SVEUČILIŠTE U SPLITU  
FAKULTET ELEKTROTEHNIKE, STROJARSTVA I BRODOGRADNJE

SEMINARSKI RAD

GameMaker Studio 2

Vlaho Petković

Leo Štroliga

Split, 16.12.2023.

**SADRŽAJ**

[1 UVOD 1](#_Toc153718613)

[2 INSTALACIJA I POSTAVLJANJE ALATA GAMEMAKER STUDIO 2 2](#_Toc153718614)

[3 OSNOVNE ZNAČAJKE GMS-a 2 7](#_Toc153718615)

[3.1 Stvaranje novog projekta 7](#_Toc153718616)

[3.2 Objekti, ”sprajtovi” i zvukovi 7](#_Toc153718617)

[3.3 Programiranje u GameMaker jeziku (GML) 8](#_Toc153718618)

[4 NAPREDNE ZNAČAJKE GMS-a 2 10](#_Toc153718619)

[4.1 Dizajn prostora i prijelazi 10](#_Toc153718620)

[4.2 Fizika i sudari 10](#_Toc153718621)

[4.3 Čestice i posebni efekti 11](#_Toc153718622)

[5 OPTIMIZACIJA, TESTIRANJE I IMPLEMENTACIJA 12](#_Toc153718623)

[5.1 Optimizacija 12](#_Toc153718624)

[5.2 Testiranje 12](#_Toc153718625)

[5.3 Implementacija 12](#_Toc153718626)

[6 STAR WARS 13](#_Toc153718627)

[6.1 Programiranje svemirskog broda na način da se može kretati u prostoru 13](#_Toc153718628)

[6.2 Programirati svemirske stijene kako bi eksplodirale 18](#_Toc153718629)

[6.3 Završiti igru 24](#_Toc153718630)

[7 ZAKLJUČAK 28](#_Toc153718631)

[LITERATURA 29](#_Toc153718632)

[PRILOZI 30](#_Toc153718633)

[Kazalo slika, tablica i kodova 30](#_Toc153718634)

[Kazalo slika 30](#_Toc153718635)

[Popis oznaka i kratica 31](#_Toc153718636)

[SAŽETAK I KLJUČNE RIJEČI 32](#_Toc153718637)

Tablica 1-1 Doprinosi koautora rada

|  |  |
| --- | --- |
| **Koautor** | **Kratki opis doprinosa koatura u izradi rada** |
| Vlaho Petković | izrada igrice, poglavlje 6, popis oznaka i kratica |
| Leo Štroliga | poglavlja 1-5, poglavlje 7, sažetak i ključne riječi |

# UVOD

Razvoj igara razvio se kroz godine, te je postao dostupniji budućim programerima i ljubiteljima video igrica. Jedan od ključnih sudionika u svijetu igara jest *GameMaker* *Studio* *2* (GMS 2). Ovaj jaki program za izradu igara stekao je popularnost zbog svog korisnički prijateljskog sučelja, fleksibilnosti i snažnih značajki. Samim time upravo je idealan izbor za početnike i iskusne programere.

GMS 2 nudi intuitivnu platformu za dizajniranje, stvaranje prototipa i kreiranje igara na različitim platformama. Alati za razvoj igara igraju ključnu ulogu u oblikovanju industrije video igara. Oni omogućavaju programerima pretvaranje njihovih ideja u interaktivna iskustva, potičući inovacije i kreativnost. GMS 2 ističe se kao alat koji postiže ravnotežu između jednostavnosti korištenja i naprednih mogućnosti, omogućavajući programerima da se usredotoče na dizajn igara i mehanike umjesto na složeno programiranje.

Rad se sastoji od sedam poglavlja. U poglavlju dva opisana je instalacija i postavljanje GMS-a 2 na računalo. Poglavlja tri i četiri definiraju osnovne i napredne značajke GMS-a 2. Poglavlje pet fokusira se na završne stavke izrade igre kao što su optimizacija, testiranje i implementacija. U poglavlju šest opisana je izrada igrice „Star Wars“ pomoću GMS-a 2. Sedmo poglavlje je zaključak svega navedenog.

# INSTALACIJA I POSTAVLJANJE ALATA GAMEMAKER STUDIO 2

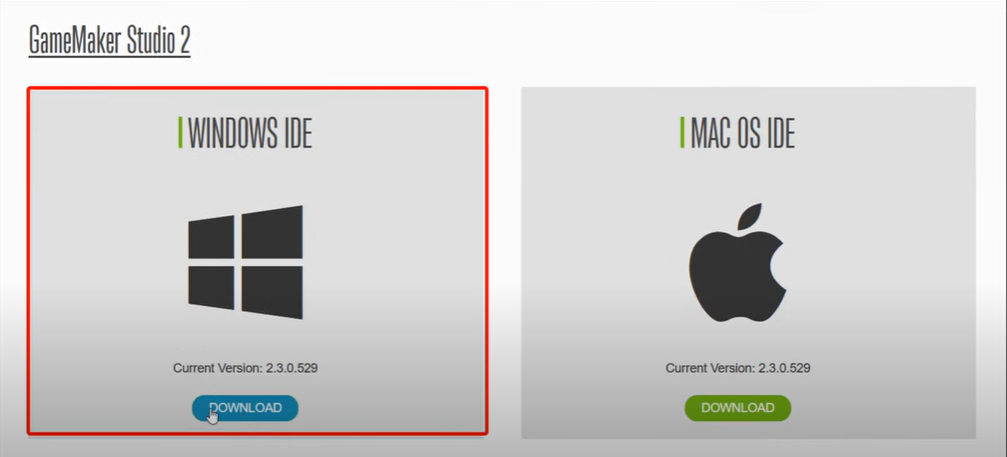
*GameMaker* *Studio* *2*, razvijen od strane *YoYo* *Games*, predstavlja evoluciju u svijetu razvoja video igara. Usporedba s drugim platformama za razvoj igara otkriva da GMS 2 pruža intuitivno sučelje i snažne značajke, čineći ga idealnim alatom za razvoj igara bez obzira na iskustvo korisnika.

Ključno za učinkovitu upotrebu GMS-a 2 je razumijevanje njegovog korisničkog sučelja. Korisničko sučelje pruža pregled radnog prostora GMS-a 2, uključujući ključne panele i prozore s kojima se radi tijekom procesa razvoja igara.

Potrebno je instalirati softver na računalo, kako bi se započelo s korištenjem GMS-a 2. Ovdje će biti razmotreni minimalni zahtjevi sustava, postupak preuzimanja i instalacije, te početno konfiguriranje GMS-a 2 kako bi bilo moguće stvaranje igara.

1. **Korak**

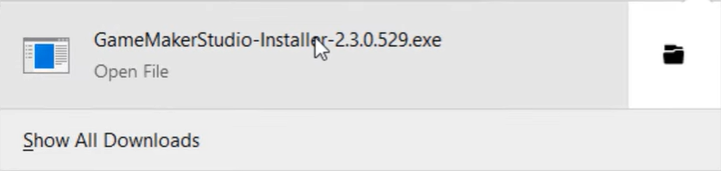
Preuzimanje GMS 2 verzije za Windows ili Mac, ovisno o operacijskom sustavu računala, sa službene stranice *YoYo Games* ili stranice *Game* *Maker*. Klikom miša na *Download* preuzima se željena verzija.



Slika 2‑1 Preuzimanje Windows/Mac verzije GameMaker Studia 2.

1. **Korak**

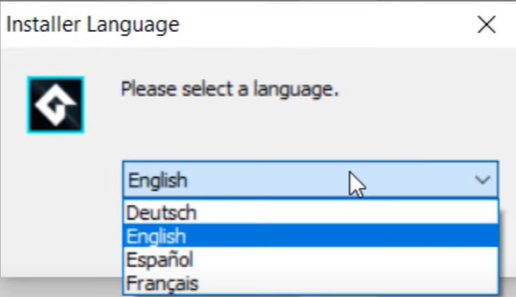
Otvaranje preuzete .exe datoteke.



Slika 2‑2 Otvaranje .exe datoteke.

1. **Korak**

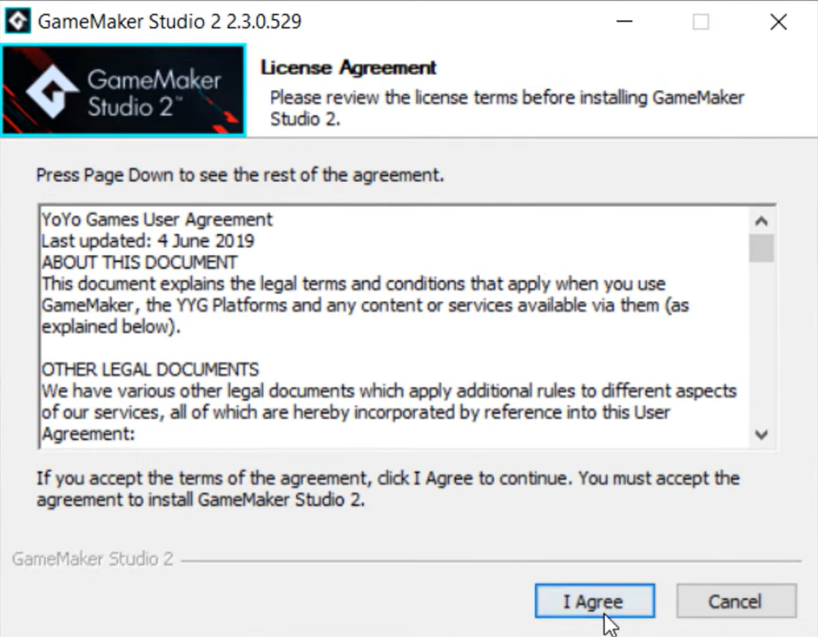
Odabir željenog jezika programa.



Slika 2‑3 Odabiranje jezika.

1. **Korak**

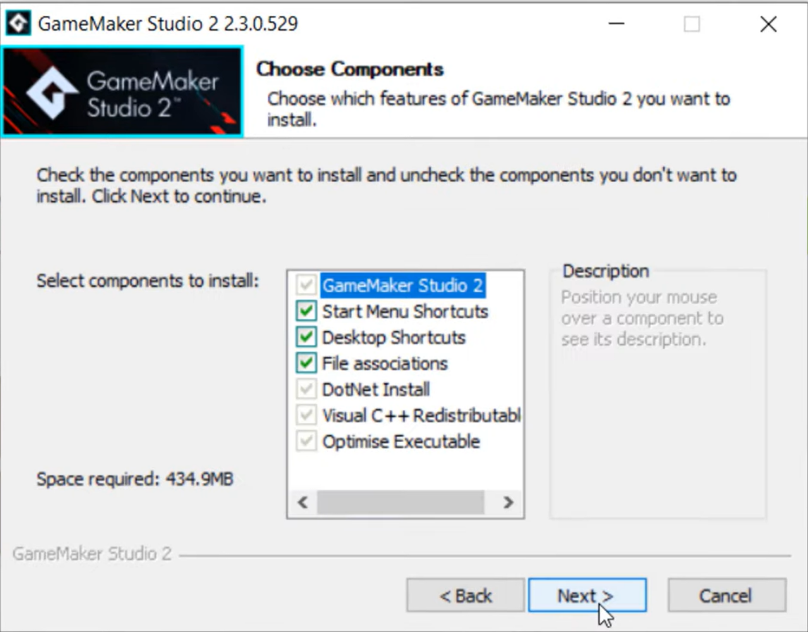
Prihvaćanje općih uvjeta upotrebe programa klikom miša na tipku *I Agree*.



Slika 2‑4 Prihvaćanje općih uvjeta upotrebe.

1. **Korak**

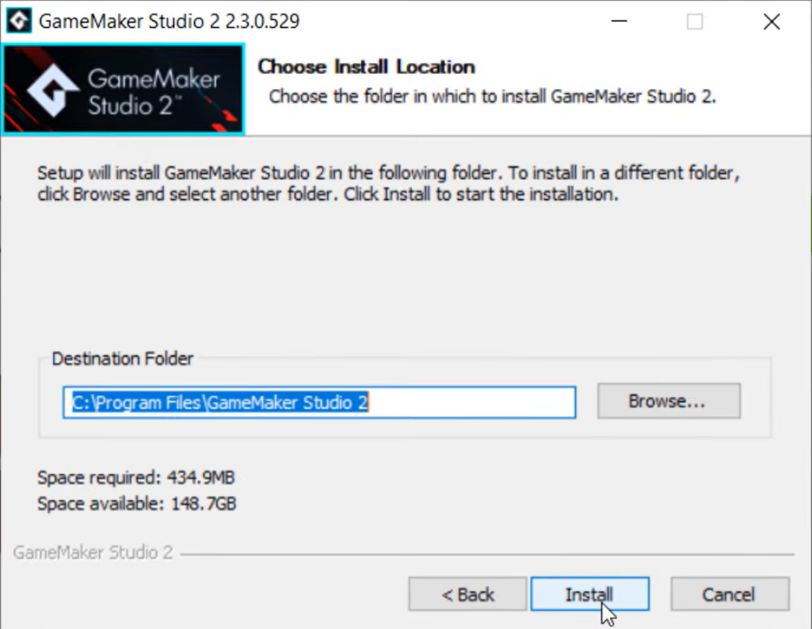
Odabir komponenti za instaliranje. Tri komponente su već odabrane automatski. Ostale su opcionalne. Nakon odabira komponenti klikne se na tipku *Next >*



Slika 2‑5 Odabir komponenti za instaliranje.

1. **Korak**

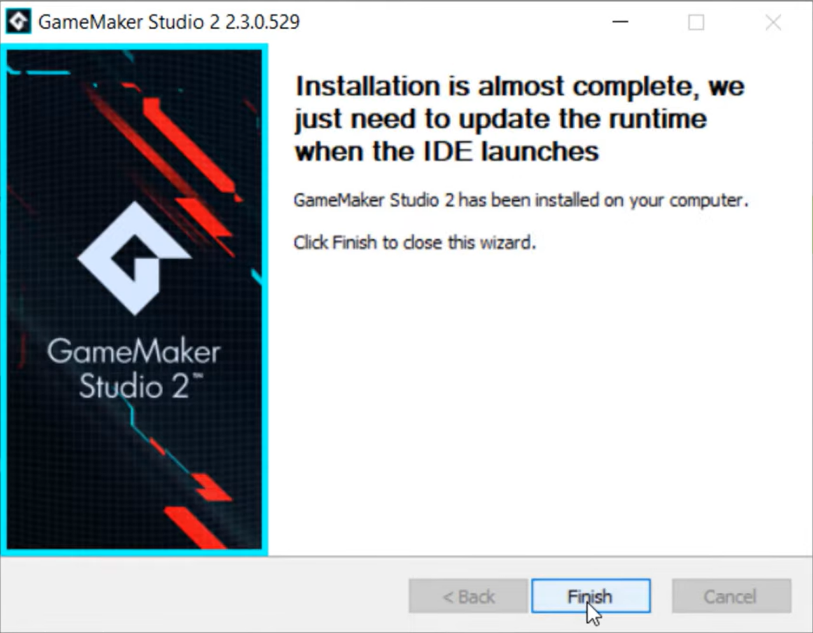
Odabir mjesta spremanja, odnosno mape u koju će se spremiti *GameMaker* *Studio* *2* nakon instalacije. Kada je mjesto odabrano, mišem se klikne na tipku *Install*.



Slika 2‑6 Instaliranje GMS-a 2.

1. **Korak**

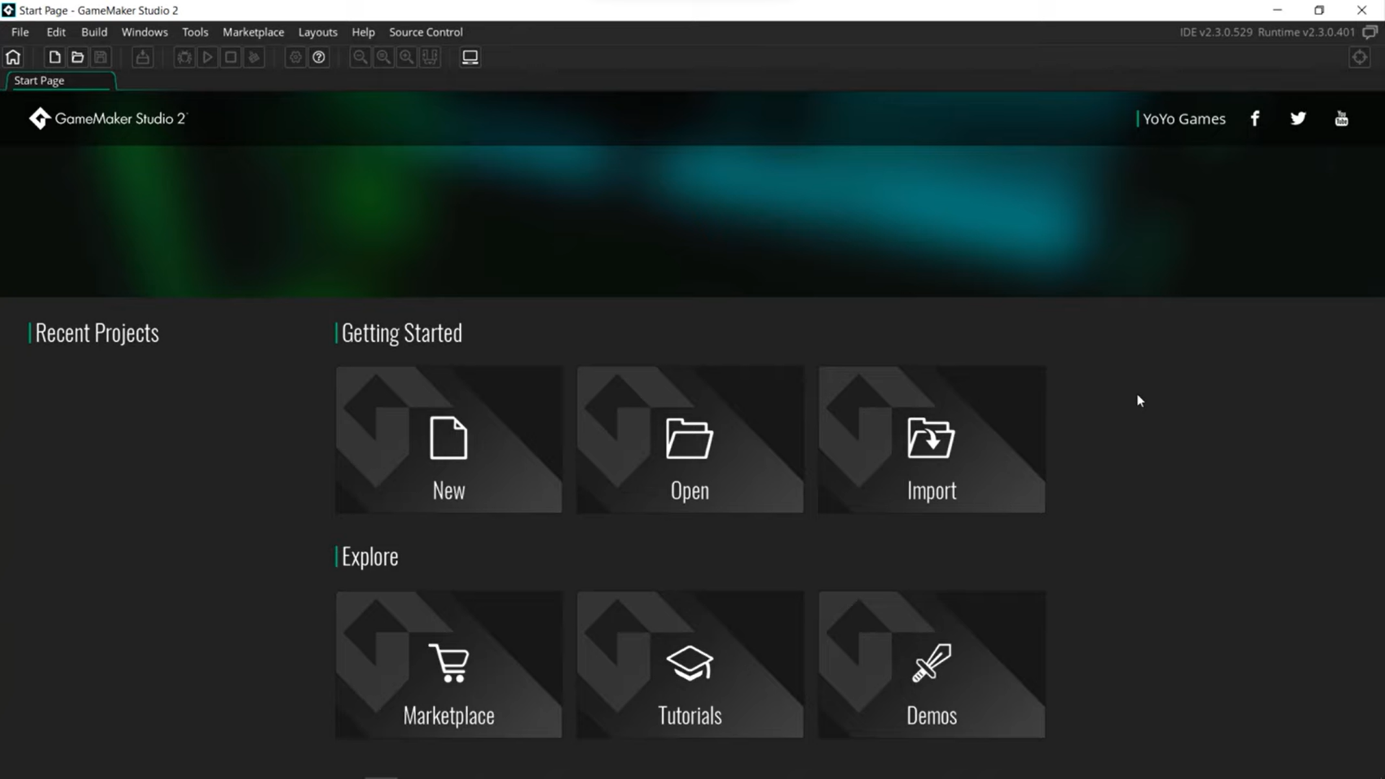
Nakon instalacije GMS-a 2, mišem se klikne na *Finish.*



Slika 2‑7 Završni korak instalacije programa.

1. **Korak**

*GameMaker* *Studio* *2* spreman je za korištenje.

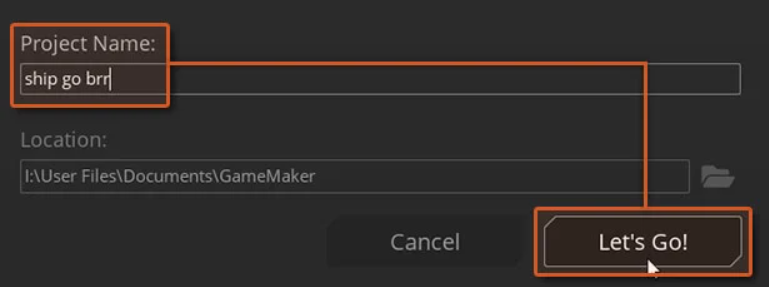


Slika 2‑8 GameMaker Studio 2.

# OSNOVNE ZNAČAJKE GMS-a 2

## Stvaranje novog projekta

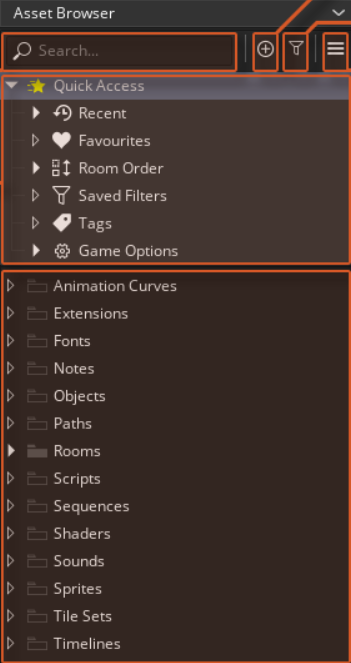
Glavna osnovna značajka za razvoj igara u GMS-u 2 je stvaranje novog projekta. Ovaj proces opisuje pokretanje novog projekta, kao i odabir platforme za koju se želi razvijati igra, bilo da je to Windows, MacOS, Android, iOS ili neka druga platforma. Na slici 2‑8, klikom na *New* kreira se novi projekt. Također, klikom na *Open* može se otvoriti već postojeći projekt na računalu, a klikom na *Import* „uvozi“ se projekt treće strane. Nakon kreiranja novog projekta projektu se da ime i pokrene se klikom na *Let's Go!*.



Slika 3‑1 Pokretanje novog projekta

## Objekti, ”sprajtovi” i zvukovi

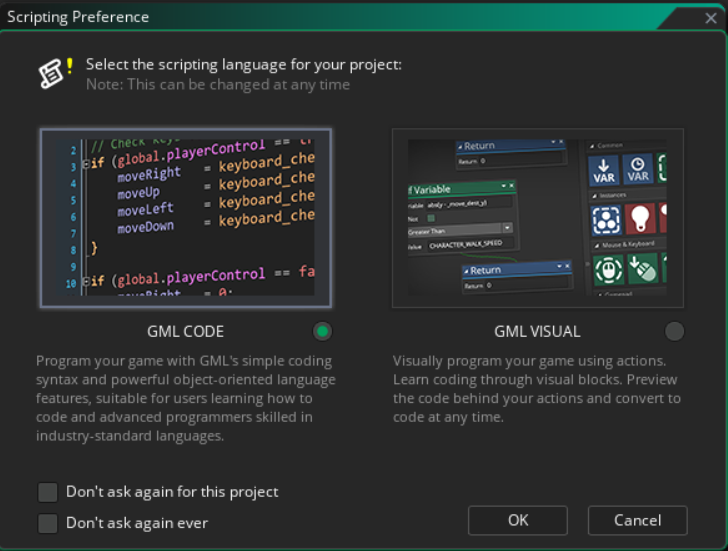
Razumijevanje osnovnih građevnih blokova igre ključno je za uspješan razvoj. To uključuje objekte, „sprajtove“ i zvuk. „Sprajtovi“ su samo (animirane) slike koje nemaju nikakvo ponašanje. Objekti obično imaju „sprajtove“ koji ih predstavljaju, ali objekti imaju ponašanje. Bez objekata, nema igre. GMS 2 ima potpuni audio program koji se temelji na zvukovnim formatima .ogg, .mp3 i .wav. Zvukovi ovih vrsta dodani u IDE-u mogu se koristiti u igri pomoću audio funkcija. Za stvari složenije od osnovnih zvučnih efekata ili reprodukcije jednog glazbenog komada, mogu se koristiti napredne audio funkcije koje omogućavaju modificiranje načina na koji se zvuk reproducira. Tu su i specijalizirane funkcije posvećene reprodukciji zvuka, pozicioniranju zvuka (za stvaranje 3D zvuka), grupiranju zvuka, sinkroniziranju zvuka i zvučnim efektima. Objekti, „sprajtovi“ i zvukovi nalaze se u dijelu *Asset Browser* (ukoliko nije vidljiv klikne se na *Windows* na navigacijskoj traci i odabere se *Asset Browser*)*.*



Slika 3‑2 Asset Browser.

## Programiranje u GameMaker jeziku (GML)

*GameMaker* *Language* (GML) omogućava programerima implementaciju funkcionalnosti igre. Može se koristiti GML za programiranje igara na dva različita načina: postavljanjem blokova za "pisanje" koda vizualno ili pisanjem koda ručno. Prva opcija, GML *Visual*, najbolje odgovara početnicima i hobi programerima koji žele brzo krenuti s izradom vlastitih igara. Druga opcija, GML *Code*, omogućava korištenje GML-a pisanjem koda ručno. Iako je GML *Visual* lakši za početak, GML *Code* je i dalje jednostavan programski jezik s moćnim značajkama (funkcije, metode, strukture i konstruktori, itd.).



Slika 3‑3 GameMaker Language

# NAPREDNE ZNAČAJKE GMS-a 2

## Dizajn prostora i prijelazi

U GMS-u 2, "prostor" je mjesto gdje se odvija igra. Dizajn prostora igra ključnu ulogu u stvaranju privlačnog i uzbudljivog igračkog iskustva. Razvojni programeri koriste uređivač prostora (*Room* *Editor*) za oblikovanje okoline, postavljanje objekata i definiranje rasporeda svojih igara. Ključni dijelovi dizajna prostora su pozadina kojom se oblikuje okolina, objekti i instance koji se stavljaju u prostor kao igrački objekti i kamera koja definira ono što igrač vidi na ekranu.

Prijelazi su vizualni i često animirani efekti koji se javljaju prilikom kretanja iz jednog prostora u drugi ili pri prijelazu između različitih dijelova igre. Dobro osmišljeni prijelazi doprinose ukupnom sjaju i profesionalnosti igre. Postoje dvije vrste prijelaza: prijelazi iz prostora u prostor i prijelazi scene. Prijelazi iz prostora u prostor javljaju se kada igrač prelazi iz jednog prostora u drugi. Unutar prostora, prijelazi se također mogu koristiti za prelazak između različitih scena ili dijelova igre. Na primjer, prijelaz može nastati pri prelasku s ekrana izbornika na stvarnu igru ili prilikom napredovanja s jedne razine na drugu.

## Fizika i sudari

U razvoju igara, uključivanje fizike u igranje dodaje sloj realizma i dinamike virtualnom svijetu. *GameMaker* *Studio* *2* pruža snažan sustav fizike koji omogućava programerima simuliranje različitih fizičkih interakcija poput gravitacije, trenja i kretanja objekata. Fizika u igrama omogućava realne animacije, odziv na pokrete likova, te angažirajuće mehanike igre. Također GMS 2 omogućava spojeve i ograničenja čime je omogućena simulacija složenih mehaničkih sustava.

Sudari su temeljni aspekt razvoja igara koji definira kako se objekti međusobno ponašaju u igračkom svijetu. *GameMaker* *Studio* *2* pruža svestran sustav sudara koji omogućava programerima otkrivanje i reagiranje na sudare između različitih objekata. Također, moguće je definirati akcije koje će se poduzeti ukoliko se sudar dogodi.

## Čestice i posebni efekti

Čestice su ključni vizualni elementi koji često služe za poboljšanje estetike igre i stvaranje raznih posebnih efekata. U GMS-u 2, čestice se mogu postaviti na proizvoljna mjesta i koriste se za stvaranje efekata poput dima, magle, iskri, kiše i mnogo toga.

Posebni efekti dodaju sloj vizualne privlačnosti i dinamike u igru. *GameMaker* *Studio* *2* pruža niz ugrađenih funkcija i mogućnosti za implementaciju raznih posebnih efekata. To mogu biti svjetlosni efekti, zamućivanje, transformacija čestica, itd.

# OPTIMIZACIJA, TESTIRANJE I IMPLEMENTACIJA

## Optimizacija

Optimizacija igre ključna je za postizanje glatke i učinkovite izvedbe. Obavlja se istraživanje strategije optimizacije u GMS-u 2, uključujući smanjenje broja instanci, efikasno korištenje resursa i druge tehnike koje će poboljšati performanse igre.

## Testiranje

Kvalitetno testiranje igre ključno je prije nego što se objavi širokoj publici. Ovaj dio pokriva različite aspekte testiranja, uključujući funkcionalno testiranje, testiranje korisničkog iskustva, i rješavanje problema vezanih uz performanse i stabilnost igre.

## Implementacija

Kada je razvoj igre završen, vrijeme je za implementaciju i dijeljenje s drugima. *GameMaker* *Studio* *2* pruža različite opcije implementacija, uključujući platforme poput Windows-a, Mac-a, Android-a i iOS-a.

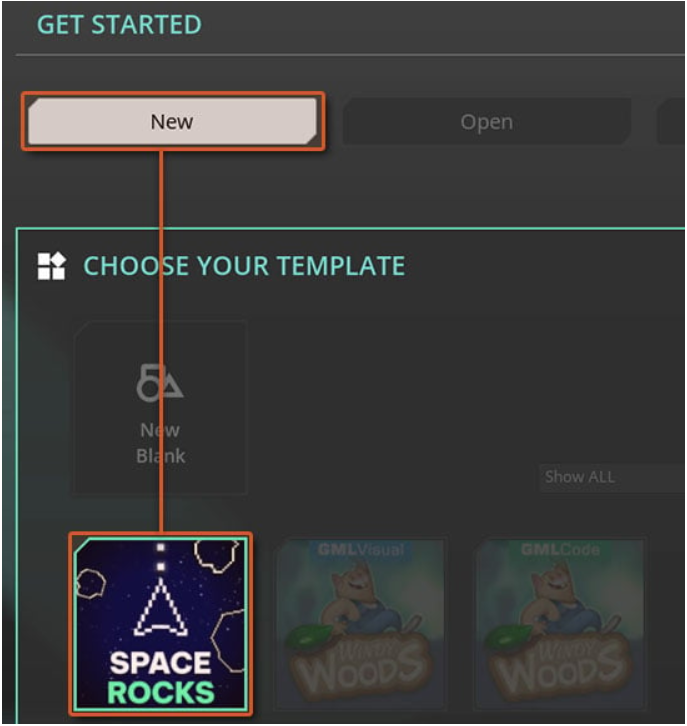
# STAR WARS

Video igrica *Star* *Wars* jednostavna je igrica napravljena pomoću GMS-a 2 u GML *Visual* jeziku. Predstavlja verziju igrice „pucača u svemiru“ („*Space* *Shooter*“). Igrica se sastoji od svemirskog broda koji uništava svemirske stijene u svemiru. *Star* *Wars* igrica napravljena je u tri koraka:

1. korak: programirati svemirski brod na način da se može kretati u prostoru
2. korak: programirati svemirske stijene kako bi eksplodirale
3. korak: završiti igru

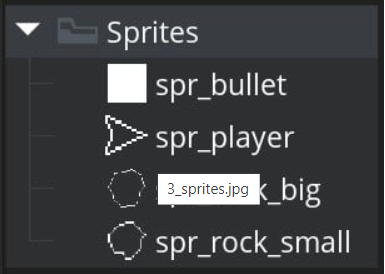
## Programiranje svemirskog broda na način da se može kretati u prostoru

Prvo se kreira novi projekt. Kreiranje novog projekta objašnjeno je u poglavlju 3.1. Nakon klika mišem na *New* odabire se željeni predložak. U ovom slučaju postoji predložak koji se zove *Space Rocks.*



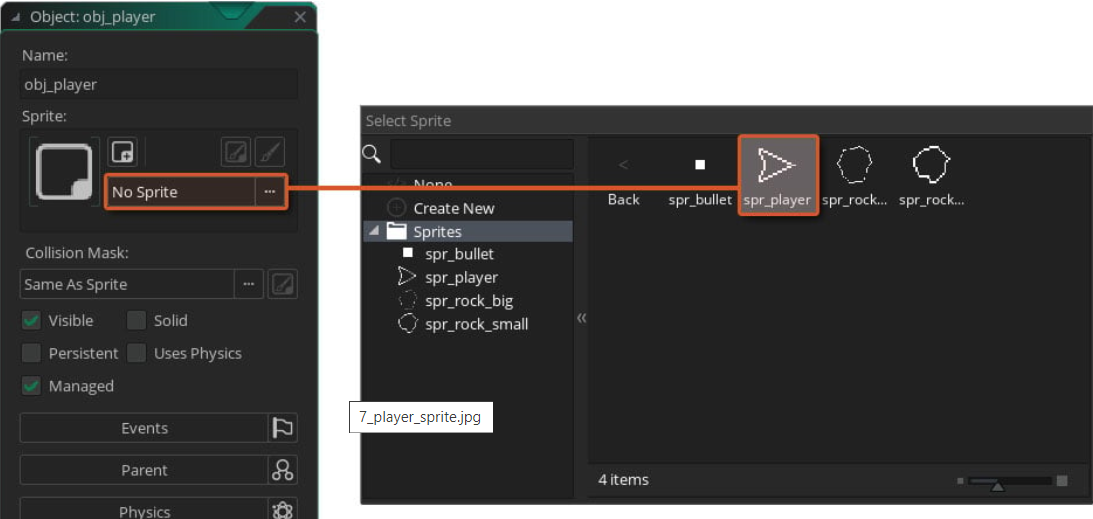
Slika 6‑1 Kreiranje projekta Star Wars.

Nakon odabira predložka, projektu se pridoda naziv, te ga se pokrene. Nakon pokretanja, s desne strane nalazi se *Asset Browser* koji sadrži sva sredstva potrebna za izradu igrice. Sadrži „sprajtove“ koji predstavljaju slike koje se koriste u igrici.



Slika 6‑2 „Sprajtovi“ Star Wars-a.

Kako bi se „sprajtovi“ „oživili“, oni se pridružuju objektima. Objekti „oživljavaju“ slike programirajući ih da se kreću i pucaju. Svemirski brod, metci i svemirske stijene su kreirani kao objekti, te je svakom od njih pridružen „sprajt“. Kreirani je objekt obj\_player, te mu je pridružen „sprajt“ spr\_player.



Slika 6‑3 Keriranje objekta obj\_player.

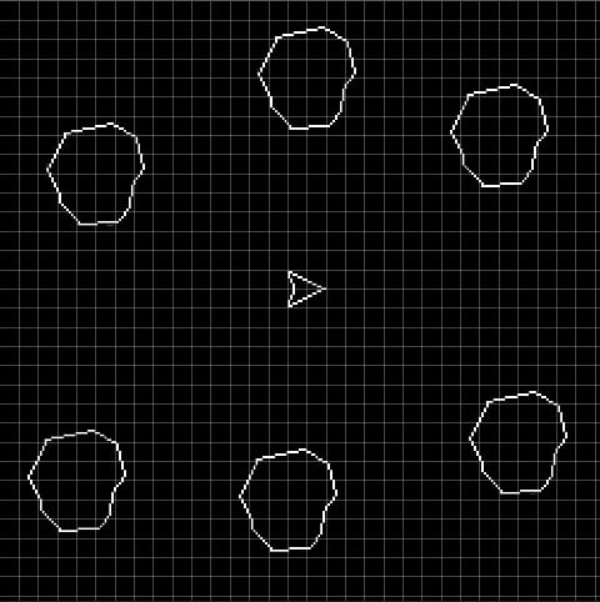
Isto to napravljeno je i za ostale objekte. Objektu obj\_bullet pridružen je „sprajt“ spr\_bullet, objektu obj\_rock spr\_rock\_big i objektu obj\_game nije pridružen „sprajt“ jer je to „tihi“ objekt koji upravlja igrom.

Nakon uspješnog kreiranja objekata, potrebno ih je staviti u prostor. To se radi na način da se klikne u *Asset Browser* na stavku *Rooms,* te dvostrukim klikom na *Room1.*



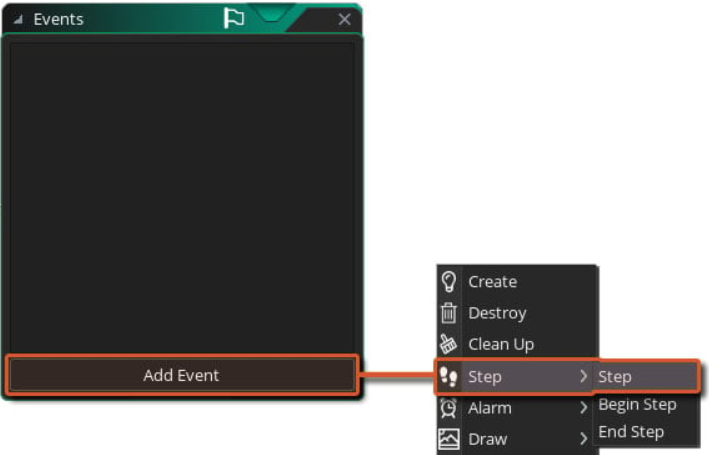
Slika 6‑4 Odabiranje prostora.

S lijeve strane će se pojaviti uređivač prostora (*Room Editor*). Klikom na instance (*instances*) mogu se dodavati kreirani objekti u prostor. Stoga se doda jedan objekt tipa obj\_player (strelica koja predstavlja svemirski brod kojim korisnik upravlja) i šest objekata tipa obj\_rock (stijene koja predstavljaju svemirske stijene) pomoću metode povuci i spusti (*Drag and Drop*). Svi objekti koji su potrebni za igranje igrice su u sceni, samo nisu programirani da poduzimaju određene akcije.

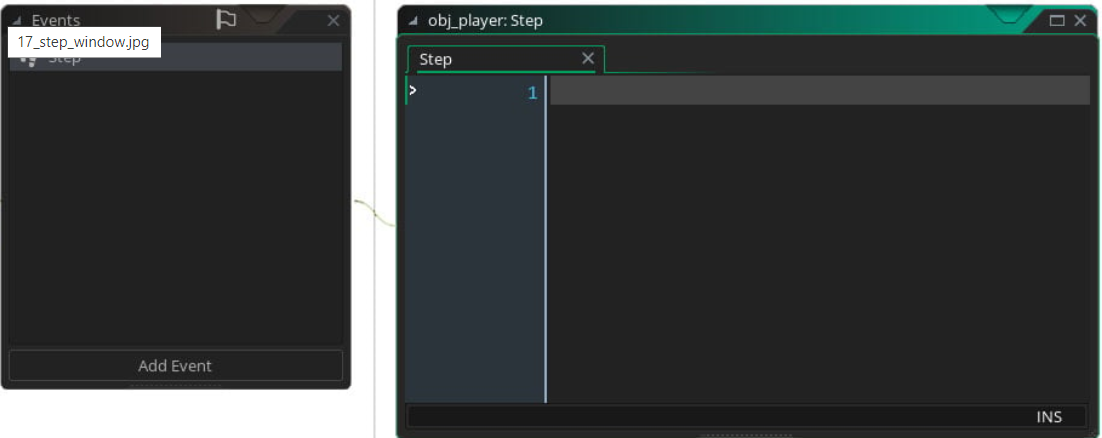


Slika 6‑5 Objekti u prostoru.

Kako bi poduzimali određene akcije radi se sljedeće. Dvostrukim klikom na obj\_player pojavi se prozor s događajima, takozvani *Events Window.* Klikne se na *Add Event -> Step -> Step.*

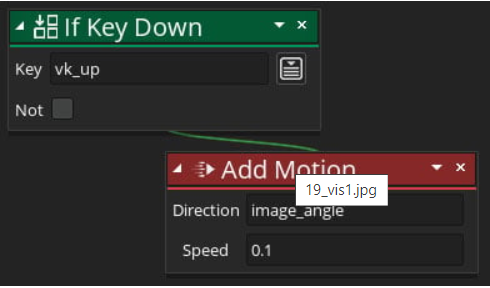


Slika 6‑6 Add Events.



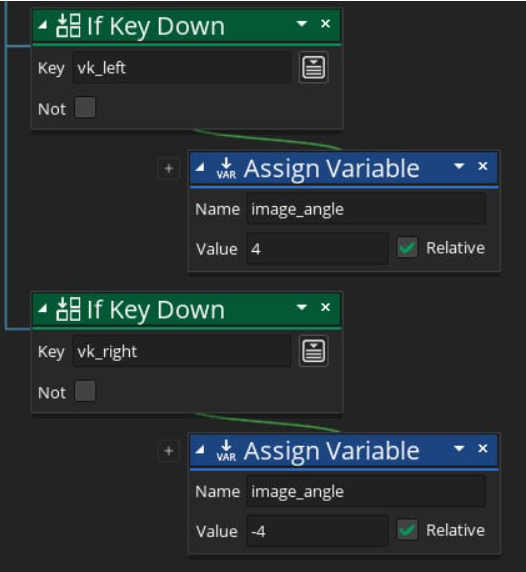
Slika 6‑7 Step Event prozor

Sada se obavlja programiranje događaja. Budući da se radi o GML Visualu, komponente se kreiraju metodom Povuci i Spusti (*Drag and Drop*). Sve komponente nalaze se s desne strane *Step* prozora. Dodaju se sljedeće komponente u *Step* prozor.



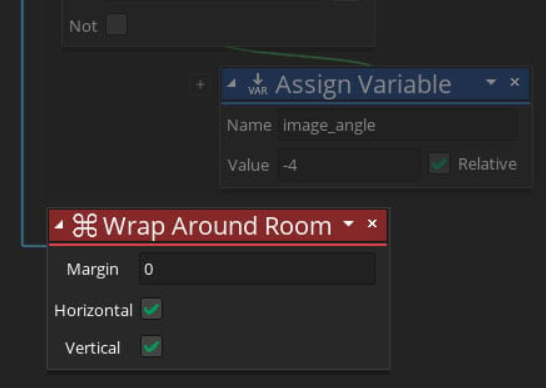
Slika 6‑8 Dodavanje mogućnosti kretanja objekta obj\_player.

Ovim komponentama omogućeno je kretanje objekta obj\_player pomoću gornje strelice na tipkovnici računala. *image\_angle* označava usmjerenje broda, a brzina (Speed) kolikom brzinom se obj\_player kreće. Sada se dodaje mogućnost zakretanja objekta obj\_player u lijevo ili u desno.



Slika 6‑9 Dodavanje mogućnosti rotiranja objekta obj\_player.

Ukoliko se pritisne lijeva strelica na tipkovnici računala oduzima se četiri od image\_angle, a ukoliko se pritisne desna strelica na tipkovnici računala doda se četiri na image\_angle. Sada je omogućeno rotiranje i kretanje, u svim smjerovima, objekta obj\_player koji predstavlja svemirski brod. Kako bi obj\_player ostao unutar odabranog prostora dodaje se sljedeći događaj (*Event*) u *Step* prozor.

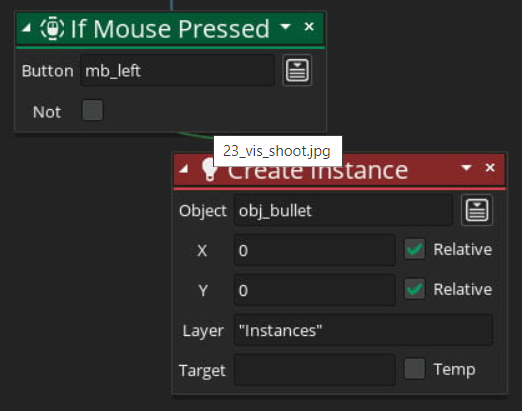


Slika 6‑10 Dodavanje događaja Wrap.

Dodavanjem događaja *Wrap* omogućeno je da obj\_player nikad ne nestane iz promatranog prostora. Ukoliko se dogodi da on izađe iz promatranog prostora, on će se opet pojaviti u prostoru, ali s druge strane.

## Programirati svemirske stijene kako bi eksplodirale

U prethodnom poglavlju kreiran je objekt obj\_bullet. Kako bi sve svemirski brod mogao pucati, treba kreirati instancu objekta obj\_bullet u obj\_player *Step* *Event* prozoru.



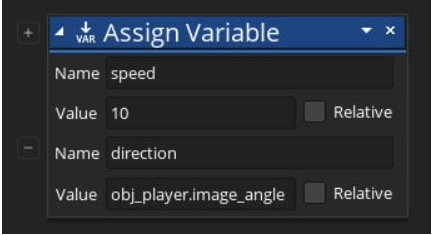
Slika 6‑11 Kreiranje instance objekta obj\_bullet.

Dakle, kreiranjem instance omogućeno je pucanje i to klikom na lijevu tipku miša. Za X i Y instanci su prenijete koordinate svemirskog broda tako da su metci kreirani na istom mjestu gdje je i brod. Kako bi metci postali pokretni, odnosno kako bi se gibali kroz prostor u željenom smjeru ili prema meti, radi se sljedeće. Dvostruko se klikne na obj\_bullet. Pojavi se prozor s događajima. Klikne se na *Add Event -> Create.*



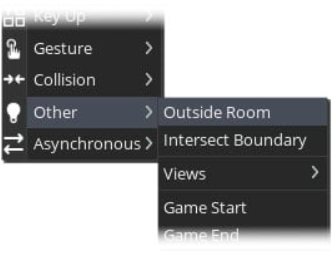
Slika 6‑12 Prozor s događajima objekta obj\_bullets.

U Create prozor dodaje se komponenta:



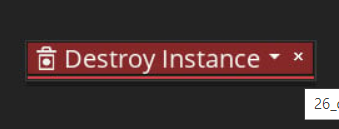
Slika 6‑13 Assign Variable komponenta.

Komponentom *Assign Variable* dodijeli se metcima brzina vrijednosti 10. Označava broj piksela. U ovom slučaju to je 10, po okviru. Ovim se dobiva kretanje, odnosno pucanje metaka, u smjeru prema kojem je okrenut svemirski brod. Također, želi se postići uništenje metaka ukoliko izađu iz prostora. U *Event* prozoru klikne se na *Add Event -> Other -> Outside Room.*



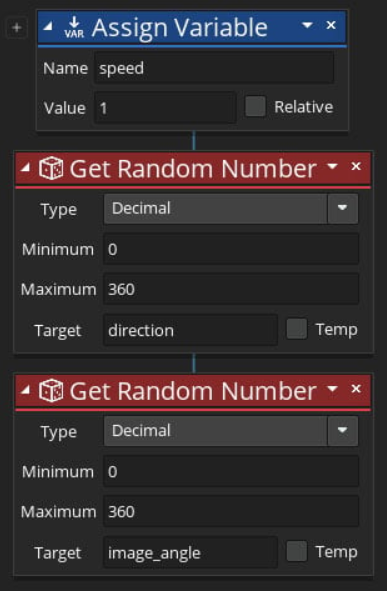
Slika 6‑14 Kreiranje prozora Outside Room.

U prozor *Outside Room* doda se komponenta *Destroy Instance* kojom se uništavaju metci ukoliko izađu iz prostora.



Slika 6‑15 Komponenta Destroy Instance.

Nadalje, svemirske stijene se trebaju pomicati u proizvoljnom smjeru. Dvostruko se klikne na objekt obj\_rock. Zatim *Add Event -> Create.* U *Create* prozor dodaju se komponente.



Slika 6‑16 Dodavanje komponenti u Create prozor objekta obj\_rock.

Brzina kretanja stijena postavljena je s *Assign Variable* na vrijednost 1. Zatim se postavi smjer kretanja s *Get Random Number* koji može biti bilo koja nasumična vrijednost od 0 do 360. S istom tom komponentom postavi se također nasumična vrijednost rotiranja „sprajta“. Također, javlja se isti problem kao i kod kretanja svemirskog broda. Ukoliko stijena napusti prostor, ona će nestati. U obj\_rock dodaje se *Step Event* i u njegov prozor dodajemo komponente.



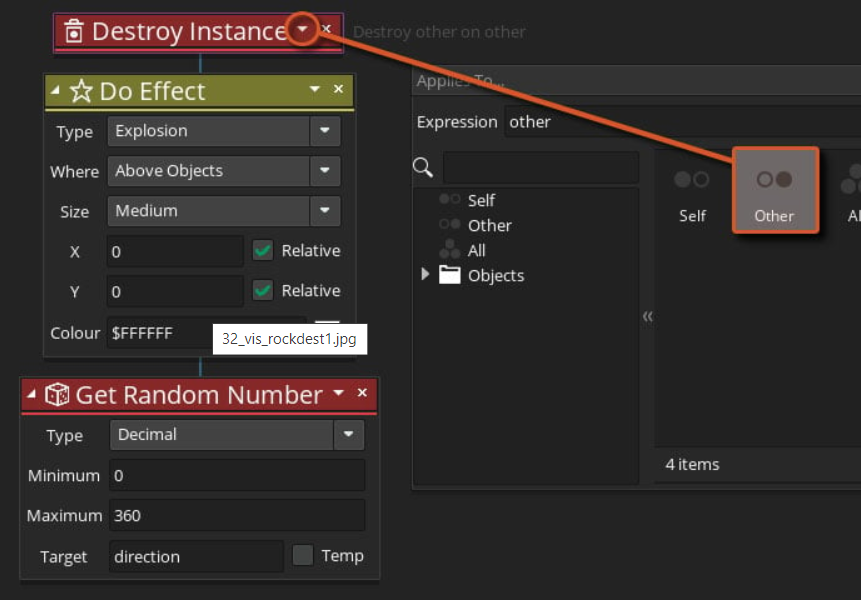
Slika 6‑17 Dodavanje Step Event-a.



Slika 6‑18 Komponente Step Event prozora.

Komponentom *Wrap Around Room* mogućava se omotavanje za stijene, s razmakom od 100. To znači da će se stijena početi prebacivati na drugi dio prostorije tek kad bude udaljena više od 100 piksela izvan prostorije. Pomoću *Assign Variable* komponente omogućava se konstanto rotiranje stijene.

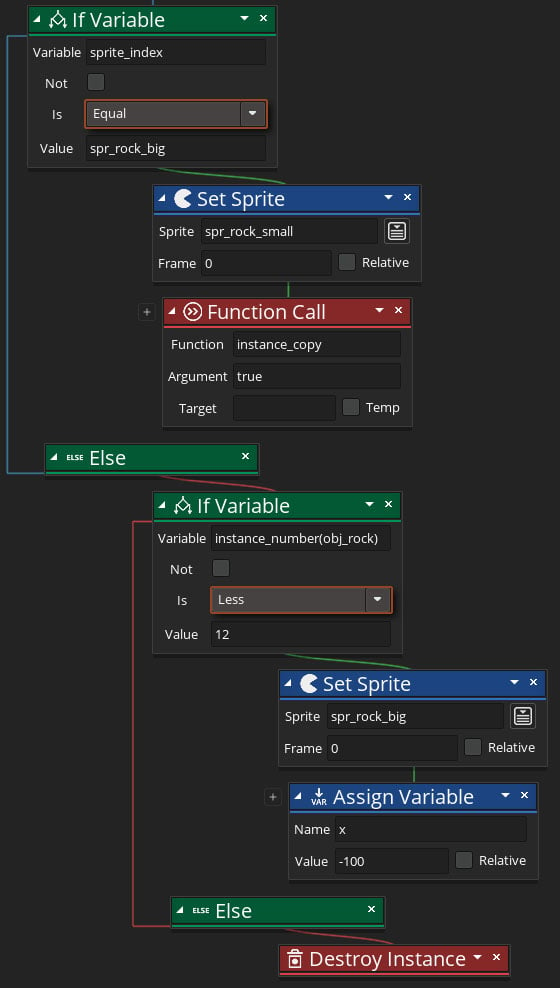
Sada slijedi proces uništavanja metaka i stijena. Želi se postići da prilikom sudara metka i stijene, dođe do uništenja i metka i stijene. Treba biti omogućeno da se prilikom uništavanja stijene, velika stijena pretvori u manje, a manj stijena pretvori ponovno u veliku. Kada se dvije instance sudare to se zove kolizija (sudar). Kreće programiranje sudara metaka i stijena pomoću komponenti. Dvostruko se klikne na objekt obj\_rock. Zatim *Add Event -> Collision* i odabere se obj\_bullet. Dodaju se komponente u novostvoreni prozor *Collision.*



Slika 6‑19 Dodavanje komponenti u Collision prozor.

Komponente rade četiri stvari. Uništava se "druga" instancu, koja je metak. *GameMaker* daje varijablu *Other* u događaju sudara, tako da se može izmijeniti instanca s kojom se sudara. U GML *Visual* jeziku, koristi se strelica prema dolje kako bi se dodijelila radnja *Other* instanci, tako da se metak uništi. Stvara se efekt eksplozije na lokaciji stijene. Njegova veličina je srednja i boja mu je bijela. Postavlja se smjer na nasumičnu vrijednost. Kada stijena konačno postane mala stijena ili se ponovno pojavi, kretat će se u novom smjeru.

Sada će se isprogramirati pretvaranje velikih stijena u manje i pretvaranje manjih u velike.



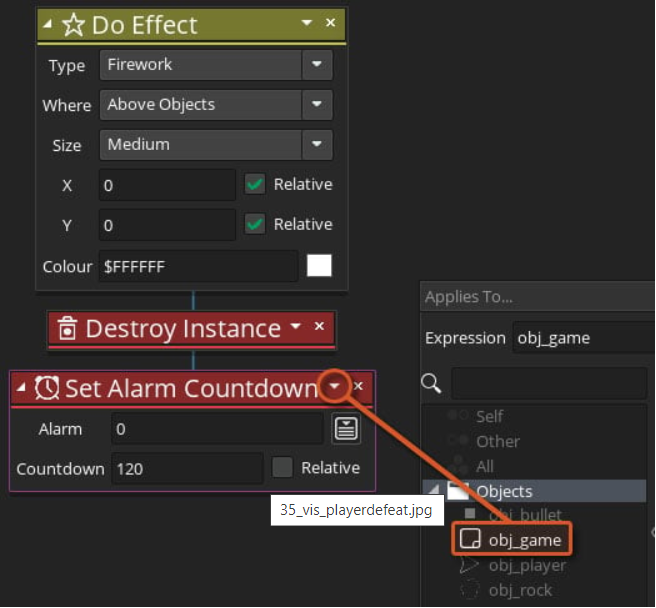
Slika 6‑20 Pretvaranje velikih stijena u manja i manjih u velika.

Prvi blok provjerava je li stijena velika, provjerom njenog „sprajta“. Ako je velika, mijenja se u malu stijenu, a zatim se instanca kopira kako bi se dobile dvije male stijene. Drugi blok počinje s *else*, što znači da se izvršava samo ako prethodni uvjet nije ispunjen, što znači da je stijena mala. Zatim se provjerava je li u prostoru manje od 12 stijena, i ako je to točno, mijenja stijenu u veliku stijenu. Također, postavlja se x (horizontalna) pozicija stijene na -100 kako bi stijena izašla izvan prostora i kako bi se mogla ponovno pojaviti. Završni blok izvršava se samo ako su oba prethodna uvjeta lažna, što znači da je stijena mala i već ima 12 ili više stijena u prostoru. U tom slučaju, stijena se uništava.

Stijene se sada mogu uništavati i ponovno pojavljivati, te se može igrati beskonačno vremena.

## Završiti igru

Kako se igra ne bi igrala beskonačno vremena, u igru će biti dodana stavka koja će predstavljati opasnost. Ukoliko se svemirski brod i stijena sudare, svemirski brod će se uništiti. Igrica će se resetirati dvije sekunde nakon uništenja broda. To će biti omogućeno korištenjem *Alarma*. Dvostrukim klikom na obj\_player ući će se u taj objekt, te će se dodati događaj kolizije putem *Add Event -> Collision ->* obj\_rock. U *Collision* prozoru dodaju se sljedeće komponente:



Slika 6‑21 Dodavanje komponenti u Collision prozor.

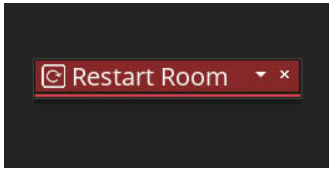
Ovdje se događaju tri stvari. Stvaranje efekta *vatrometa* sa srednjom veličinom i bijelom bojom. Uništavanje instance igrača. Postavljanje alarma 0 u obj\_game na 120. obj\_game će pokrenuti svoj prvi alarm (Alarm 0) nakon 120 sličica (dvije sekunde). U tom događaju ponovno se pokrene prostor. Alarm se nije mogao staviti u obj\_player jer je to instanca koja je uništena.

Kako bi ponovno pokrenuli prostor, dvostrukim klikom pozicionira se u objekt obj\_game. Klikne se *Add Event -> Alarm -> Alarm 0.*



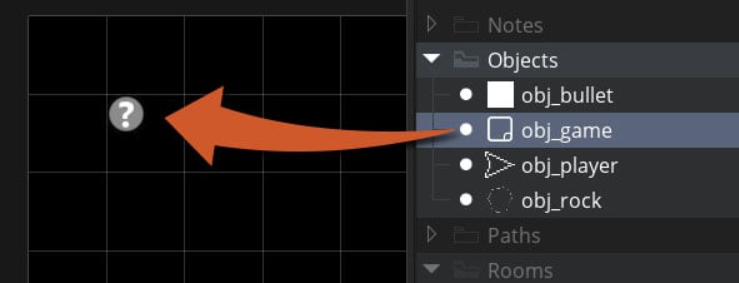
Slika 6‑22 Kreiranje Alarm 0 prozora.

U prozor se doda komponenta:



Slika 6‑23 Komponenta Restart Room.

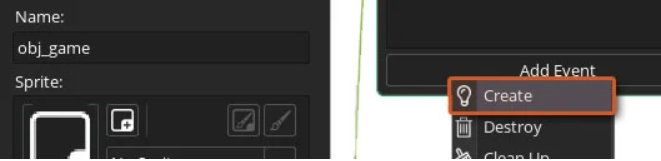
Ova akcija će ponovno pokrenuti sobu, te se igra može igrati ponovno. Kako bi objekt obj\_game funkcionirao, treba njegovu instancu dodati bilo gdje u prostor *Room1.* Dodaje se kao što su dodane i ostale instance (Slika 6‑5), što je i objašnjeno u poglavlju 6.1.



Slika 6‑24 Dodavanje instance objekta obj\_game.

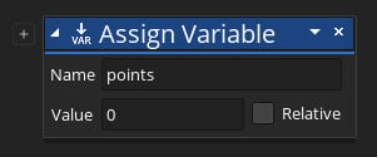
Igrica se sada igra dok se ne udari svemirskim brodom u stijenu. Nakon sudara, točno dvije sekunde nakon, igrica se ponovno pokreće.

Igrica je spremna za uporabu, ali još samo treba napraviti komponentu koja će bilježiti rezultat u igrici. To se radi na sljedeći način. Otvori se objekt obj\_game, te se klikne na *Add Event -> Create*.



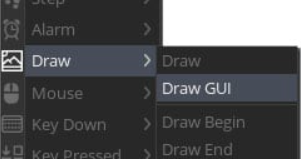
Slika 6‑25 Dodavanje događaja.

U novostvoreni *Create* prozor doda se komponenta varijable kojoj je dodijeljeno ime „points“, te je postavljena na vrijednost 0.

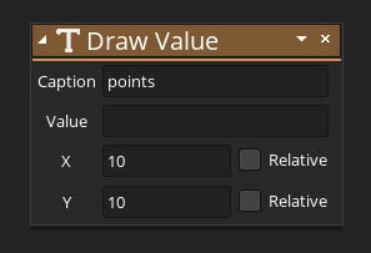


Slika 6‑26 Varijabla „points“.

Također, potrebno je rezultat vizualno prikazati u odabranom prostoru na sljedeći način. U objekt obj\_game klikom na *Add Event -> Draw -> Draw GUI* kreira se *Draw* *GUI* prozor. Dodaje se komponenta koja će prikazati trenutni zgoditak u prostoru. Odabir koordinata gdje će se rezultat prikazati je proizvoljan. U ovom slučaju koordinate su (10, 10).

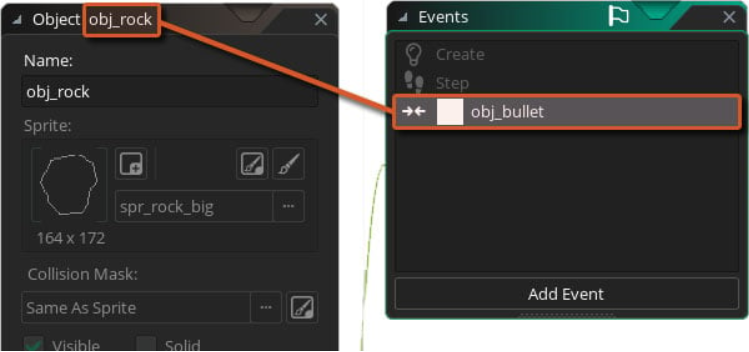


Slika 6‑27 Stvaranje Draw GUI prozora.

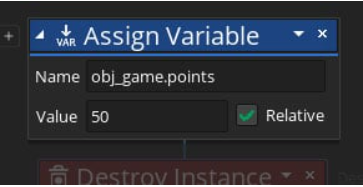


Slika 6‑28 Dodavanje komponente u Draw GUI prozor.

U prostoru se sada pokazuje rezultat, ali je cijelo vrijeme jednak nuli. Sada će se dodati komponenta koja će povećavati rezultat kada se stijena uništi. Otvori se objekt obj\_rock, te se u postojeći *Collision* događaj s obj\_bullet objektom doda komponenta varijable koja će povećavati rezultat za proizvoljnu vrijednost. U ovom slučaju će se rezultat povećavati za 50.



Slika 6‑29 Otvaranje postojećeg Collision događaja.

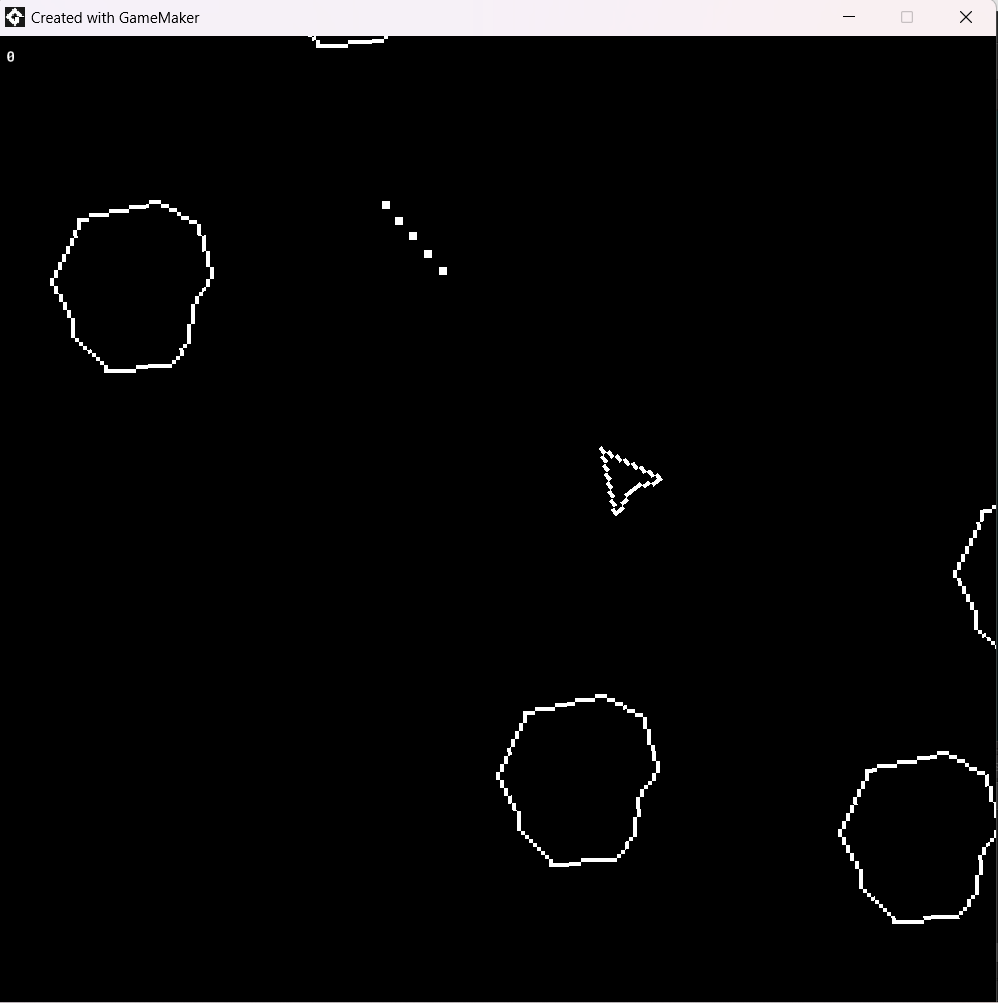


Slika 6‑30 Dodavanje komponente varijable u Collision prozor.

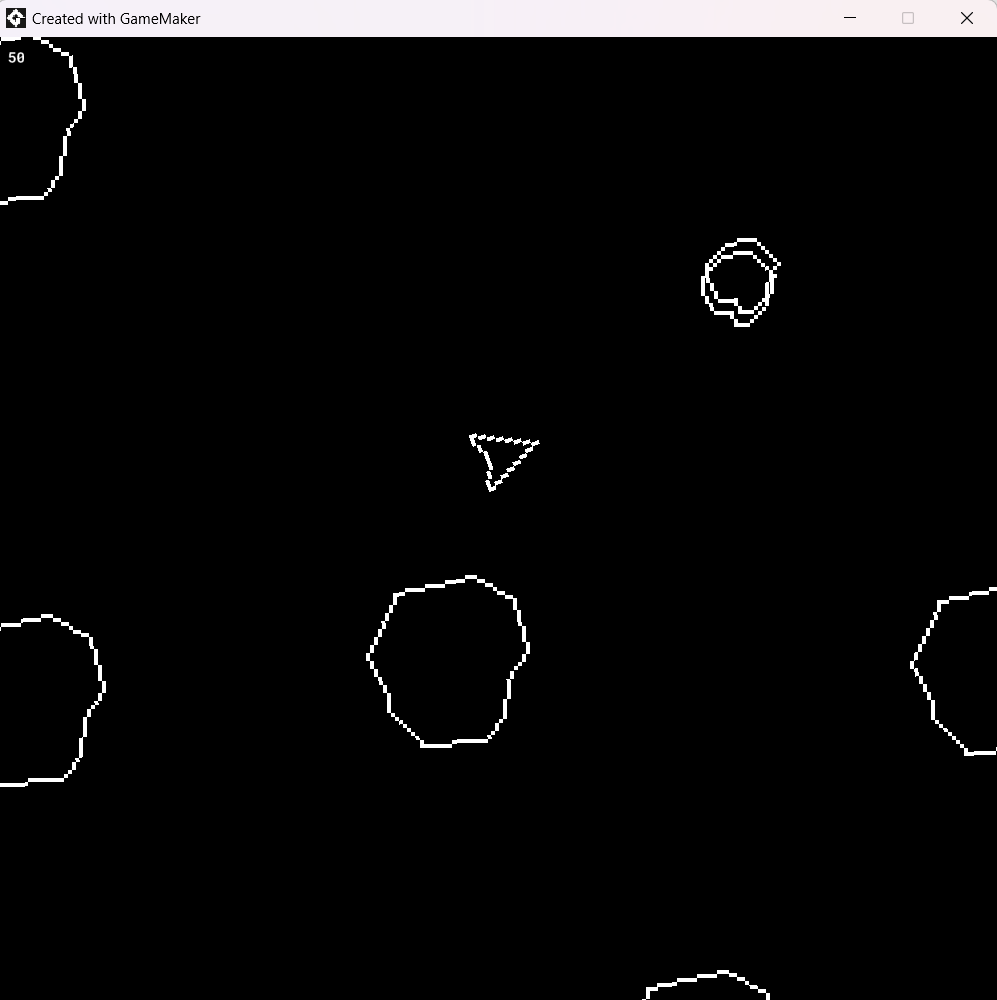
Igrica je konačno gotova i potpuna i spremna je za upotrebu.



Slika 6‑31 Pokrenuta igrica Star Wars



Slika 6‑32 Pucanje u igrici



Slika 6‑33 Razbijanje stijene na dvije manje



Slika 6‑34 Star Wars igrica

# ZAKLJUČAK

U dinamičnom svijetu razvoja igara, *GameMaker* *Studio* *2* ističe se kao svestran i dostupan alat koji omogućava korisnicima pretvaranje svojih vizija u interaktivna digitalna iskustva. *GameMaker* *Studio* *2* nije samo softverska aplikacija. Ona ujedno potiče i kreativnosti, inovacije i omogućava ambicioznim programerima pretvaranje svojih želja u stvarnost.

U svojoj suštini, *GameMaker* *Studio* *2* ističe se u smanjivanju razlika između kompleksnosti kodiranja i pristupačnosti. Sa svojim intuitivnim sučeljem povlačenja i ispuštanja, čak i oni bez velikog iskustva programiranja mogu zaroniti u svijet razvoja igara. Omogućava pojedincima sudjelovanje u procesu razvoja igara, bilo da se radi o hobi programerima ili iskusnim programerima.

Jedna od izuzetnih značajki GMS-a 2 je snažan skriptni jezik GML (*GameMaker* *Language*). Ovaj moćan jezik pruža programerima fleksibilnost za implementaciju složenih sustava i ponašanja, nudeći ravnotežu između jednostavnosti za početnike i dubine za napredne korisnike. Uključivanje skriptnog jezika proširuje privlačnost alata, privlačeći korisnike koji traže i jednostavnost upotrebe i mogućnost istraživanja granica svojih projekata.

U svijetu modernog razvoja igara, gdje je inovacija ključna, *GameMaker* *Studio* *2* stoji kao dokaz napretka industrije. Njegove kontinuirane nadogradnje i razvoj potaknut zajednicom pokazuju predanost praćenja trendova i novih tehnologija industrije. Procvat zajednice *GameMaker*, bogate forumima, kursevima i dijeljenim resursima, dodatno utvrđuje platformu kao suradnički centar gdje se znanje slobodno dijeli, potičući rast i inspiraciju.

Zaključno, *GameMaker* *Studio* *2* izlazi kao dinamičan i korisnički prijateljski alat za razvoj igara koji ohrabruje korisnike da pretvore svoje vizije u stvarnost. Njegova kombinacija pristupačnosti, snažnog skriptiranja i opcija razvoja na više platformi pozicionira ga kao vrijedan alat u rukama raznolikog skupa programera. Dok se opus razvoja igara nastavlja mijenjati, *GameMaker* *Studio* *2* ostaje dominantan, pozivajući korisnike da pomaknu granice, istraže mogućnosti i oslobode svoju kreativnost.

LITERATURA

1. Wikipedia (2023.), GameMaker, s Interneta: <https://en.wikipedia.org/wiki/GameMaker>, zadnji pristup: 13.12.2023.
2. YoYo Games (2023.), GameMaker Manual, s Interneta: <https://manual.gamemaker.io/monthly/en/#t=Content.htm>, zadnji pristup: 13.12.2023.
3. YoYo Games (2023.), GameMaker, s Interneta: <https://gamemaker.io/en>, zadnji pristup: 15.12.2023.

PRILOZI

Kazalo slika, tablica i kodova

Kazalo tablica

[Tablica 1-1 Doprinosi koautora rada 2](#_Toc154337232)

Kazalo slika

[Slika 2‑1 Preuzimanje Windows/Mac verzije GameMaker Studia 2. 2](#_Toc154341016)

[Slika 2‑2 Otvaranje .exe datoteke. 3](#_Toc154341017)

[Slika 2‑3 Odabiranje jezika. 3](#_Toc154341018)

[Slika 2‑4 Prihvaćanje općih uvjeta upotrebe. 4](#_Toc154341019)

[Slika 2‑5 Odabir komponenti za instaliranje. 4](#_Toc154341020)

[Slika 2‑6 Instaliranje GMS-a 2. 5](#_Toc154341021)

[Slika 2‑7 Završni korak instalacije programa. 5](#_Toc154341022)

[Slika 2‑8 GameMaker Studio 2. 6](#_Toc154341023)

[Slika 3‑1 Pokretanje novog projekta 7](#_Toc154341024)

[Slika 3‑2 Asset Browser. 8](#_Toc154341025)

[Slika 3‑3 GameMaker Language 9](#_Toc154341026)

[Slika 6‑1 Kreiranje projekta Star Wars. 13](#_Toc154341027)

[Slika 6‑2 „Sprajtovi“ Star Wars-a. 14](#_Toc154341028)

[Slika 6‑3 Keriranje objekta obj\_player. 14](#_Toc154341029)

[Slika 6‑4 Odabiranje prostora. 15](#_Toc154341030)

[Slika 6‑5 Objekti u prostoru. 15](#_Toc154341031)

[Slika 6‑6 Add Events. 16](#_Toc154341032)

[Slika 6‑7 Step Event prozor 16](#_Toc154341033)

[Slika 6‑8 Dodavanje mogućnosti kretanja objekta obj\_player. 17](#_Toc154341034)

[Slika 6‑9 Dodavanje mogućnosti rotiranja objekta obj\_player. 17](#_Toc154341035)

[Slika 6‑10 Dodavanje događaja Wrap. 18](#_Toc154341036)

[Slika 6‑11 Kreiranje instance objekta obj\_bullet. 19](#_Toc154341037)

[Slika 6‑12 Prozor s događajima objekta obj\_bullets. 19](#_Toc154341038)

[Slika 6‑13 Assign Variable komponenta. 19](#_Toc154341039)

[Slika 6‑14 Kreiranje prozora Outside Room. 20](#_Toc154341040)

[Slika 6‑15 Komponenta Destroy Instance. 20](#_Toc154341041)

[Slika 6‑16 Dodavanje komponenti u Create prozor objekta obj\_rock. 21](#_Toc154341042)

[Slika 6‑17 Dodavanje Step Event-a. 21](#_Toc154341043)

[Slika 6‑18 Komponente Step Event prozora. 21](#_Toc154341044)

[Slika 6‑19 Dodavanje komponenti u Collision prozor. 22](#_Toc154341045)

[Slika 6‑20 Pretvaranje velikih stijena u manja i manjih u velika. 23](#_Toc154341046)

[Slika 6‑21 Dodavanje komponenti u Collision prozor. 24](#_Toc154341047)

[Slika 6‑22 Kreiranje Alarm 0 prozora. 25](#_Toc154341048)

[Slika 6‑23 Komponenta Restart Room. 25](#_Toc154341049)

[Slika 6‑24 Dodavanje instance objekta obj\_game. 25](#_Toc154341050)

[Slika 6‑25 Dodavanje događaja. 26](#_Toc154341051)

[Slika 6‑26 Varijabla „points“. 26](#_Toc154341052)

[Slika 6‑27 Stvaranje Draw GUI prozora. 26](#_Toc154341053)

[Slika 6‑28 Dodavanje komponente u Draw GUI prozor. 27](#_Toc154341054)

[Slika 6‑29 Otvaranje postojećeg Collision događaja. 27](#_Toc154341055)

[Slika 6‑30 Dodavanje komponente varijable u Collision prozor. 27](#_Toc154341056)

[Slika 6‑31 Pokrenuta igrica Star Wars 28](#_Toc154341057)

[Slika 6‑32 Pucanje u igrici 28](#_Toc154341058)

[Slika 6‑33 Razbijanje stijene na dvije manje 29](#_Toc154341059)

[Slika 6‑34 Star Wars igrica 29](#_Toc154341060)

Popis oznaka i kratica

GMS GameMaker Studio

Mac Macintosh

iOS iPhone Operating System

GML GameMaker Language

IDE Integrated Development Environment

SAŽETAK I KLJUČNE RIJEČI

**Sažetak**

Seminarski rad obrađuje temu *GameMaker* *Studio* *2.* GMS 2 jednostavni je alat za izradu 2D video igara. Dominanatan je u odnosu na druge alate za izradu igara radi jednostavnosti korištenja i korisnički prijateljskog sučelja. U seminarskom radu definirane su osnovne i napredne značajke GMS-a kao i procesi usavršavanja aplikacije što obuhvaća optimizaciju, testiranje i implementaciju. U poglavlju šest opisana je izrada igrice Star Wars u GMS-u. To je jednostavna 2D igrica u kojoj korisnik upravlja svemirskim brodom, koji je predstavljen strelicom, te uništava svemirske stijene, koje su predstavljene slikama stijena. Zadnje poglavlje, zaključak, sumira čitavi seminarski rad, te objedinjuje sve navedeno.

**Ključne riječi**

*Android, Asset Browser, GML, GMS 2, Mac, iOS*

**Abstract**

This seminar paper explores the topic of GameMaker Studio 2. GMS 2 is a simple tool for creating 2D video games. It dominates other game development tools due to its ease of use and user-friendly interface. In this seminar paper, the basic and advanced features of GMS are defined, along with the processes of refining the application, which includes optimization, testing, and implementation. In the sixth chapter, the creation of the Star Wars game in GMS is described. It is a straightforward 2D game in which the user controls a spaceship, represented by an arrow, and destroys space rocks, represented as images of rocks. The final chapter, the conclusion, summarizes the entire seminar paper and consolidates all the information provided.

**Keywords**

*Android, Asset Browser, GML, GMS 2, Mac, iOS*