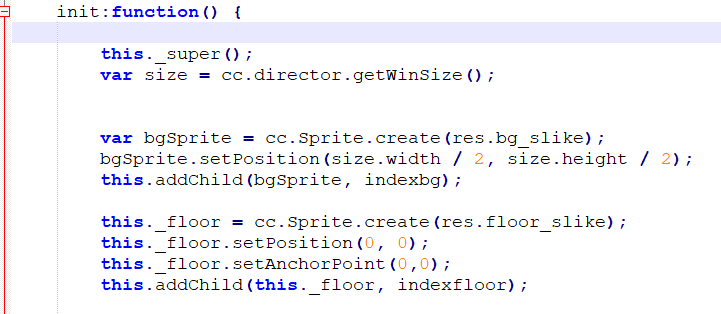
## 3. PRIKAZ STVARANJA IGRICE

## 3.1 DODAVANJE I NAMJEŠTANJE POZADINE

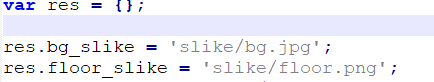
Prva stvar koja je bitna za napravit je učiniti da se ekran igrice može razvlačiti te da se tim razvlačenjem ne gubi slika nego da se automatski i proporcionalno smanjuje/povečava , ovdje su dodane i dimenzije samog ekrana , kao primjer koda skaliranja poslužila je igrica „Moon Warrior“ .



*Slika 5. Mogučnost skaliranja*

**

*Slika 6. Game.js kod za dodavanje pozadine*

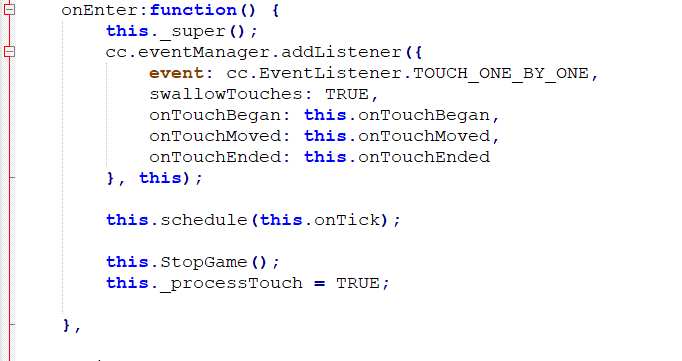
**

*Slika 7. gamemanager.js kod za prikazivanje puta*

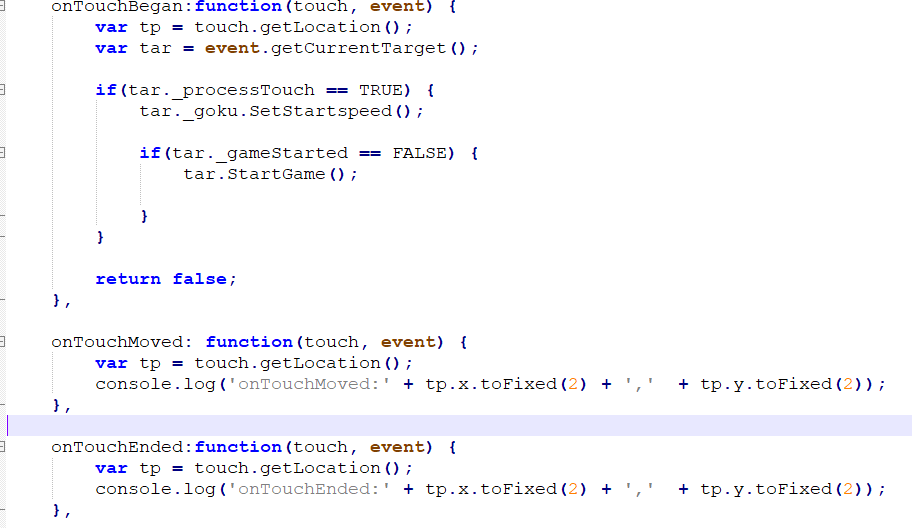
Slike šest i sedam prikazuju dodavanje pozadinske slike , poda i načina na koji se slike vade iz datoteka . Indeks komponenti(slika) označava kako će koja komponena stajati, hoće li pozadina biti vidljiva ispred poda ili obrnuto , a taj indeks je dodan isto u gamemanager.js pod nazivima indeksfloor, indeksbg. Naravno svaki taj indeks sadrži svoj broj. Još jedna razlika u podu i pozadinskoj slici je što je podu dodano this. umjesto var jer će se poslije morati opet pozvati floor.

## 3.2 DETEKCIJA DODIRA NA EKRANU

Ovdje vidimo sličnu sintaksu kao u C++ za dodavanje detekcije dodira. Potrebno je bilo dodati Listener koji će sadržavati argumente koji će na kraju slušati sami Listener. Dodavanje dodira jedan po jedan kako bi detektirali pritisak miša . Uz sve to vidimo kod koji sadrži funkcije kada je kraj,početak ili za samo vrijeme dodira koji javlja kada provjeravamo kod u pregledniku vrijeme i poziciju pritiska miša. Na samom kraju „On Touch Began „ korištena je negacija jer program ne treba javiti lokaciju miša kada je miš stisnut i kada se kreče nego samo kada korisnik stisne bez zadrške.



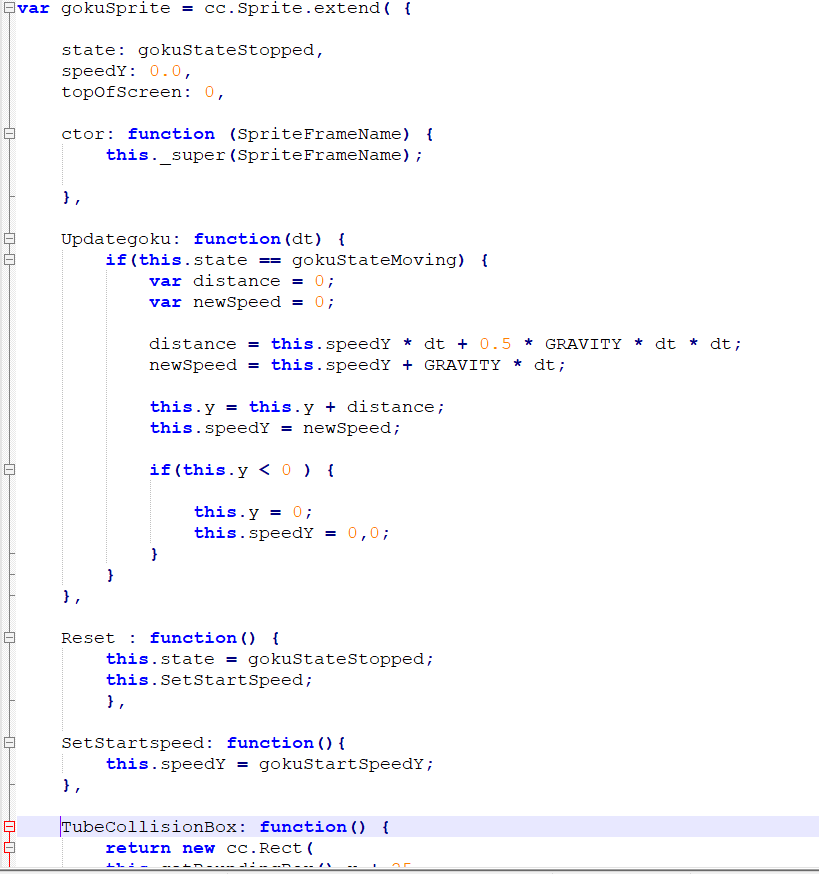
*Slika 8. Prikaz koda za detekciju pokreta*



*Slika 9. Prikaz koda u game.js-u*

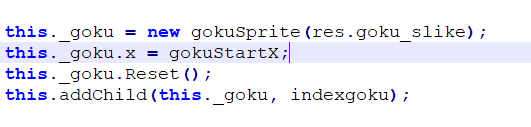
## 3.3 DODAVANJE LIKA

Slika broj 9 prikazuje skriptu „Goku.js“ koja je kreirana za dodavanje nekih funkcija i kreiranje samog lika . Skripta se sastoji od Update , Reset, Set Start Speed i Tube Collision Box . Svaka od njih ima svoju ulogu. Update funkciji ima dt(delta time ) što prikazuje vrijeme od zadnjeg updata. Reset funkcija vrača lika na početnu poziciju , a set start speed vuče podatak od brzini lika iz gamemanager.js . TubeCollisionBox predstavlja mjere koje lik ima a nisu vidljive, a korigirane su tako da odgovaraju propozicijama lika da kad rub lika dotakne tubu igrica odmah detektira dodir.



*Slika 10. Prikaz koda za dodavanje lika*

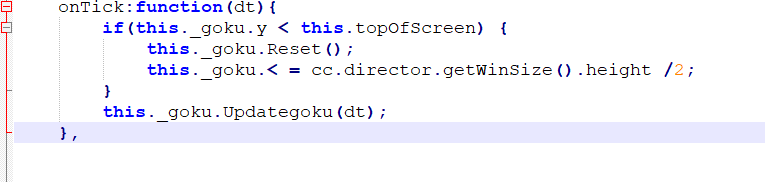
Na slici vidimo kod koji se ispisuje za stvaranje lika u game.js . Postupak je sličan kao i za dodavanje pozadine i poda. Poziva se slika lika iz res-a , poziva se reset funkcija , dodaje mu se indeks kako bi bio ispred svih likova u igrici.

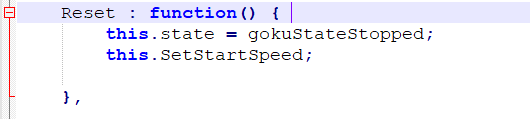


*Slika 11. Prikaz kreiranja lika u game.js*

## 3.4 DODAVANJE FUNKCIJE SKAKANJA

Na slici broj 11 vidimo funckiju kojom se definira mogučnost skakanja lika u igrici , kada lik dirne strop igrica staje i lik se restartira na početnu poziciju iz koje je krenuo.



**

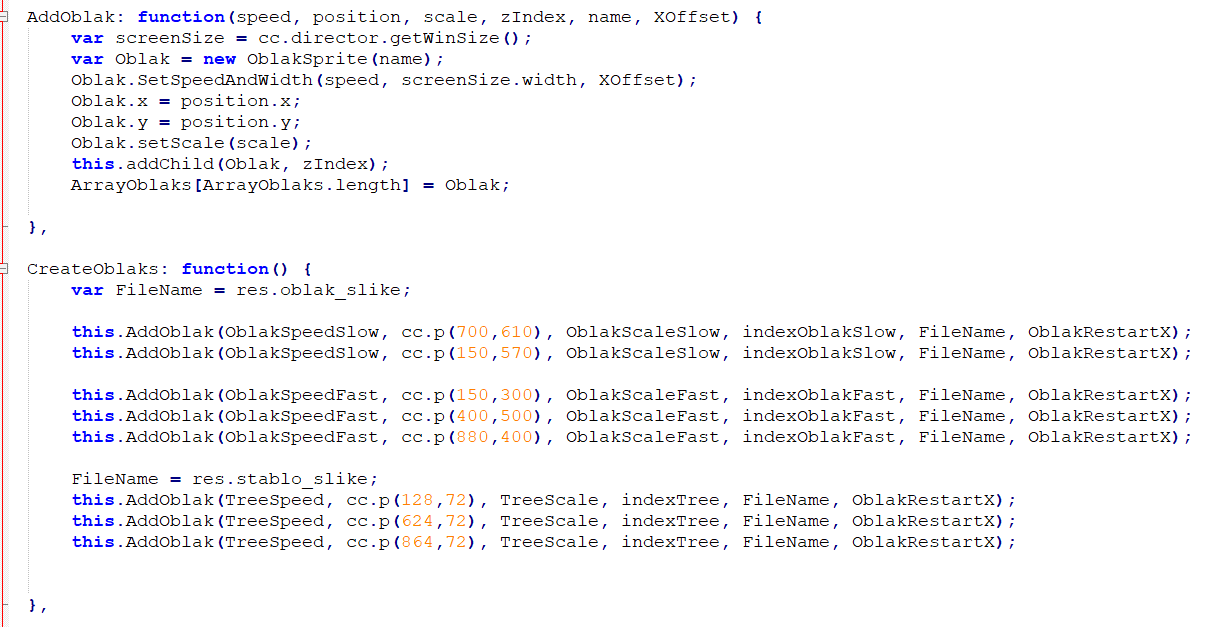
*Slika 12. Prikaz funkcij namjenjen skakanju*

## 3.5 DODAVANJE KRETANJA SPOREDNIH ELEMENATA

U ovom dijelu igrice dodan je oblak.js , napravljen je po uzoru na goku.js zato i sadrži slične varijable . Svi dijelove oblaka i tree-a mogu se svrstati pod AddOblak jer će se svi kretati identično s desna na lijevo pa ih je zato bolje grupirat da se ne radi više nepotrebnih funkcija . Oblak.js sadrži funkcije stop, start, reached destination . Kod označava distance koju oblak mora preći kretanjem sa desne na lijevu stranu i kada dođe do kraja tada xoffset prelazi u negativno I onda je željena destinacija izvršena , nakon toga restira se opet na desnu stanu. U gamemanageru je sadržan indexi, brzina i mjesto odakle će se vaditi slike . Ovdje prikazani OblakSlowSpeed I OblakFastSpeed predstaljaju velike I male oblake te njihovu brzinu .



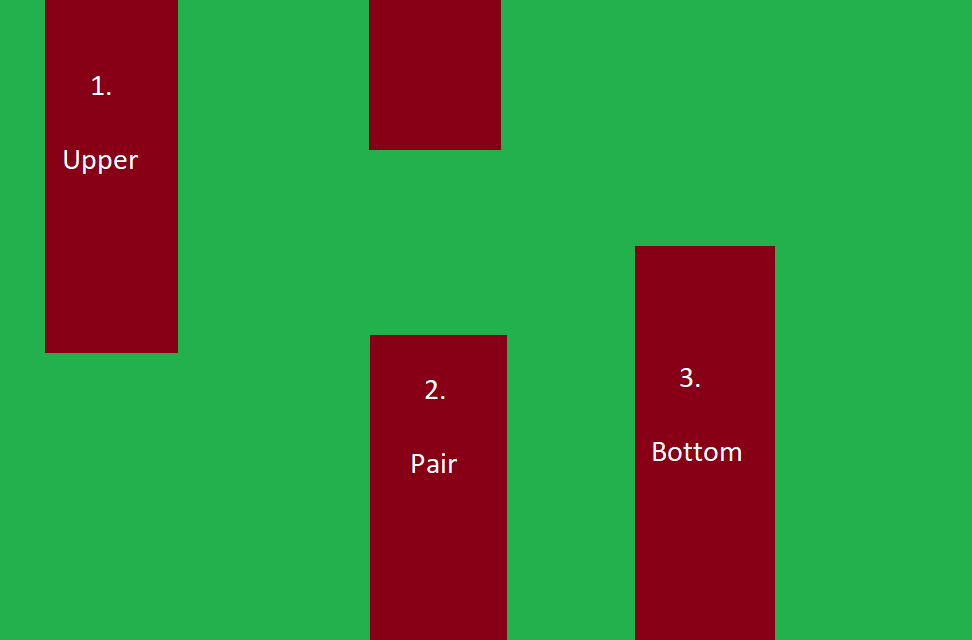
*Slika 13. Funckije u oblak.js*



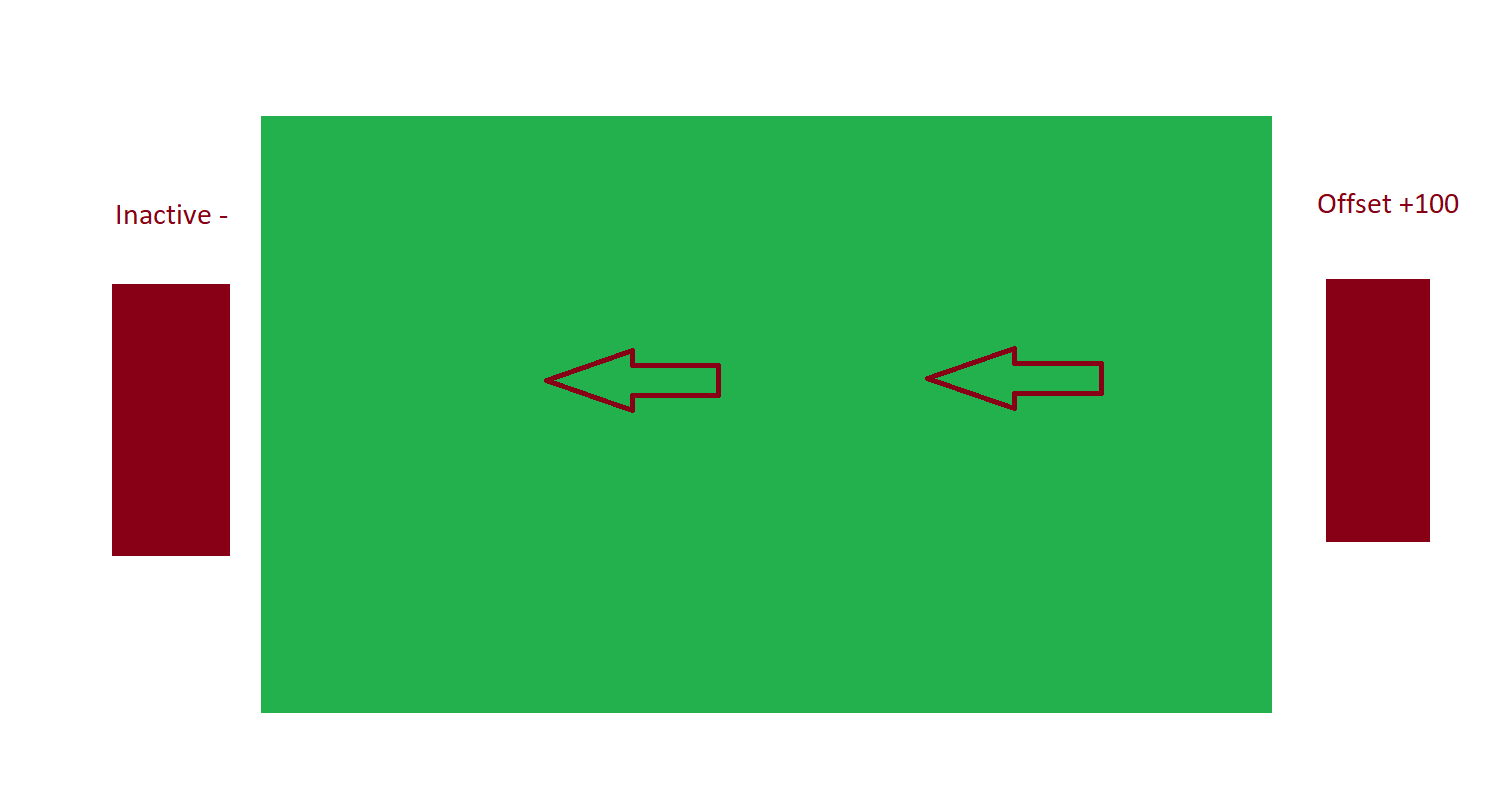
*Slika 14. Funckije vezane za kretanje u game.js*

## 3.6 DODAVANJE TUBA I NJIHOVOG NAČINA RADA

Postoje 3 para tubi u igrici : pair , upper , lower. Da se izbjegne preveliko odmicanje tubi u ekran definiran je SingleGapTop i SingleGapBottom koji definira kretanje tubi u novom stvaranju , ali sa dređenom granicom I to vrijedi za sve . Offset služi za kontrolu micanja tubi , kada tuba dode do kraja ekrana offset prelazi u inactive stanje tj. prelazi u minus.

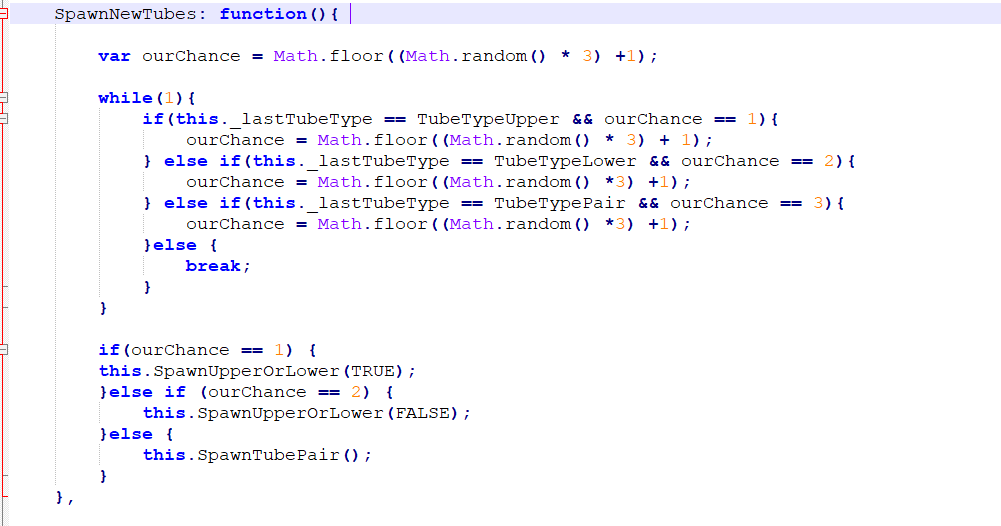


*Slika 15. Vrste tuba*



*Slika 16. Prikaz kako je zamišljen Offset*

Slika broj 17 predstavlja jednu od 3 funkcije za stvaranje tubi. Ovdje se bira koja će se vrsta tube prikazati , a to se određuje nardnom if . Problem je ovdje što igrica neće znat koji je zadnji oblik tube bio prikazan u igrc zato se dodaje while za projeruju .



*Slika 17. Dio koda za Stvaranje Tubi*

## 3.7 STVARANJE TUBI NA EKRANU

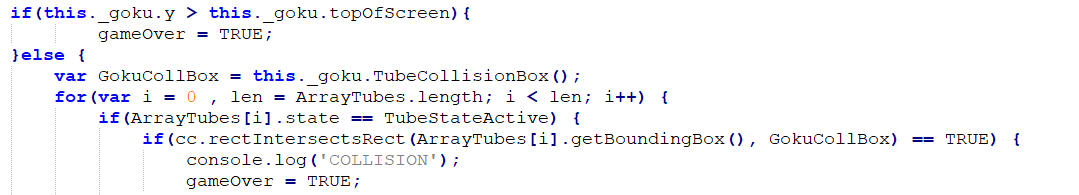
Za stvaranje stvarnog prikaza tubi bilo je potrebno dodati novu js. skriptu koja je bila nazvana Tube.js . Po svom izgledu jako je slična Oblak.js , ali se naravno razlikuje po nekim funkcijama . Slika tube dodana je u game.js u funkciji GetNextTube te nije kao u prijašnjim stvaranjima dodana u init funkciju . GetStart funkcija sadrži stopiranje akcija , kalukira se distanca I vrijeme koje treba proći da dođe di korištenja offseta I onda se stvara vidljiva tuba i action move pokreće kretanje tuba



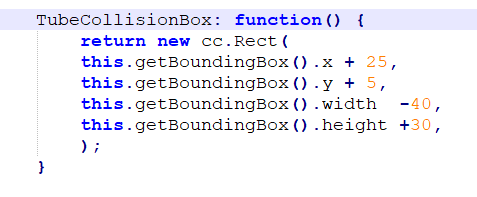
Slika 18. Funkcije u Tube.js

## 3.8 DODAVANJE COLLISIONA

U funckiji OnTick dodana je od prije if funckija i ako lik dodirne pod igrica je završena , sada je tu još dodan else koji za svako diranje tube prekida igru. Na slici 20 vidimo boundingbox koji predstavlja mjere koješ lik ima a nisu vidljive, I korigirane su tako da odgovaraju propozicijama lika da kad rub lika dotakne tubu igrica odmah detektira collision.



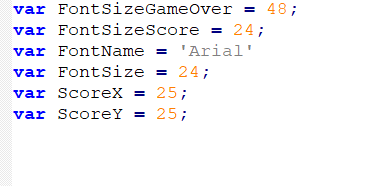
*Slika 19. Dodavanje else u onTick funkciju*



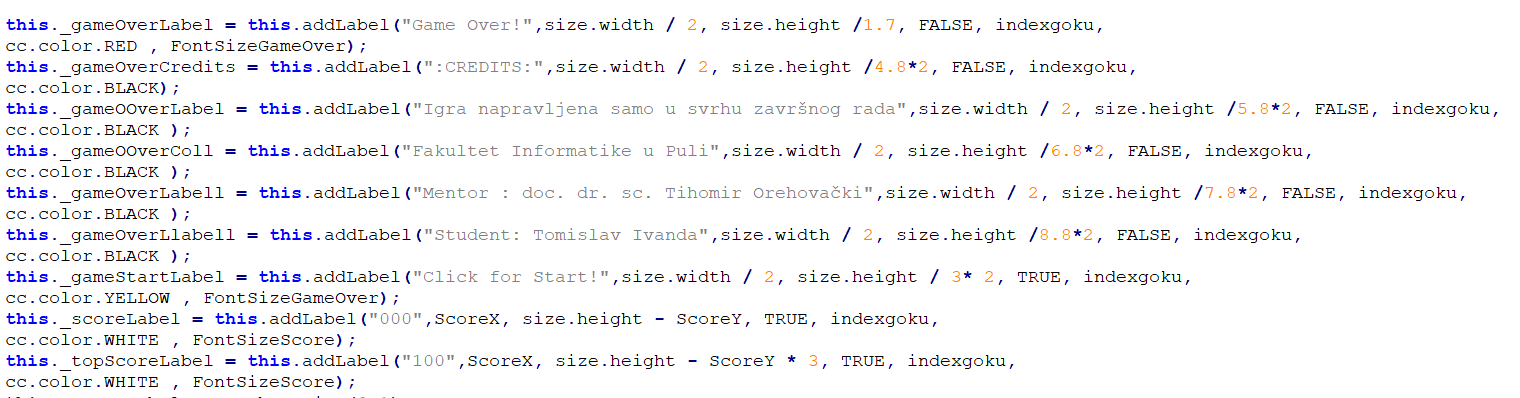
*Slika 20. Korigiranje mjera za veličinu našeg lika*

## 3.9 DODAVANJE TEKSTA

Na slici 21 definirane su vrijable u gamemanager.js sa svojim značenjima i vrijednostima te su poslje kao takve pozvane u game.js gdje su bile potrebne . Za prikazivanje/ skrivanje teskta koristilo se “.visible = true/false.” . Ovdje na slici 21 namještao se tekst na pozadini I dodavala boja



*Slika 21. Namještanje varijabli u gamemanager.js koje poslje zovemo u game.js*



*Slika 22. Postavljanje teksta za gameplay*

# 4. ZAKLJUČAK

Današnjica u kojoj živimo se jako brzo razvija kao i sva tehnologija oko nas.   
Ljudi ne mogu više zamisliti odlazak na kavu ,a da ne koriste svoj mobilni telefon . Možemo reči da je doba kompjuterskih igrica prošlo, ali ne i web igrica . Većina mladih ljudi više nema naviku skidati nove igrice koje dođu , nego svoje vrijeme bolje vole provesti na igricama koje im se nude na internetu i oko kojih nema posla sa kompliciranim instalacijama .

Tako sam i ja kao malo dijete na nastavi informatike uživao u zadnjih pola sata koje bi nam professor ostavio za uživanje u igrama i uvijek sam se pitao tko radi ovake igrice i kakvi to mozgovi moraju biti . Sada kada sam završio ovaj rad i zavirio u svijet programiranja igara samo želim nastaviti i upoznavati nova razvojna okruženja .

Moj izbor za pravljenje ovako lagane stare igrice je bio taj što kada je igrica došla nako par tjedana više nije bila dostupna na google storu i nisam ju mogao skinut, nekako sam uvijek čekao da netko postavi neku sličnu njoj na internet.

# 5. LITERATURA

[1]. [https://discuss.cocos2d-x.org](https://discuss.cocos2d-x.org/), 12.09.2019.

[2]. <https://github.com/cocos2d/cocos2d-js> 12.09.2019.

[3].<https://www.gamefromscratch.com/post/2014/09/29/Cocos2D-x-Tutorial-Series-Installation-Creating-a-Project-and-Hello-World.aspx> 12.09.2019.

[4]. <https://github.com/SonarSystems/Cocos-Helper>, 12.09.2019.

[5].<https://cocos2d-x.org/reference/native-cpp/V3.5/d2/dc0/namespacecocos2d.html>, 12.09.2019.

[6].<https://www.gamefromscratch.com/post/2012/06/03/Cocos2D-HTML-release-a-tech-demo-Moon-Warriors.aspx>, 12.09.2019.

.

# 6. POPIS SLIKA

[Slika 1. Početni zalon sa uputama………………………………………………………………….9](#_Toc18968332)

Slika 2. Početak igrice [10](#_Toc18968333)

Slika 3. Igrica u tijeku [10](#_Toc18968334)

Slika 4. Kraj igrice i prikaz credita [11](#_Toc18968335)

Slika 5. Mogučnost skaliranja [11](#_Toc18968336)

Slika 6. Game.js kod za dodavanje pozadine [11](#_Toc18968338)

Slika 7. Gamemanager.js kod za prikaz puta [12](#_Toc18968338)

[Slika 8. Prikaz koda za detekciju pokreta .12](#_Toc18968332)

Slika 9. Prikaz koda u game.js [13](#_Toc18968333)

Slika 10. Prikaz koda za dodavanje lika [13](#_Toc18968334)

Slika 11. Prikaz kreiranja lika u game.js [15](#_Toc18968335)

Slika 12. Prikaz funkcije namjenjene skakaknju [15](#_Toc18968336)

Slika 13. Funkcije u oblak.js [16](#_Toc18968338)

[Slika 14. Funckije vezane za kretanje u game.js 17](#_Toc18968332)

Slika 15. Vrste tuba [18](#_Toc18968333)

Slika 16. Prikaz kako je zamišljen offset [18](#_Toc18968334)

Slika 17. Dio koda za stvaranje tuni [19](#_Toc18968335)

Slika 18. Funkcije u tube.js [20](#_Toc18968336)

Slika 19. Početni zaslon [21](#_Toc18968336)

Slika 20. Korigiranje mjera za veličinu našeg lika  [22](#_Toc18968336)

Slika 21. Namještanje varijabli u gamemanager.js koje poslje zovemo u game.js  [22](#_Toc18968336)

Slika 22. Postavljanje teksta za gameplay. [23](#_Toc18968336)