

Programi za izradu računalnih obrazovnih igara

Štimac, Edvard

Master's thesis / Diplomski rad

2020

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Rijeka / Sveučilište u Rijeci**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:195:969411>

Rights / Prava: [In copyright](#)/[Zaštićeno autorskim pravom.](#)

Download date / Datum preuzimanja: **2024-08-08**



Repository / Repozitorij:

[Repository of the University of Rijeka, Faculty of Informatics and Digital Technologies - INFORI Repository](#)



Sveučilište u Rijeci – Odjel za informatiku

Informacijski i komunikacijski sustavi

Edvard Štimac

Programi za izradu računalnih obrazovnih igara

Diplomski rad

Mentor: prof. dr. sc. Nataša Hoić-Božić

Rijeka, srpanj 2020.

Rijeka, 3.6.2019.

Zadatak za diplomski rad

Pristupnik: Edvard Štimac

Naziv diplomskog rada: Programi za izradu računalnih obrazovnih igara

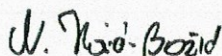
Naziv diplomskog rada na eng. jeziku: Software for educational digital games development

Sadržaj zadatka:

Game Based Learning (GBL) ili učenje zasnovano na igri je proces kojim se znanje usvaja na zabavniji i zanimljiviji način putem različitih igara. Veliki potencijal za GBL danas imaju računalne igre koje su vrlo snažan aspekt zabave za gotovo sve naraštaje. Računalne igre za obrazovanje razvijaju se u određenoj softverskoj razvojnoj okolini koja je posebno dizajnirana za izradu igara. Danas imamo na raspolaganju velik broj različitih softvera za razvoj igri. Zadatak ovog diplomskog rada je analizirati i opisati softver kojim se efektivno može kreirati računalna igra u svrhu obrazovanja. Opisat će se postupci i faze kreiranja obrazovne igre te analizirati načini modeliranja, postupci animiranja i ostale bitne funkcije programa. Kao praktični dio kreirati će se obrazovna računalna igra za učenje pomoću odabranog softvera i opisati postupci kreiranja. Igra će biti namijenjena za učenike osnovne škole i obuhvatiti će odabrane nastavne sadržaje iz kurikuluma Informatike za osnovnu školu.

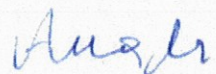
Mentor:

prof. dr. sc. Nataša Hoić-Božić



Voditeljica za diplomske radove:

Izv. prof. dr. sc. Ana Meštrović



Komentor:

Zadatak preuzet: 06.06.2019.

(potpis pristupnika)



Sažetak

Računalne igre za obrazovanje, ali i ostale vrste računalnih igara razvijaju se u određenoj softverskoj razvojnoj okolini koja je posebno dizajnirana za izradu igara. Danas imamo na raspolaganju velik broj različitih softvera u tu svrhu. Svaki softver za izradu igre sastoji se od grafičkog sučelja u kojem se modeliraju, animiraju, povezivanju i ubacuju objekti igre, te svaki program sadrži logiku koja sve to povezuje. Jedan od najpristupačnijih softvera za izradu igara je GDevelop 5.

GDevelop je moderan program namijenjen za brzu i fleksibilnu izradu različitih računalnih igara bez dodatnog znanja programiranja. Grafičko sučelje je vrlo pregledno i pristupačno što omogućuje lakšu navigaciju i upravljanje raznim editorima za izrađivanje scena igre i određivanje drugih bitnih svojstva. Osnovni koncept rada programa je jednostavan i bazira se na nekoliko važnijih čimbenika: objekti, događaji i ponašanja. Dodatni važni čimbenik su varijable, a njihovo korištenje omogućava naprednije manipuliranje igre.

GDevelop 5 se koristio za izradu didaktičke igre namijenjene za obrazovanje. Igra obuhvaća sve važnije čimbenike programa te dodatne mogućnosti koje su ponuđene za bolji izgled i bolji ugođaj. Igra se zove „INFO – Igra za Uvod u Informatiku“ i obrađuje uvodno nastavno gradivo „Osnove Računala“ za 5. razred osnovne škole.

Ključne riječi: GDevelop 5, Game Based Learning, Game Engine, SADDIE model, INFO – Igra za Uvod u Informatiku.

Sadržaj

1.	Uvod.....	7
2.	Učenje zasnovano na računalnim igrama.....	8
3.	Programi za kreiranje igre	9
4.	GDevelop 5	10
4.1.	Kreiranje projekta	11
4.2.	GDevelop Sučelje	12
4.2.1.	Voditelj projekta.....	13
4.2.2.	Editor Scena	15
4.2.3.	Editor Događaja.....	19
4.2.4.	Debugger	21
4.2.5.	Kreiranje računa	22
4.3.	Objekti	22
4.3.1.	Sprite objekt	25
4.3.2.	Tiled Sprite objekt.....	28
4.3.3.	Emiter čestica	29
4.3.4.	Text objekt.....	30
4.3.5.	Panel Sprite objekt.....	31
4.3.6.	Objekt za unos teksta.....	31
4.3.7.	Objekt za crtanje oblika.....	32
4.3.8.	BBText objekt	33
4.4.	Ponašanja	33
4.4.1.	Ponašanje Platforme	34
4.4.2.	Platformer	35
4.4.3.	Uništavanje tijekom izlaska iz scene.....	35
4.4.4.	Vučni objekt	36

4.4.5.	Top-Down kretanje.....	36
4.4.6.	Anchor	37
4.4.7.	Pronalaženje puta	37
4.4.8.	Pathfinding prepreke	37
4.4.9.	Physics Engine 2.0	38
4.4.10.	Tween	40
4.5.	Događaji.....	41
4.5.1.	Standardni događaj	43
4.5.2.	For Each događaj.....	44
4.5.3.	Repeat i while događaji	44
4.5.4.	Povezani događaj.....	45
4.5.5.	Grupiranje događaja	46
4.5.6.	JavaScript Code događaj	47
4.6.	Ostali dodaci programa.....	47
4.6.1.	Napredni uvjeti	47
4.6.2.	Audio	48
4.6.3.	Inventura.....	48
4.6.4.	Timer	49
4.6.5.	Varijable	49
4.6.6.	Dijalog stablo ekstenzija	50
4.7.	Objavljivanje igre	51
5.	Proces izrade igre prema SADDIE modelu.....	54
6.	Projektna dokumentacija igre.....	55
6.1.	Kratak pregled igre: INFO – Igra za Uvod u Informatiku	55
6.2.	Specifikacija.....	56
6.2.1.	Nastavni sadržaj	56

6.2.2.	Učenici kojima je nastavni sadržaj namijenjen	56
6.2.3.	Svrha i ishodi učenja	56
6.2.4.	Oblik i metode rada	57
6.2.5.	Opis priče kojom se prezentira gradivo.....	58
6.3.	Analiza	60
6.3.1.	Okvirni vremenski plan izrade	60
6.3.2.	Okolina za izradu igre i ostala tehnologija.....	60
6.3.3.	Usklađenost obrazovne tematike s kurikulumom	60
6.4.	Scenarij	61
6.4.1.	Popis i skica mape scena	61
6.4.2.	Popis likova	62
6.4.3.	Razrada scenarija.....	64
7.	Zaključak.....	86
8.	Literatura	87
9.	Slike i tablice.....	89

1. Uvod

U današnjem razdoblju računalne igre su dio svakodnevnog života velikom broju ljudi i važan su aspekt zabave i opuštanja za sve naraštaje. Jedna od zanimljivijih namjena računalnih igara je u svrhu edukacije s ciljem da se potakne učenje na zabavan način. [2] Takav način igranja igre danas nazivamo *Game Based Learning* ili učenje temeljeno na igranju igre. Iako se pojam danas najviše koristi za računalne didaktičke igre i njihovo istraživanje, definicija samog *Game Based Learning* pristupa datira i prije konstrukcije prvog računala, te se temelji na općenitim igrama. [7][8]

Za kreiranje računalnih igara danas su na raspolaganju brojni programi s različitim metodama i funkcijama kreiranja. Svaki program sastoji se od grafičkog sučelja u kojem se modeliraju, animiraju, povezivanju i ubacuju objekti igre, te svaki program sadrži logiku koja sve to povezuje. Jedan od najpristupačnijih programa je GDevelop 5, *open-source* program za dizajniranje igara na brz i fleksibilan način. Program je besplatan i sadrži velik broj dodataka i funkcija bez potrebnog dodatnog znanja programiranja. Podržava više platformi poput Windows, Mac i Linux, te se koristi kao edukacijski uvod za kreiranje video igre u mnogim školama i sveučilištima. [13]

Zadatak ovog diplomskog rada je analizirati i opisati softver GDevelop 5 kojim se efektivno može kreirati računalna igra u svrhu obrazovanja. Opisat će se na koji način program radi, kako koristiti razne dodatke koje nudi, te kako kreirati logiku igre pomoću dostupnih editora u programu. Nakon toga, kao drugi dio ovog rada, pomoću analiziranog programa izradit će se video igra namijenjena za edukacijske svrhe. Igra će obuhvaćati sve važnije čimbenike programa koje smo analizirali, te će na osnovni način prikazati rad programa. Igra će biti namijenjena za učenike osnovne škole i obuhvatit odabrane nastavne sadržaje iz kurikuluma Informatike za osnovnu školu.

2. Učenje zasnovano na računalnim igrama

Game Based Learning (GBL) ili učenje zasnovano na igri je proces kojim se znanje usvaja na zabavniji i zanimljiviji način putem različitih igara. Ovaj trend prikupljanja znanja putem igre može se implementirati na mnogim područjima poput edukacije, tijekom osposobljavanja za radno mjesto, tijekom korištenja aplikacija na društvenim medijima i slično. Mnogi ljudi su izloženi učenju zasnovanom na igri na različite načine, ali najveći potencijal tog trenda pripisuje se računalnim igrama koje su danas vrlo snažan aspekt zabave za gotovo sve naraštaje. [7][8][12]

Postoji velik broj različitih tipova računalnih igara koje obuhvaćaju sve moguće žanrove, i koje su namijenjene za različite svrhe. Jedna od zanimljivijih svrha kreiranja računalne igre je za edukaciju. U današnjem razdoblju, eksponencijalno brzi razvoj tehnologije računala i njegovih mogućnosti, potiče sve veći razvoj novih igara koje svi naraštaji konzumiraju na svakodnevnoj bazi. Taj rast tehnologije i sve veće korištenje video igara za zabavu, pogotovo od strane učenika u školama i djece, potiče developere na razvoj računalnih igara namijenjenih za edukaciju. Tako *Game Based Learning* dobiva određenu važnost i počinje se istraživati nove potencijalne inovacije za daljnji razvoj procesa obrazovanja i edukacije. [7][8][12]

Danas postoje mnoga istraživanja na toj tematici. U procesu istraživanja i eksperimentiranja implementiraju se različite edukativne igre u nastavne procese te se mjeri sveukupni utjecaj GBL računalnih igara na obrazovnu performansu. Velik broj istraživanja prikazao je da računalne igre namijenjene za edukaciju pozitivno djeluju na učenike i njihovu sposobnost rješavanja problema, prikupljanja novih informacija, te na motivaciju i sudjelovanje u radu. S druge strane dio istraživanja nema još konkretno definirane rezultate i na njima se pokazalo da edukativne igre nemaju veliki utjecaj na uspjeh u obrazovanju. Stoga se dalje analiziraju uspješni rezultati istraživanja, i preispituje se da li ta uspješnost može biti održiva. [1][7][8][12]

Game Based Learning je tema koja se još istražuje i analizira. Istraživanja ukazuju da učenje temeljeno na igrama može poboljšati motivaciju i sudjelovanje, ali nije još konkretno dokazano da igre poboljšavaju postignućima učenika iz tog razloga nastavnici se kurikulumi još ne oslanjaju na *Game Based Learning* metode. [7][8][12]

3. Programi za kreiranje igre

Računalne igre za obrazovanje, ali i ostale vrste računalnih igara razvijaju se u određenoj softverskoj razvojnoj okolini koja je posebno dizajnirana za izradu igara. Danas imamo na raspolaganju velik broj različitih softvera u tu svrhu. Svaki softver koristi određeni *game engine* odnosno razvojnu okolinu s određenom arhitekturom i grafički korisničkim sučeljem. Svaki *game engine* ima svoju metodu razvoja igre, način kreiranja likova i objekata, način povezivanja interakcije likova i objekata s određenim akcijama, način ubacivanja likova i objekata u scenu, te implementiranje određenih animacija i drugih akcija koje se izvode na ekranu. Kako bi pomoću softvera uspješno i efikasno razvijali igru, unutar svakog softvera nalaze se editori za različite namijene, od kreiranja likova i objekata do programiranja tijeka igre i akcija koje će se izvršavati pod određenim uvjetima. [2][13]

Isto tako, softver se može koristiti određenim programskim jezikom poput: Python, C++, C, Java, JavaScript i slično. Korištenjem programskog jezika kreira se logika igre odnosno način na koji će igra funkcionirati, kako će se ponašati objekti i kako će se izvršavati interakcija s objektima. Stoga kod velikog broja programa potrebno je napredno znanje u programiranju te potrebno je imati dobru vještinu korištenja određenog programskog jezika. Ali kako tehnologija sve više napreduje razvijaju se noviji programi koji ne zahtijevaju napredno znanje programiranja, te se igra može razviti i bez programiranja i pisanja skripti. Takvi programi imaju naprednu arhitekturu koja zaobilazi pisanje koda pomoću sučelja gdje se sintaksa koda interpretira u naredbe, simbole i vizualne predloške koje svatko može koristiti. Tako je svakome omogućena izrada računalne igre, pa i samim nastavnicima koji mogu na brz i efikasan način kreirati računalnu didaktičku igru. Glavne vrline koje se traže tijekom korištenja takvih programa su sama volja za izradu, te dobra vizija i velik broj ideja. [2][13]

Danas, najpoznatiji programi za izradu igre su: Unreal Engine, Unity, Construct, Godot, Game Maker Studio, Adventure Game Studio, Quest, RPG Maker, eAdventure, GDevelop i mnogi drugi. Svaki od tih programa je različit i nudi velik broj zanimljivih funkcija te mogućnost kreiranja edukativnih igara. U ovom radu zadatak je analizirati jedan od programa, prikazati i analizirati sve njegove mogućnosti, te na kraju kreirati igru namijenjenu za obrazovanje.

4. GDevelop 5

GDevelop je open-source program za dizajniranje igara. Program je brz i fleksibilan te pristupačan gotovo svakome sa željom izrađivanja 2D video igara različitih žanrova bez dodatnog znanja programiranja. Podržava više platformi poput Windows, Mac i Linux, te se koristi kao edukacijski uvod za kreiranje video igre u mnogim školama i sveučilištima. [13]

Osnovni koncept rada programa je jednostavan i bazira se na nekoliko važnijih čimbenika. Sve što je predstavljeno na prozoru unutar sučelja definirano je kao objekt ili instanca. Različiti objekti predstavljaju različite elemente koji se mogu manipulirati. Pozicioniranje objekta na ekranu je važan dio kreiranja i izvršava se korištenjem XY koordinatama. Koordinate predstavljaju horizontalni i vertikalni položaj objekta na ekranu. Sljedeći važan čimbenik programa su *Events* odnosno događaji kojim određujemo pravila igre te na koji način će objekti izvršavati međusobnu interakciju i interakciju s igračem. Događaji se sastoje od uvjeta i akcija (*conditions* i *actions*). Uvjeti pokreću test nad objektima te definiraju uvjet aktivacije određene akcije nad objektima. S druge strane Akcije manipuliraju objekte npr. mogu promijeniti položaj objekta, njihov izgled, i slično. Kada se izvršavaju *Eventi* bitan je njihov redoslijed. Događaji na vrhu liste se prvi izvršavaju, a pokreću se svaki puta kada pokrenemo igru. Sljedeći bitan čimbenik programa su *Behaviours* odnosno ponašanja. To su predefinirana pravila koja se mogu ubaciti u objekte, te se time objekti nadograđuju određenom logikom. GDevelop raspolaže velikim brojem već definiranih ponašanja za objekte, ali također omogućuje izradu vlastitih ponašanja što developeri programa i preporučuju nakon što ovladate svim čimbenicima i vaša igra se počinje znatno širiti. Posljednji važniji čimbenik odnosi se na korištenje varijabli. Kao što znamo varijabla omogućuje spremanje podataka (broj ili tekst). GDevelop koristi varijable u različite svrhe kao npr. broj života koji vaš lik ima, broj bodova koji se sakupi odnosno *HighScore*, broj preostalih metaka, broj uništenih protivnika i slično. Sve navedeno može biti spremljeno u varijablu, a akcije i uvjeti mogu manipulirati i mijenjati spremljene podatke. Veličina varijable određuje mjesto kojem se varijabli može pristupiti. U programu dostupni su tri veličine varijabli: globalna varijabla, varijabla scene i varijabla objekta. [13]

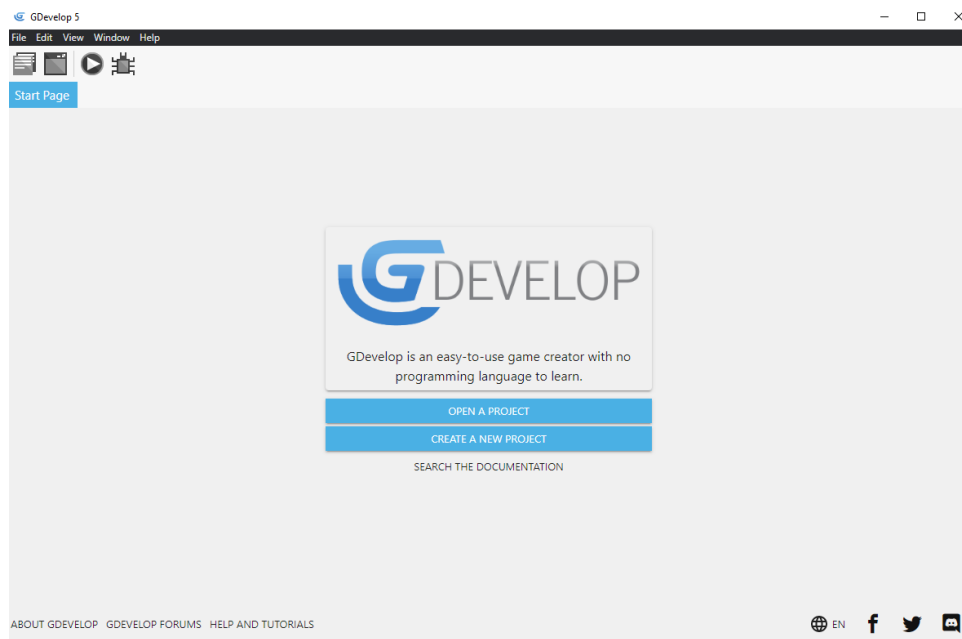
Navedeni čimbenici određuju osnovni koncept kako program radi. Na službenoj stranici programa ponuđen je velik broj tutorijala i vježbi koji su pristupačni svima i služe kao uvod u „umjetnost“ izrade video igara. Uz vježbe ponuđen je i velik broj gotovih primjera s jednom scenom u kojoj je analiziran određen dio programa. Svaka igra se može

preuzeti, manipulirati i prilagoditi autorovim željama, te svaka igra prikazuje i objašnjava određene mogućnosti koje program nudi i koji se svi čimbenici koriste tijekom izrade. [13]

4.1. Kreiranje projekta

GDevelop se može koristiti kao online program ili se može preuzeti desktop verzija sa službene stranice. Program podržava više platformi te se na stranici nalaze linkovi za Windows, MacOS i Linux. Trenutna verzija programa je GDevelop 5, ali na stranici je ponuđena i prijašnja verzija GDevelop 4 za preuzimanje i slobodno korištenje. [13]

Pokretanjem programa otvara se početna stranica iz koje možemo započeti s kreiranjem novog projekta klikom na gumb *CREATE A NEW PROJECT*, ili pokrenuti postojeće projekte na kojima smo radili prije, klikom na gumb *OPEN A PROJECT*. [13]

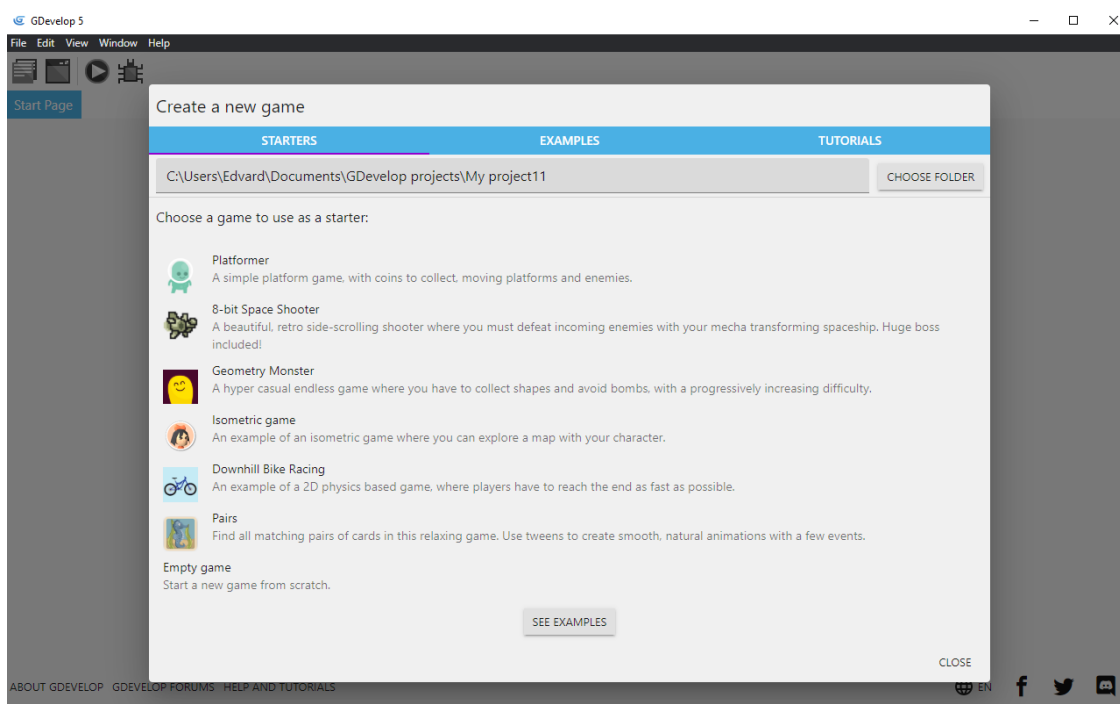


Slika 1 Početna stranica GDevelop 5

Klikom na gumb za kreiranje novog projekta otvara se novi prozor s više različitih opcija pomoću kojih možemo birati određeni predložak ili primjer na kojem će se naša igra bazirati. Na vrhu prozora su tri gumba koji nam nude određene opcije. Prvi gumb *STARTERS* daje nam listu osnovnih igara, odnosno gotovih predložaka na temelju kojih želimo bazirati našu igru. Svaka od ponuđenih igara koristi određenu mehaniku i logiku, te određene

možnosti koje GDevelop nudi. Svaka igra može se mijenjati i modificirati prema našim željama, i dobar su početak za proučavanje složenijih igara. [13]

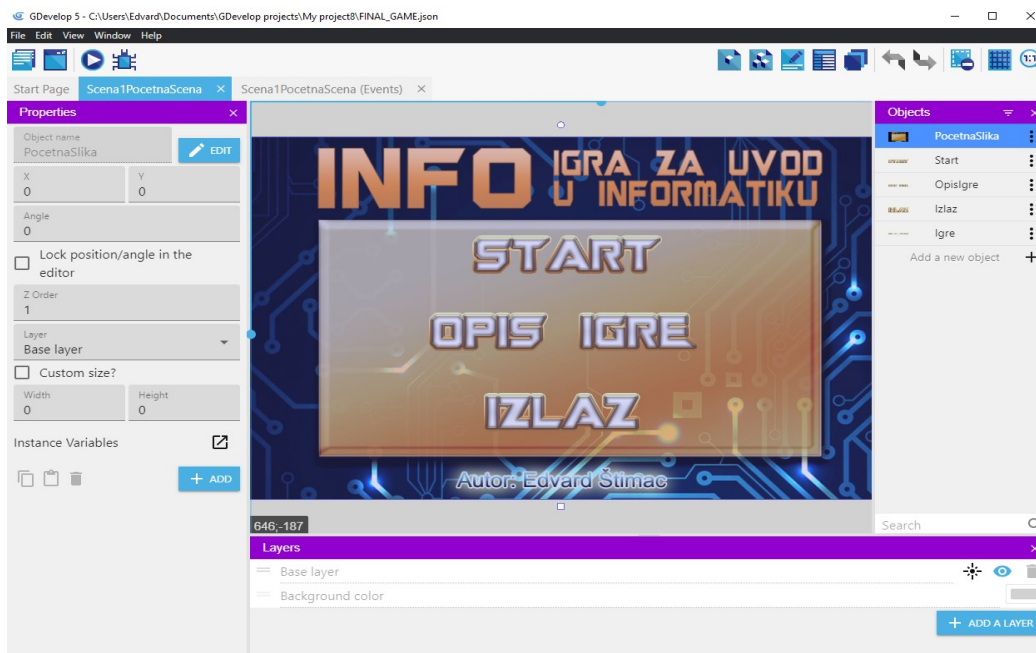
Gumb *EXAMPLES* otvara nam prozor u kojem su izlistane sve mogućnosti i dodaci programa, kako se njima koristiti i kako ih implementirati u igru. Posljednji gumb *TUTORIALS* otvara prozor koji služi kao vodič za kreiranje video igre pomoću određenih mogućnosti koje program nudi. Prozor nudi dokumentaciju o samom programu te o procedurama izrade određenih igara. Svaki od ponuđenih *tutorijala* povezan je sa službenom stranicom i zajedničkim forumom GDevelop programa. [13]



Slika 2 Kreiranje Nove Igre

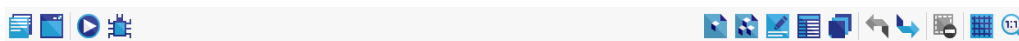
4.2. GDevelop Sučelje

Radno sučelje programa bazirano je na editorima koji su prikazani i razmješteni oko centralnog prozora gdje se prikazuje igra. Svaki od tih editora omogućava modificiranje i izmjenu dijelova igre. Radno sučelje se sastoji od sljedećih dijelova za modifikaciju: *Project Manager* (voditelj projekta), *Scene editor* (editor scena) i *Events editor* (editor događaja). [13]



Slika 3 Radno Sučelje

Na vrhu sučelja nalazi se alatna traka koja sadrži određena svojstva, dodatne editore, debugger, gumb za otvaranje voditelja projekta i druge dodatke. Na lijevoj strani alatne trake nalazi se (od lijeva na desno) gumb za otvaranje voditelja projekta, export gumb za eksportiranje kreirane igre, gumb za pokretanje igre i debugger. Na desnoj strani nalaze se (od lijeva na desno) gumb koji otvara editor objekta, gumb za otvaranje editor grupnih objekata, gumb za otvaranje svojstva objekata na sceni, gumb za otvaranje liste instanci, gumb za otvaranje editora slojeva igre, zatim gumbovi za upravljanje izmjenama (strelice). Ostali gumbovi služe za postavljanje mreže na scenu, te za povećavanje i smanjenje scene. [13]

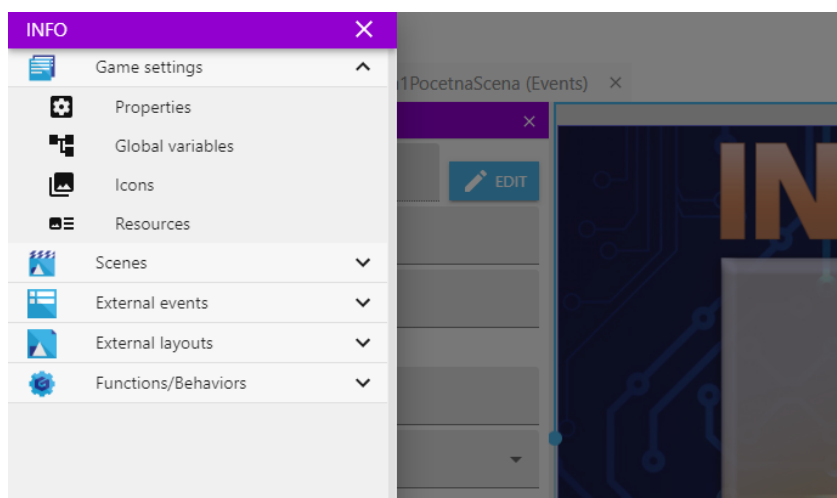


Slika 4 Alatna traka

4.2.1. Voditelj projekta

Voditelj projekta ili *Project Manager* prikazuje strukturu igre, odnosno strukturu trenutno otvorenog projekta. Korištenjem Voditelja možemo dodavati nove elemente za

igru. Klikom na ikonu u gornjem lijevom kutu alatne trake, otvara nam se kartica podijeljena u kategorije, na kojoj se prikazuje struktura cijele igre. [13]



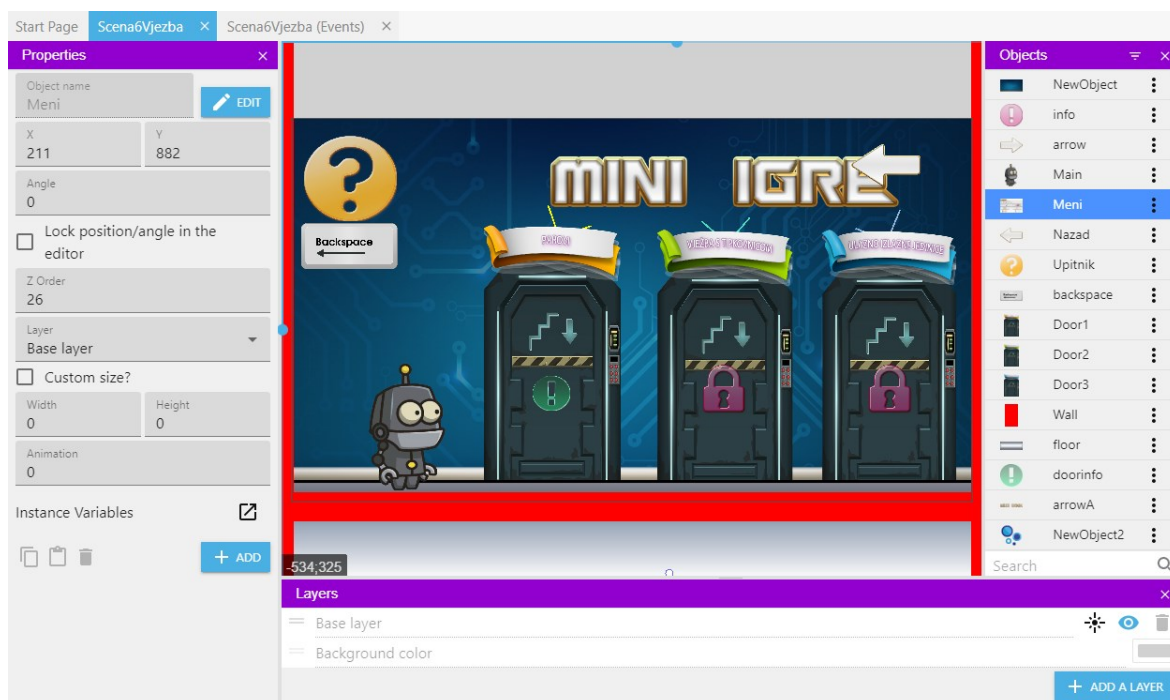
Slika 5 Voditelj projekta (Project Manager)

Klikom na strelicu svake kategorije, otvaraju se dodatne opcije pomoću kojih možemo modificirati igru. U prvoj kategoriji, Postavke igre (*Game Settings* na eng. jeziku), možemo prilagođavati i izmjenjivati globalne postavke igre kao npr. odrediti osnovnu veličinu prozora igre te druga određena svojstva, odrediti naziv igre i odabrati desktop ikonu igre, dodavanje globalnih varijabli i ubacivanje vanjskih resursa. U drugoj kategoriji pod nazivom Scene, nalazi se popis svih scena kreiranih za igru i klikom na jednu od njih otvara se prozor namijenjen za tu određenu scenu. Sljedeća kategorija se odnosi na Vanjske događaje (*External events* na eng. jeziku). Kako bi omogućio fleksibilniji i brži razvoj igre mogu se kreirati posebni događaji koji nisu vezani za jednu određenu scenu, već se mogu gotovi povezati s više scena kako bi se spriječilo nepotrebno ponovno kreiranje istog događaja. Ti posebni događaji izlistani su u kategoriji Vanjski događaji. [13]

Sljedeća kategorija Voditelja projekta su Vanjski slojevi (*External layouts* na eng. jeziku). Isto kao i kod Vanjskih događaja mogu se kreirati i spremati određeni dijelovi scene koji se zatim mogu ponovno pozvati u drugim scenama. Posljednja kategorija su Funkcije/Ekstenzije (*Functions/Extensions* na eng. jeziku) gdje možemo dodavati vlastita prilagođena ponašanja (*behaviours*), nove akcije i izraze. [13]

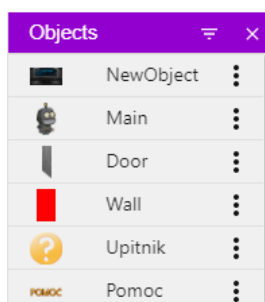
4.2.2. Editor Scena

Kada odaberemo željenu scenu za modificiranje ili nadograđivanje, otvori nam se editor scena. Pomoću tog editora možemo dizajnirati i graditi scenu igre, različite nivoe igre, meni igre, inventure i mnogo drugih dijelova određene scene. [13]



Slika 6 Editor scena

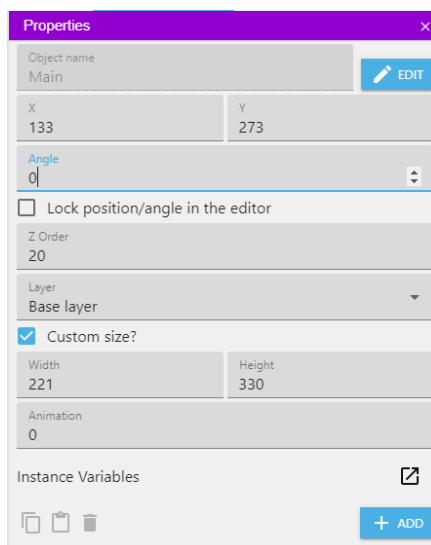
Editor scena se sastoji od nekoliko dijelova koji su od početka vidljivi na sučelju ili im možemo pristupiti pomoću alatne trake. Na desnoj strani sučelja editora scena nalazi se editor objekta koji sadrži listu svih objekata odabrane scene. Na editor objekta mogu se dodavati novi objekti za korištenje, ali isto tako mogu se i izbrisati, te modificirati. [13]



Slika 7 Editor objekta

Objekte također možemo grupirati u Editoru grupe objekata. Editor otvaramo klikom na određeni gumb alatne trake i otvara se s desne strane editora objekta. U editoru grupe objekata nalazi se lista grupa objekata koje se pojavljuju u sceni. Grupe objekata sadržavaju veći broj objekata i mogu se koristiti u jednom događaju koji će utjecati na sve objekte unutar te određene grupe. [13]

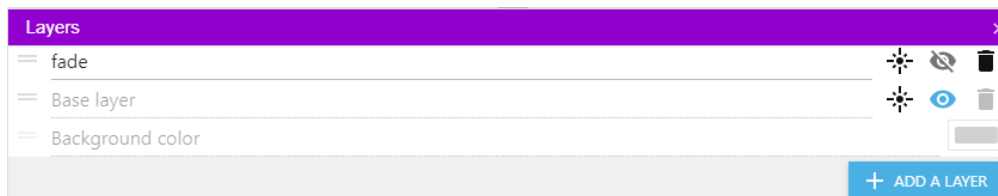
Na lijevoj strani sučelja nalazi se panel za svojstva instance odnosno svih objekata koji se nalaze u sceni. Klikom na objekt koji se nalazi u sceni, panel prikazuje sva svojstva tog objekta. Objekt kojeg stavimo u scenu nazivamo instancom i panel za svojstva instanci omogućava nam izmjenu određenih svojstva odabrane instance. U svojstva koja možemo manipulirati ubrajamo koordinate instance odnosno položaj objekta u sceni, njegov Z-položaj, kut, veličina instance, njezin naziv i sloj u kojem pripada. Važno je napomenuti kako u GDevelop programu razlikujemo termine objekt i instanca. Objekt u GDevelop programu je nacrt ili shematski plan kojeg dodajemo u editor objekta, dok svaki objekt koji je uveden u scenu nazivamo instancom. [13]



Slika 8 Panel za svojstva instance

U donjem dijelu sučelja nalaze se Editor slojeva i Lista instanci odnosno editor instanci. Lista instanci prikazuje listu svih objekata koji se nalaze u odabranoj sceni. Pomoću liste možemo na brz način odabrati bilo koji objekt u sceni. Također na listi su prikazana zadana svojstva svih instanci. [13]

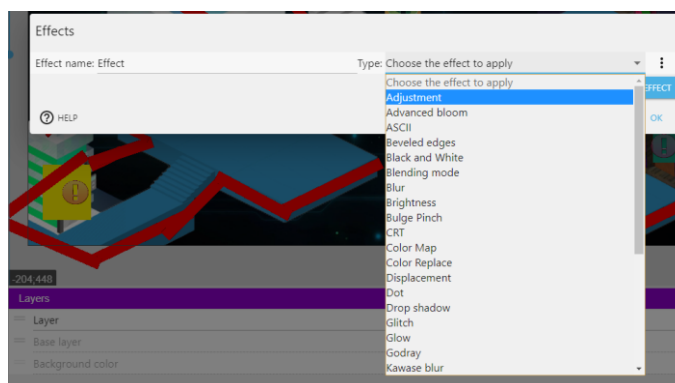
Editor slojeva prikazuje sve slojeve koje se koriste u sceni. Sloj djeluje kao virtualni spremnik za sve objekte prikazane u sceni. Objekte koje uvedemo na scenu pripadaju osnovnom prvom sloju *Base Layer*. Nadalje možemo kreirati više slojeva koji mogu biti prikazani na scenu. Dobar primjer korištenja više slojeva je kod android igrice na dodir, gdje se jedan sloj koristi za prikaz kontrola na ekranu, dok drugi sloj prikazuje pozadinu. [13]



Slika 9 Editor slojeva

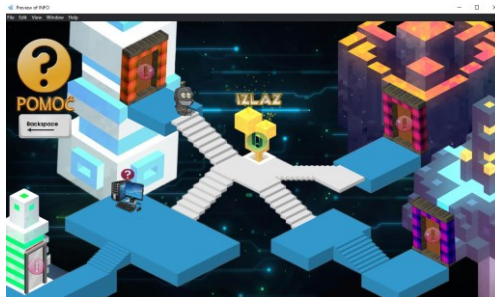
Novi sloj kreiramo klikom na gumb *ADD A LAYER* koji se nalazi na dnu editora slojeva. Slojeve možemo preimenovati, uključivati i isključivati vidljivost klikom na ikonu oka, dodavati efekte klikom na ikonu sunca i izbrisati klikom na ikonu smeća. Vidljivost sloja može se mijenjati pomoću editora sloja ili pomoću događaja (*events*). Ako je sloju isključena vidljivost, sve instance koje se nalaze na tom sloju također nisu vidljive, ali se njime i dalje može manipulirati. [13]

Kao što je već navedeno, željenom sloju može se dodati i određeni efekt. Dodavanje efekta može promijeniti ambijent i atmosferu scene. Klikom na ikonu sunca na desnoj strani editora slojeva otvara nam prozor za dodavanje efekta. Na prozoru možemo odabrati tip efekta te mu dati naziv po vlastitoj želji. [13]

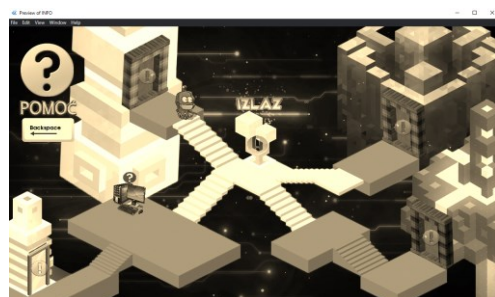


Slika 10 Dodavanje efekta

Odabirom jednog tipa efekta otvara nam se prozor s parametrima tog efekta pomoću kojeg možemo prilagoditi način prezentacije efekta u sceni. [13]

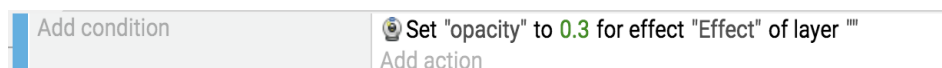


Slika 11 Scena bez sloja s efektom



Slika 12 Scena s efektom na sloju

Parametri efekta se također mogu prilagođavati i mijenjati pomoću događaja (*events*). Prvi parametri koje smo zadali za jedan sloj mogu se automatski izmijeniti tijekom igre kreiranjem događaja s akcijom koja će promijeniti parametre efekta pod određenim uvjetom. [13]

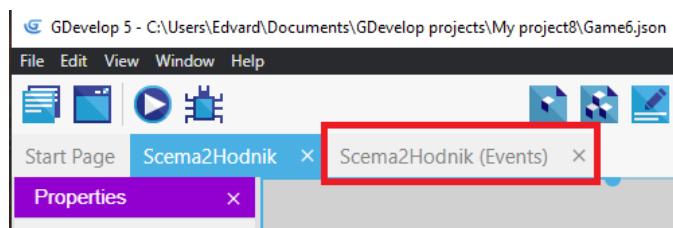


Slika 13 Akcija koja mijenja vidljivost efekta "Efekt" na jednom sloju

Još jedan koristan dodatak za slojeve je korištenje Vanjskih slojeva. Oni se mogu koristiti za izradu izgleda objekata isto kao i editor scena. Objekti se mogu dinamički umetnuti u scenu pomoću akcija u kategoriji Vanjskih slojeva. Cijeli leveli igre mogu se spremiti u različite vanjske slojeve, a ti se slojevi onda mogu pozvati u određenu scenu. [13]

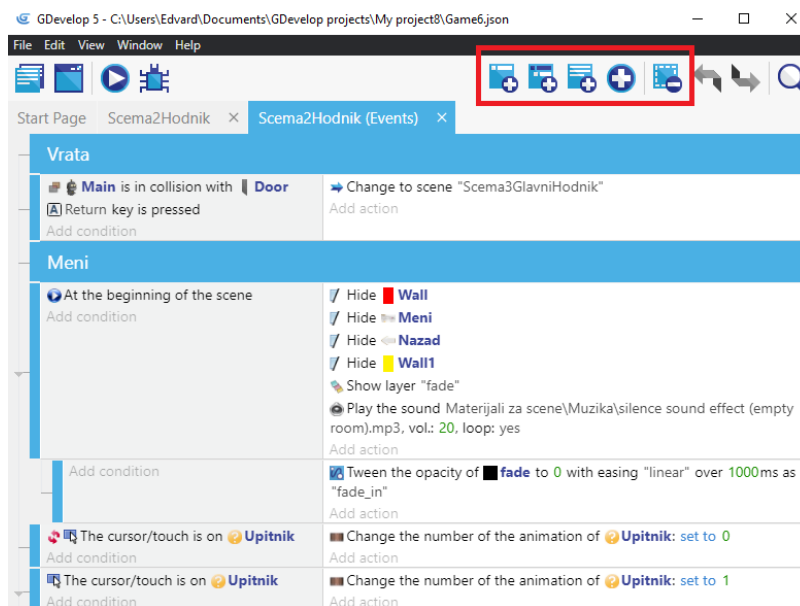
4.2.3. Editor Događaja

Sljedeći važni dio GDevelop 5 sučelja je Editor događaja ili *Events editor*. Prilikom otvaranja jedne scene, ispod alatne trake otvore se tri kartice: *Start Page*, Scena (Editor scene), Događaji scene (Editor događaja). Prva kartica *Start Page* vodi nas na početni prozor gdje možemo kreirati novi projekt. Druga kartica se odnosi na Editor scene gdje prilagođavamo i modificiramo scenu. Posljednja kartica je Editor događaja gdje kreiramo događaje kojim određujemo pravila igre, odnosno na koji način će objekti izvršavati međusobnu interakciju i interakciju s igračem. [13]



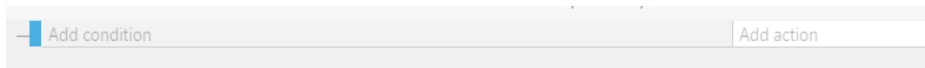
Slika 14 Kartica za Editor događaja

Editor događaja omogućava vizualno programiranje naše igre na brz i jednostavan način bez prethodnog znanja programiranja. Klikom na karticu za Editor događaja, otvara se poseban prozor namijenjen za kreiranje događaja s novom alatnom trakom koja sadrži određene funkcije. [13]



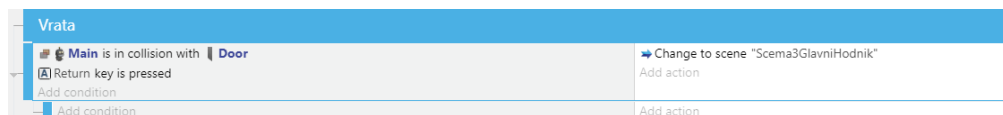
Slika 15 Editor događaja s novim funkcijama na alatnoj traci

Na desnoj strani alatne trake nalaze se novi gumbi pomoću kojih kreiramo događaje i pod-događaje, dodajemo komentare, grupiramo događaje i uvodimo dodatne mogućnosti za određene događaje. Klikom na prvi novi gumb (s lijeva na desno) uvodimo novi prazni događaj u kojem možemo definirati uvjete i akcije koje će utjecati na igru. [13]



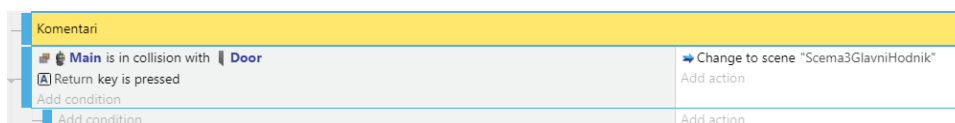
Slika 16 Novi događaj

Pomoću sljedećeg gumba možemo uvesti prazni pod-događaj ili *sub-event* u kojem isto definiramo uvjete i akcije, ali njihova funkcija će nastupiti u sceni samo ako je uvjet događaja odnosno *eventa* tog *sub-eventa* istinit (*True*). Pod-događaji pružaju način kako bi se izbjegle repeticije istih uvjeta. [13]



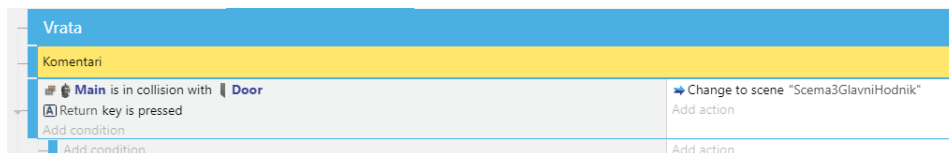
Slika 17 Novi Pod-događaj

Sljedeći gumb se koristi za dodavanje komentara za događaje i pod-događaje. Komentari su vrlo korisni za objašnjavanje složenijih događaja što omogućava pregledniji prozor editora i brže kretanje kroz događaje i brže shvaćanje na koji način rade. [13]



Slika 18 Ubacivanje komentara u događaj

Nakon funkcije ubacivanja komentara, sljedeći gumb koji izgleda na veliki plavi plus otvara nam dodatne mogućnosti za događaje. Gumb nudi funkcije poput grupiranja događaja radi dodatne preglednosti, povezivanje vanjskih događaja, otvaranje terminala za korištenje JavaScript programskog jezika, određivanje broja ponavljanja događaja, ubacivanje dodatnih događaja, te mogućnost korištenja petljama *for* i *while*. [13]

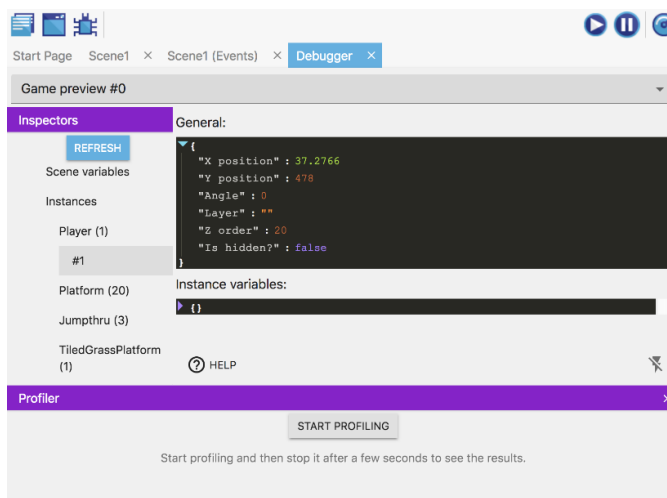


Slika 19 Grupiranje događaja

Osim tih važnijih novih funkcija alatna traka editora događaja ima još gumb za brisanje događaja, gumbove za izmjenu promjena te tražilicu. [13]

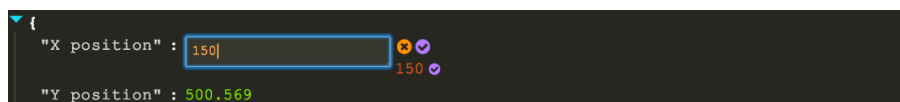
4.2.4. Debugger

Debugger se odnosi na alat koji omogućava provjeravanje sadržaja naše igre. Pomoću alata provjeravaju se globalne varijable, varijable scena i varijable instanci. Tako se istražuje ako dijelovi igre rade kako su i određeni, ali isto tako alat omogućava puno bolje razumijevanje potencijalne greške koja se može dogoditi tijekom pokretanja igre. [13]



Slika 20 Debugger

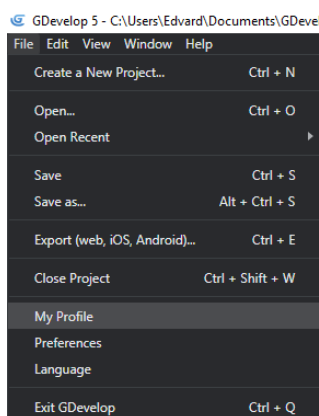
Debugger pokrećemo klikom na gumb u obliku na plavog kukca koji se nalazi na desnoj strani alatne trake. Gumb otvara novu karticu zvanu *Debugger* te istodobno pokreće igru. Time se igra povezuje s debuggerom preko kojeg možemo pregledati svaki dio igre, od pojedine scene do svake instance. Preko debuggera mogu se izmijeniti ili ispraviti dijelovi igre za koje smatramo da zahtijevaju određenu promjenu ili ne rade po zadanom pravilu. [13]



Slika 21 Promjena određenih svojstva instanci korištenjem debugera

4.2.5. Kreiranje računa

GDevelop program nudi velik broj dodataka, te novosti i poboljšanja koje dobije nakon ažuriranja. Kako bi korisnik dobio sve dodatke aplikacije ponuđena je mogućnost kreiranja besplatnog GDevelop računa. Klikom na *File* karticu u lijevom gornjem kutu prozora otvara se panel, te klikom na opciju *My Profile* otvara se prozor za kreiranje računa. [13]

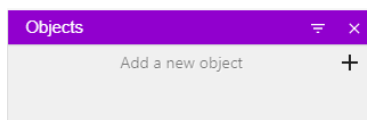


Slika 22 Kreiranje GDevelop računa

Iako je program besplatan dodatno se nudi i pretplata kojom se otvaraju dodatne mogućnosti programa. Pomoću računa mogu se kreirati dvije igre za android u jednom danu, dok s pretplatom broj se povećava ovisno o razini cijene. [13]

4.3. Objekti

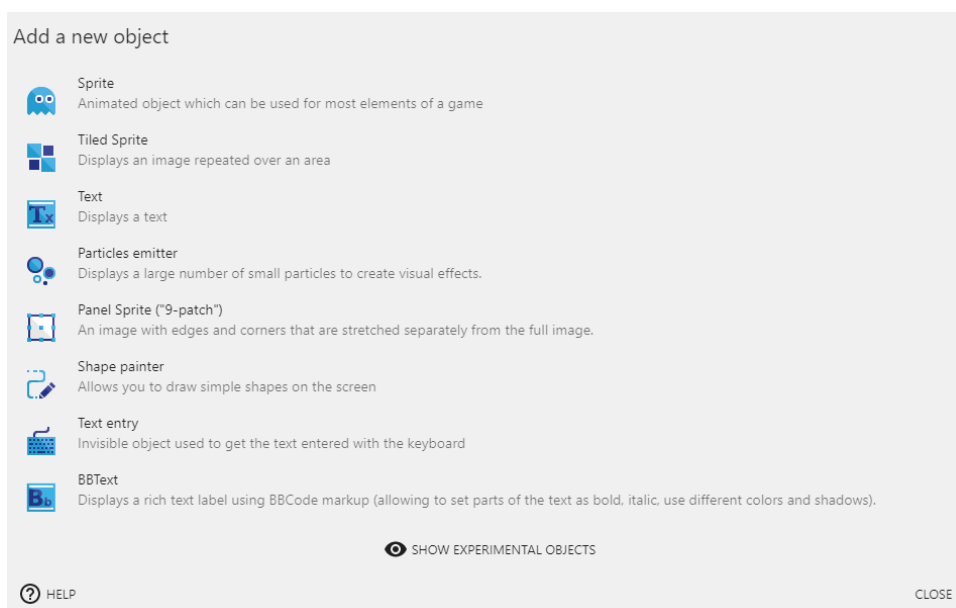
Sve igre kreirane pomoću GDevelop programa bazirane su na objektima predstavljenim u sceni. Sve što vidimo na ekranu je skup različitih objekata s različitim funkcijama namijenjenim za određenu ulogu. Formiranje prvih instanci i samo kreiranje igre započinjemo ubacivanjem objekata u editor objekta. [13]



Slika 23 Ubacivanje objekta u Editor objekta

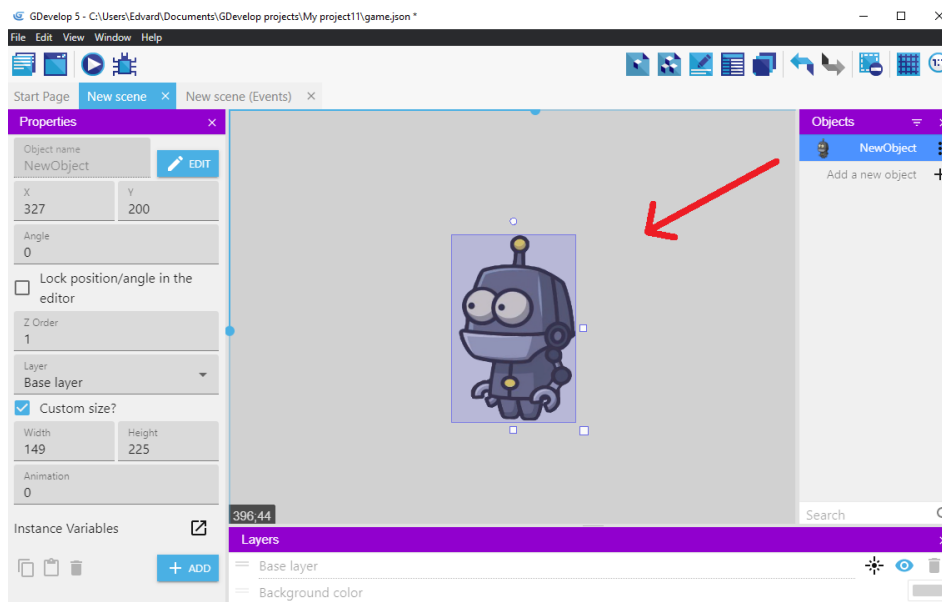
Klikom na gumb *Add a new object*, koji se nalazi na vrhu editora ili na dnu ovisno o tome koliko se objekata nalazi u editoru, otvara nam se prozor gdje koji sadrži opcije za ubacivanje više različitih tipova objekta s različitim svojstvima i specifičnim ulogama. Ovisno o odabranom tipu objekta, editor i svojstva objekta te svi dostupni dodaci u editoru događaja bit će drugačiji. Tipove objekta koje možemo odabrati su sljedeći: [13]

- Sprite – animirani objekt koji se koristi za većinu stvari u igri
- Tiled Sprite – slika koja se prikazuje više puta preko zaslona
- Text – tekstualni objekt
- Particles emitter – animirane čestice koje kreiraju vizualni efekt
- Panel Sprite – slika s određenom prilagođenom veličinom okvira
- Shape painter – mogućnost crtanja jednostavnih oblika na ekranu
- Text entry – nevidljiv objekt koji omogućava unos teksta pomoću tipkovnice
- BBText – tekstualni objekt koji koristi BBCode



Slika 24 Tipovi objekta

Nakon odabira tipa objekta i unosa svih traženih svojstva za odabrani tip, objekt se sprema u editor objekta. Kao što je već prije navedeno, svaki objekt koji je predstavljen na ekranu nazivamo instancom. Instancu dodajemo u scenu jednostavnim *drag & drop* metodom, odnosno selektiranjem željenog objekta i povlačenjem tog objekta iz editora na scenu. [13]



Slika 25 Instanca - Objekt prikazan u sceni

Svaka instanca objekta ima svoje koordinate na sceni, svoje varijable, animaciju, sadržaj, poziciju, i druga svojstva koja su specifična za tu instancu. Svojstva instance prikazana su na panelu *Properties* na lijevoj strani sučelja. Instancu možemo ručno pozicionirati i oblikovati po želji i potrebi. Isto tako instancu možemo modificirati pomoću navedenog panela za svojstva, gdje nam je omogućena izmjena pozicije, veličine instanci, kuta, Z-položaja, sloja kojem instanca pripada, te izmjena i definiranje varijabli instanci. Manipuliranjem Z-položaja instance određujemo raspored prikaza objekata na ekranu. Instanca s najvećom vrijednošću Z-položaja prikazana je prva i ispred svih drugih instanci koje imaju manju vrijednost. Kao što je već prije spomenuto objekte također možemo grupirati pomoću alata grupiranja na alatnoj traci i time možemo omogućiti odvijanje određenog događaja koji će utjecati samo na taj skup objekata koji je grupiran. [13]

Objekti mogu sadržavati i određene varijable koje nazivamo varijablama instanca. Varijabla je „spremnik“ koji sadrži zadanu vrijednost (tekst, broj). Ta vrijednost se može

mijenjati i modificirati tijekom igre. Tijekom pokretanja igre svaka instanca objekta sadržava vlastitu varijablu. Instanci možemo definirati varijablu u panelu za definiranje svojstva instanci. Klikom na željenu instancu objekta otvara se taj panel, te u donjem dijelu panela možemo dodavati i specificirati varijable za tu instancu. [13]



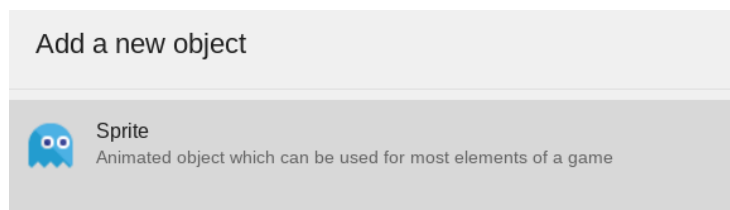
Slika 26 Definiranje varijable instance

Osim varijabli svaki objekt može imati svoj *timer* ili brojač te svaki *timer* je drugačiji za svaku instancu objekta na sceni. Timeri su korisni kada određujemo kada instanca izvršava određenu akciju ili može se odrediti da instanca izvršava akciju svakih X sekundi. [13]

Sva navedena svojstva instanci objekta mogu se također manipulirati i određivati u editoru događaja. Korištenjem događaja, instance mogu mijenjati svoja zadana svojstva ovisno o postavljenom uvjetu. U suštini ovo je glavni dio izrade igre u programu GDevelop i bit dobre, pregledne i funkcionalne igre. U svrhu ovoga rada, u scenama su predstavljene mnoge instance namijenjene za obrazovanje, i njihovom manipulacijom igrač uči o sadržaju gradiva informatike na vizualan i zabavan način. [13]

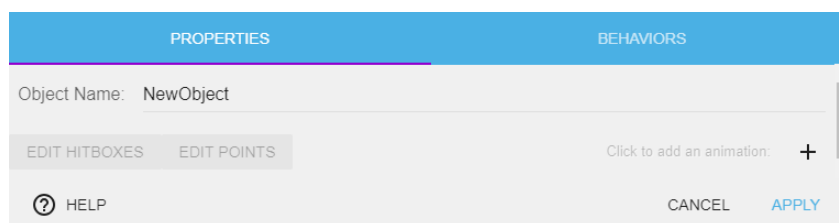
4.3.1. Sprite objekt

Tijekom izrade igre, tip objekta koji je definiran kao *Sprite* koristi se u najviše slučajeva. Sprite objekt nam omogućava prikaz slike ili serije slika koje formiraju određenu animaciju. Svaki željeni objekt od likova, pozadine, meni prozora, do namještaja i ostalih predmeta, može se definirati kao *Sprite*. Kao što je već prije navedeno sprite objekt dodajemo na isti način kao i svaki drugi objekt, klikom na gumb *Add a new object* u editoru objekta. Prozor koji nam se otvori nudi više tipova objekta i prva opcija se odnosi na sprite objekte. [13]



Slika 27 Kreiranje Sprite objekta

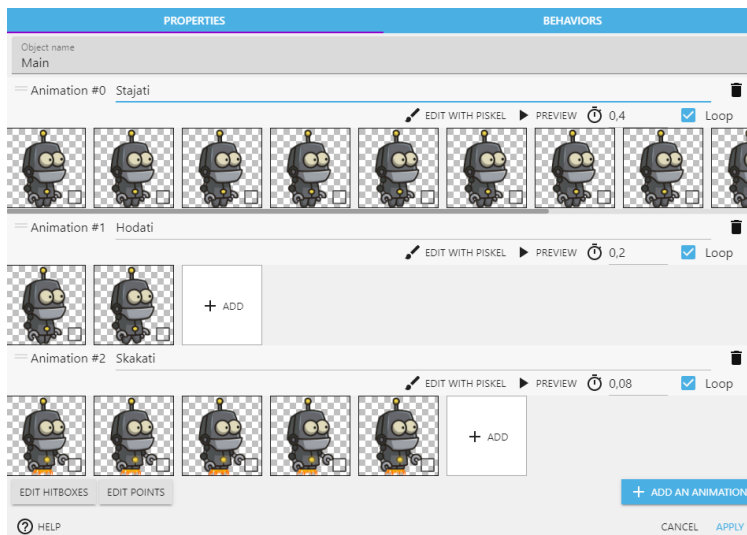
Odabirom sprite objekta otvara nam se prozor za određivanje svojstva. Tu možemo dati naziv našem objektu te ubaciti pripremljene slike koje će vizualno predstavljati objekt na sceni. [13]



Slika 28 Prozor za određivanje svojstva tijekom kreiranja sprite objekta

Klikom na gumb *Click to add an animation* prozor se dodatno otvara i nudi različite opcije pomoću kojih možemo određivati način prezentiranja objekta. Kreiranjem glavnog lika u igru ili NPC lika (*Non-player character* ili lik kojeg ne upravljamo) s kojim glavni lik može započeti interakciju, dodajemo sliku ili seriju slika koje formiraju animaciju za određenu svrhu. [13]

Svaki sprite objekt može imati više vrsta animacija koje se preko editora događaja pozivaju tijekom neke specifične akcije na sceni. U jednu animaciju mogu se dodati više slika kako bi se kreirala više dinamičnija animacija objekta. Animaciji možemo nakon toga odrediti brzinu te zadati mogućnost ponavljanja. [13]

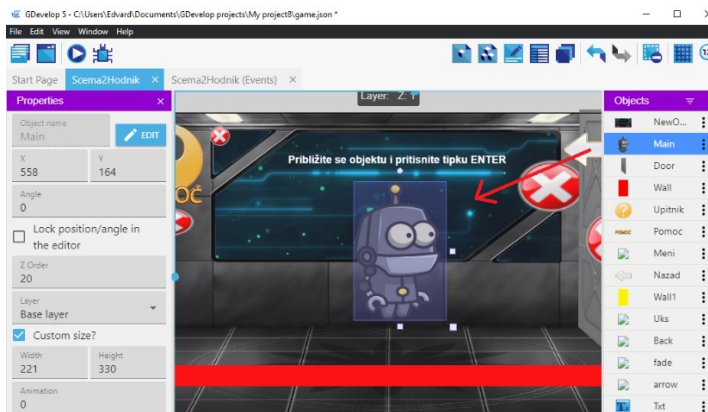


Slika 29 Animiranje objekta

Kada objekt ima više animacija namijenjenih za različite akcije, njih možemo aktivirati kreiranjem događaja u editoru događaja. Pomoću događaja zadaje se uvjet i akcija koje će aktivirati željenu animaciju objekta tijekom određene situacije. [13]

Osim kreiranja animacija, objektu možemo odrediti i točke kolizije koje prikazuju područje objekta koje će označavati kada je objekt u dodiru s drugim objektom. [13]

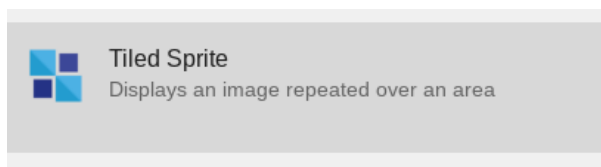
Nakon kreiranja svih potrebnih animacija, sprite objekt se sprema na listu u editoru objekta. Iz te liste objekte možemo ubacivati na scenu igre jednostavnom *drag & drop* metodom. Kao što je već spomenuto prije, kada se objekt odvuče na scenu on tada postaje instancom. [13]



Slika 30 Kreirani Sprite objekt na sceni

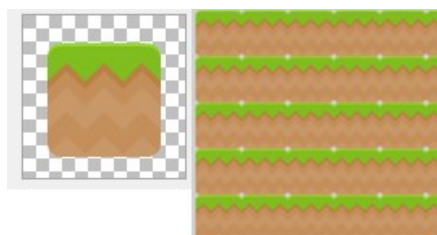
4.3.2. Tiled Sprite objekt

Tiled sprite objekt je drugi tip objekta koji je ponuđen. Njega možemo interpretirati kao sliku koja se prikazuje kao jedna pločica, te širenjem slike na sceni povećava se broj tih pločica. [13]



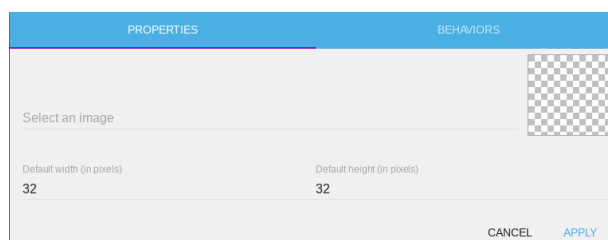
Slika 31 Kreiranje Tiled Sprite objekta

Ovaj tip objekta je vrlo koristan u izradi platformi. Umjesto da se ubacuje više sprite objekta, može se napraviti tiled sprite objekt odnosno pločica koju širenjem na sceni možemo pretvoriti u skup pločica. [13]



Slika 32 Tiled sprite objekt – jedna pločica (lijeva strana) i skup pločica (desna strana) koju dobijemo širenjem jedne pločice

Odabirom ovog tipa objekta otvara se prozor u kojem određujemo svojstva na isti način kao i kod kreiranja sprite objekta. Jedina razlika u ovom slučaju je da su svojstva koja određujemo drugačija od svojstva sprite objekta. Kada određujemo svojstva tiled sprite objekta ponuđena je samo veličina odabrane slike odnosno dužina i širina slike. [13]

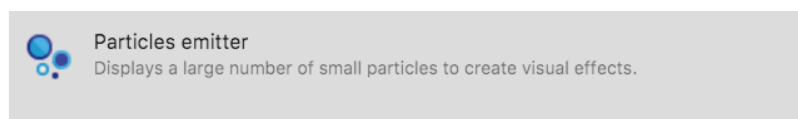


Slika 33 Prozor za određivanje svojstva tijekom kreiranja tiled sprite objekta

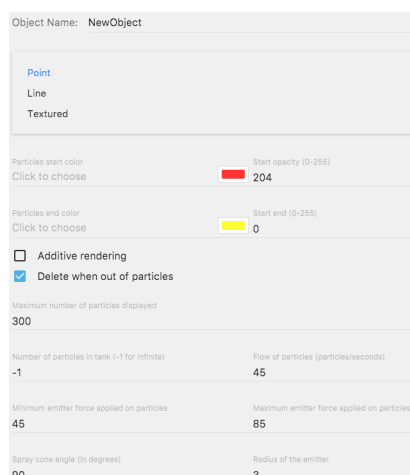
Odabirom slike i njezine veličine, kreira se tiled sprite objekt. Isto kao i svaki tip objekta, on se sprema na listu u editor objekta. Kada tiled sprite objekt odvučemo na scenu on postaje instancom s veličinom koju smo prethodno zadali. Kod instance sprite objekta, kada prilagođavamo njezinu veličinu na sceni, instanca se povećava i prilagođava željenoj veličini koju zadajemo, ali kod instance tiled sprite objekta, izmjena veličine na sceni ima drugačiji efekt. Kada se instanca tiled sprite objekta proširi preko zadane veličine, instanca se ne širi već se kreira identična kopija instance povezana s originalnom instancom. Tako se stvaraju „pločice“ odnosno grupa kopija originalne instance. [13]

4.3.3. Emiter čestica

Sljedeći tip objekta koji možemo kreirati je emiter čestica ili *Particles emitter*. Ovaj tip objekta se koristi za prikazivanje velikog broja animiranih čestica koje simuliraju efekte poput vatre, dima, prašine, kiše, snijega, eksplozije i slično. [13]



Slika 34 Kreiranje emitera partikla

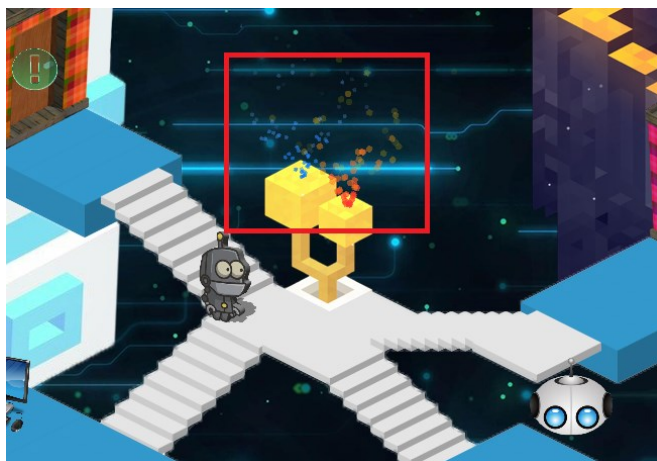


Slika 35 Prozor za određivanje svojstva tijekom kreiranja emitera partikla

Klikom na opciju emitera, otvara se prozor gdje možemo zadati željene parametre i svojstva za naš emiter čestica. Na prozoru možemo birati boju čestica te njihovu veličinu,

broj, jačinu, kut i protok. Isto tako možemo birati i oblik čestica. Na vrhu prozora za svojstva ponuđene su tri opcije oblika: točka, crta i tekstura. Odabirom teksture otvara se dodatna opcija gdje možemo ubaciti vlastiti oblik za čestice. [13]

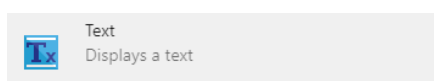
Nakon definiranja svih željenih svojstva za emiter čestica, objekt se sprema na listu u editor objekta. Iz editora ubacujemo emiter na scenu gdje mu možemo ručno prilagoditi veličinu te smjer kretanja čestica. [13]



Slika 36 Emiter čestica u sceni

4.3.4. Text objekt

Text objekt se odnosi na objekt koji je u sceni prikazan u obliku teksta. Isto kao i kod ostalih objekta, možemo prilagođavati i svojstva Text objekta. [13]



Slika 37 Kreiranje Text objekta

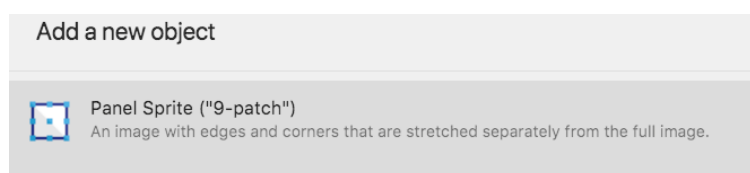
Svojstva koja možemo prilagođavati su: sadržaj teksta koji će objekt prikazivati, veličina i font teksta, stil, boja, gradijent, sijena, vidljivost teksta i skaliranje. Također, svojstva teksta se mogu mijenjati i u editoru događaja gdje se sadržaj objekta može izmijeniti tijekom specifičnog uvjeta. Dobar primjer jer iznos bodova koji lik može prikupiti. Bodovi (tekstualni objekt) se konstantno mijenjaju ovisno o našem napredovanju u igri. [13]



Slika 38 Događaj koji mijenja veličinu fonta Text objekta "Score"

4.3.5. Panel Sprite objekt

Panel Sprite objekt se koristi za prikaz tekstura prema zadanom okviru. Ovaj tip objekta je koristan za konstrukciju objekta koji imaju dinamičnu veličinu. Panel Sprite obuhvaća stvari poput okvira nekog sučelja te platforme. [13]

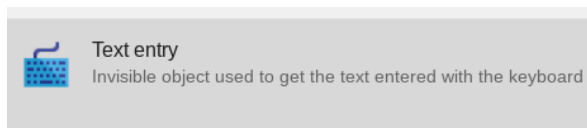


Slika 39 Kreiranje Panel Sprite objekta

Kreiranje i određivanje svojstva se obavlja na isti način kao i kod drugih tipova objekta. Klikom na *Panel Sprite "9-patch"* opciju otvara se prozor gdje unosimo željene parametre za svojstva objekta. [13]

4.3.6. Objekt za unos teksta

Objekt za unos teksta ili *Text entry* je objekt koji sprema unos teksta. Objekt omogućava unos teksta u određeno polje npr. unos imena našeg lika, ili unos točnog odgovora na pitanje. [13]



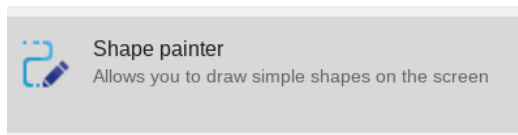
Slika 40 Kreiranje objekta za unos teksta

Na prozoru za svojstva objekta određuje se unos na osnovu naših potreba, te dodavanjem tog objekta na scenu formira se instanca koja od igrača zahtijeva unos teksta

koji se sprema u memoriju. Nakon toga pomoću dodatnih parametara u prozoru svojstva i pomoću editora događaja određuje se akcija koja bi uslijedila nakon unosa teksta. [13]

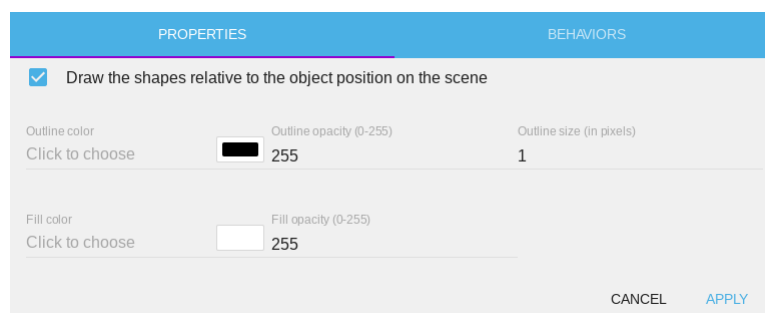
4.3.7. Objekt za crtanje oblika

Zanimljiv tip objekta je objekt koji kreiramo s funkcijom crtanja. Objekt crtanja oblika ili *Shape painter* objekt omogućava nam crtanje različitih oblika u sceni igre. Objekt dodajemo klikom na *Shape painter* na prozoru za selekciju tipa objekta. Zatim na novo otvorenom prozoru za određivanje svojstva objekta zadajemo potrebne parametre. [13]



Slika 41 Kreiranje objekta za crtanje oblika

Parametri koje možemo zadati su boja objekta i vidljivost, the relativni položaj objekta. Nakon odabira svih parametara objekt se sprema na listu u editor objekta. Iz te liste *Shape painter* objekt možemo odvući na scenu gdje postaje instancom objekta za crtanje oblika. Instanca na početku drži samo funkcionalnost crtanja na sceni, stoga ako bi željeli crtati moramo koristiti editor događaja gdje ćemo dodati događaj s akcijom pod nazivom *Primitive drawing*. [13]

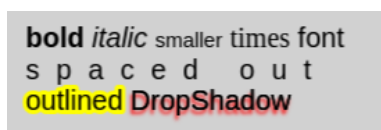


Slika 42 Prozor za određivanje svojstva tijekom kreiranja Shape painter objekta

Akcija Primitive drawing omogućava nam crtanje 3 različitih oblika: krug, crta i pravokutnik. Odabirom jednog od oblika omogućena nam je funkcija crtanja tog oblika tijekom igranja igre. [13]

4.3.8. BBText objekt

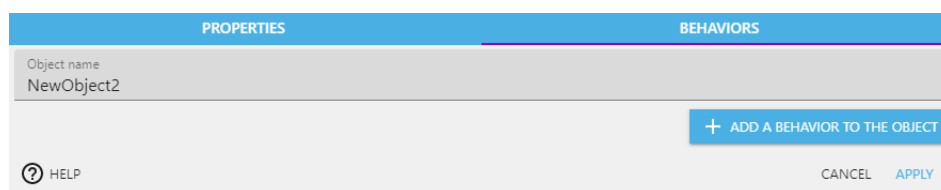
BBText objekt je objekt koji na ekranu prikazuje bogat tekst s više stilova. Tekst se prilagođava pomoću različitih svojstva stilova koristeći BBCode tagovima. Korištenjem događaja, svojstva BBText objekta mogu se modificirati tijekom igranja igre. [13]



Slika 43 BBCode tagovi

4.4. Ponašanja

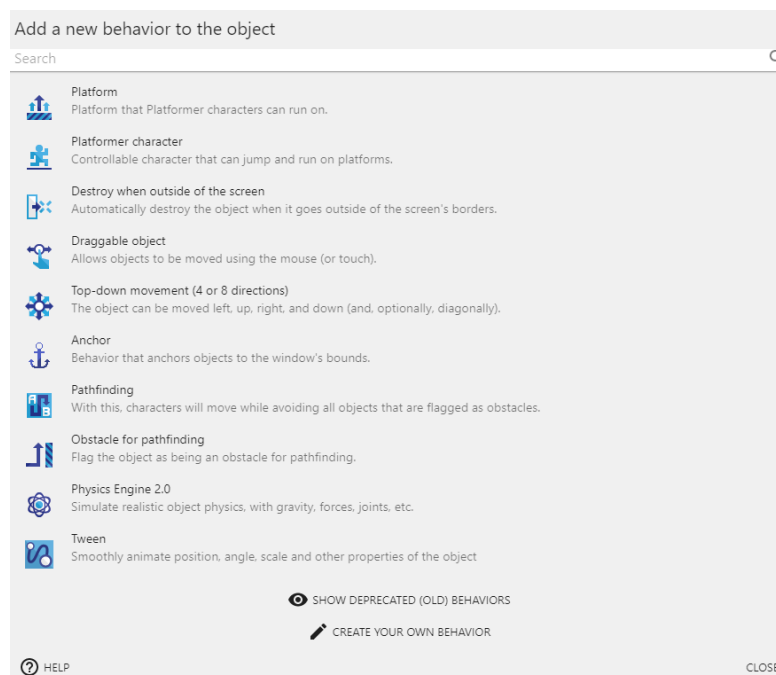
Ponašanja ili *Behaviours* su dodatne sposobnosti koje su omogućene objektima. Svaki kreirani objekt može sadržavati jedno ili više ponašanja. Tijekom kreiranja objekta i odabira željenog tipa, otvara se prozor za određivanje svojstva. Prozor sadržava dvije kartice, jedna namijenjena određivanju svojstva, a druga namijenjena kreiranju ponašanja. [13]



Slika 44 Dodavanje ponašanja u objekt

Klikom na gumb *ADD A BEHAVIOUR TO THE OBJECT* otvara se novi prozor koji nudi velik broj različitih ponašanja. Na prozoru prikazani su glavni tipovi ponašanja objekta, ali omogućena je i opcija kreiranja vlastitih ponašanja ili pretraživanje drugih ponašanja koja

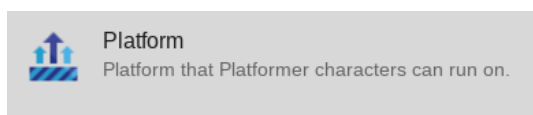
nisu prikazana na prozoru. Osim glavnih zadanih ponašanja, GDevelop sadrži mnoštvo drugih koji obavljaju različite funkcije za gotovo sve situacije. [13]



Slika 45 Zadani tipovi ponašanja objekta

4.4.1. Ponašanje Platforme

Platformsko ponašanje je tip ponašanja koje možemo zadati nekom objektu s kojim će naš lik u igri imati mogućnost kolizije. Taj tip ponašanja daje objektu osobinu platforme na koju likovi mogu stajati ili izvršavati koliziju. Platformsko ponašanje je važna komponenta objekta tijekom stvaranja platformskih igara poput *Super Mario* ili *Icy Tower* gdje likovi mogu skakati na različite platforme. [13]



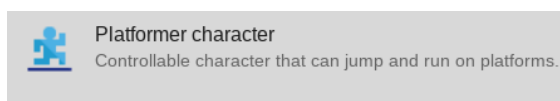
Slika 46 Kreiranje platformskog ponašanja

Odabirom ponašanja platforme, ponuđene su tri različite opcije na koji će objekt kao platforma djelovati. Tipovi platformi su: [13]

- Platforma – osnovno ponašanje platforme gdje se objekt zadaje kao tradicionalna platforma na kojoj lik može stajati i kretati se, te izvesti koliziju.
- Jumphru platforma – platforma na kojoj lik može stajati i kretati se, ali može skočiti kroz platformu od ispod.
- Ljestve – platforma koja djeluje kao ljestve za penjanje gdje lik ne može stajati i ne može izvesti koliziju s platformom, već se samo može penjati kada je u dodiru s ljestvama.

4.4.2. Platformer

Sljedeća opcija ponašanja namijenjena je za objekt koji će izvršavati funkciju lika kojeg upravljamo. Platformer ponašanje daje se svakom kreiranom liku u 2D platformskoj igri gdje lik dobiva funkcije kretanja pomoću zadanih tipki na tipkovnici, te može skakati i izvršavati kolizije s prijašnje određenim platformama. [13]

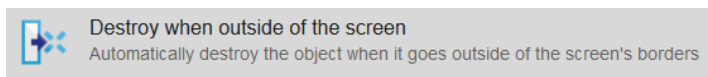


Slika 47 Kreiranje ponašanja platformera

Odabirom ponašanja platformera otvara se dodatni prozor gdje se određuju svojstva kretanja našeg lika, poput brzine kretanja, skakanje, jačina gravitacije, te opcija hvatanja ruba platforme tijekom penjanja ili skakanja. Tipke za kretanje koju lik dobiva su zadane strelice na tipkovnici, te funkciju skakanja dobiva razmaknica tipkovnice ili *Spacebar*. [13]

4.4.3. Uništavanje tijekom izlaska iz scene

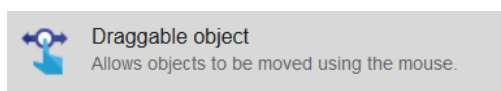
Sljedeće ponašanje koje se može odrediti objektu je ponašanje *Destroy when outside of the screen* ili Uništavanje tijekom izlaska iz scene. Ovo ponašanje je vrlo jednostavno ali istodobno vrlo korisno za velik broj različitih namjena. Objekt na sceni se odmah uništi onog trenutka kada izađe izvan scene i to vrijedi za sve instance tog objekta. [13]



Slika 48 Kreiranje ponašanja za uništavanje instance tijekom izlaska iz scene

4.4.4. Vučni objekt

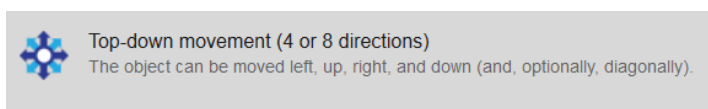
Ponašanje vučnog objekta je tip ponašanja gdje dobivamo mogućnost pomicanja instanci po ekranu pomoću računalnog miša ili prstom ako je igri omogućena *touchscreen* funkcija. Vučni objekt koristi se u mnogim igrama poput slagalica ili igrama gdje je cilj spojiti objekte s drugim objektima. [13]



Slika 49 Kreiranje ponašanja vučnog objekta

4.4.5. Top-Down kretanje

Osim platformer kretanja koje se koristi u 2D platformerskim igrama, GDevelop nudi i Top-down kretanje koje objektu omogućava kretanje u svim smjerovima. Ovaj tip ponašanja zadaje se objektu koji će imati funkciju glavnog lika ili NPC protivnika koji će se kretati po sceni. [13]



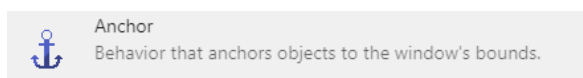
Slika 50 Kreiranje ponašanja za Top-down kretanje objekta

Klikom na tip ponašanja otvara se novi prozor gdje se određuju dodatna svojstva objekta poput brzine kretanja, brzine rotacije, omogućavanje kretanja po dijagonali što dodaje četiri novih smjerova kretanja, te određivanje kuta prema kojem će naš objekt biti okrenut dok je na mjestu. Ponašanje Top-down kretanje vrlo je korisno tijekom izrade izometričnih

igara gdje se objekti pomoću u svim smjerovima na sceni (sveukupno 8 smjerova kretanja). [13]

4.4.6. Anchor

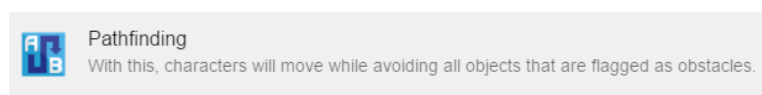
Anchor ponašanje je jednostavan tip ponašanja koji omogućava „usidrenje“ ili pričvršćivanje instance na ekran tako da se svaki kut instance pričvrsti kutovima ekrana ili scene. Ako se veličina scene izmjeni, distanca između kuta anchor objekta i kuta ekrana ostat će ista. Anchor ponašanja vrlo su korisna kod statičnih UI ili HUD objekta. [13]



Slika 51 Kreiranje Anchor ponašanje za objekt

4.4.7. Pronalaženje puta

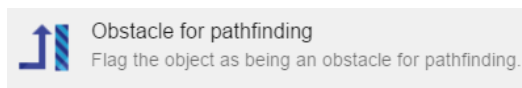
Pronalaženje puta ili Pathfinding ponašanje je tip ponašanja koji se dodaje objektu koji želimo pomicati po ekranu od točke A do točke B dok izbjegavamo sve druge objekte označene kao prepreke. U prozoru za određivanje svojstva, objektu se određuje brzina kretanja, mogućnost kretanja dijagonalama, maksimalna brzina, mogućnost i brzina rotacije objekta, te se određuje virtualna mreža po kojoj će se objekt kretati. [13]



Slika 52 Kreiranje Pathfinding ponašanja za objekt

4.4.8. Pathfinding prepreke

Isto kao što se za platformera određuju platforme, tako se za Pathfinding objekte određuju Pathfinding prepreke. Za ponašanja objekta kao prepreke odabiru sve dvije opcije u prozoru za svojstva. Prva opcija je ako želimo da je prepreka neprohodna te se stoga mora izbjeci. Druga opcija je da prepreka ima cijenu prolaznosti, gdje objekt može proći kroz prepreku ali uz određeni uvjet ili efekt koji će utjecati na objekt koji prolazi. [13]



Slika 53 Kreiranje ponašanja Pathfinding prepreke za objekt

4.4.9. Physics Engine 2.0

Jedan od novijih dodataka za GDevelop 5 program je Physics Engine 2.0 ponašanje za željeni objekt. Objekti s ovim tipom ponašanja se mogu kretati na više realističniji način prateći i pridržavajući se zakonima fizike. Ponašanje se sastoji od tri koncepta: svijet koji drži tijela i spojeve, sama tijela, i spojevi koji vežu tijela. [13]

Kada govorimo o svijetu, referiramo na prostor gdje se odvijaju simulacije tog ponašanja i gdje su tijela i spojevi povezani u svijet kako bi izvršavali interakciju s drugim entitetima. Veličina svijeta, odnosno veličina GDevelop scene u kojoj se koristi Physics Engine je beskrajna, zbog toga se objekti mogu nalaziti bilo gdje i mogu biti bilo koje veličine. Svi objekti u tom svijetu uvijek će pratiti zakone fizike. [13]

Svijet ima tri glavna svojstva: [13]

- Gravitacija – svojstvo imitira efekt gravitacije i dodaje silu na svaki objekt gurajući ih u zadani smjer
- Vremenska skala – svojstvo definira skalu vremena koju će simulacija koristiti
- Skala – svojstvo koje konvertira distance i veličine iz piksela u metre. GDevelop prema zadanom koristi piksele, ali kada se uvede Physics Engine ponašanje pikseli se pretvaraju u metre.

Drugi koncept Physics Engine ponašanja je tijelo. Tijela definiraju kako će se objekt pomicati, kakvog će biti oblika, kolika će biti veličina odskakivanja, na koji način će se rotirati i slično. Kada se ponašanje doda objektu, možemo modificirati konfiguraciju tijela na željeni način te odrediti svojstva poput tipa tijela, oblika tijela, postavke tijela i način kretanja. [13]

Kada određujemo tip tijela ponuđene su tri opcije: [13]

- Dinamičan tip – tip tijela koji na koji utječe gravitacija i može se sudariti s drugim dinamičnim, statičnim ili kinematičkim tijelima. Sva tijela koja se mogu pomicati definirana su kao dinamični tip.
- Statični tip – tip tijela na kojeg gravitacija ne utječe i ne mogu se pomaknuti silom.
- Kinematički tip – specijalan tip tijela na kojeg ne utječe gravitacija ili sila, ali možemo ih pomicati. Kinematički tip se daje objektima koji predstavljaju zidove koji se mogu pomaknuti.

Kada određujemo oblik tijela ponuđene su opcije: [13]

- Kutija – objekt je formiran kao pravokutnik s osovinom.
- Krug – objekt je formiran kao krug s prilagođenim polumjerom.
- Edge – objekt je formiran kao tanka linija bez volumena namijenjen za statične svrhe.
- Polygon – objekt je formiran u prilagođen oblik pomoću vertikalala.

Osim tipa i oblika, možemo definirati i druge dodatne postavke tijela kao što su: [13]

- Gustoća – svojstvo kojim određujemo masu tijela.
- Trenje – svojstvo kojim dodajemo silu tijelima koja su u dodiru. Dodaje se sila u suprotnom smjeru u kojem se objekti pomiču.
- Restitucija – svojstvo kojim određujemo količinu energije koju će objekt držati ili gubiti tijekom svake kolizije s drugim objektima.
- Linearno prigušivanje – svojstvo koje smanjuje linearnu brzinu objekta.
- Kutno prigušivanje – svojstvo koje smanjuje kutnu brzinu objekta.
- Skala gravitacije – svojstvo kojim određujemo gravitaciju tijela, te odnos gravitacije tijela s gravitacijom svijeta

Posljednje svojstvo koje se određuje na tijelu je kretanje. Pomoću editora događaja tijelu se definira način na koji će se kretati kada ga mi upravljamo. Definira se određen broj svojstva poput sile kretanja, okretni momenti, impulsi, brzina, kutni impulsi i kutna brzina. [13]

Posljednji koncept Physics Engine ponašanja su spojevi koji vežu tijela. Spojevi su ograničenja koja se daju objektima kako bi se pridržavali dodatnim pravilima tijekom kretanja. Njihovo korištenje može biti za različite svrhe poput omogućavanja rotacije tijela oko zadane točke, zadržavanja dva tijela spojena na jednom mjestu ili omogućavanja kompleksnijih funkcija poput rotacije više guma na vozilima. [13]

Kada se kreira jedan spoj, u njega se sprema varijabla s jedinstvenim identifikacijskim brojem (ID). Korištenjem ID-a, spoj možemo modificirati, mijenjati zadane vrijednosti te ga izbrisati. Postoje više tipova spojeva s jedinstvenim ponašanjima i svojstvima. Tipovi spojeva su: [13]

- Daljinski spoj – spojevi koji drže dva tijela odvojena na određenoj udaljenosti.
- Revolute spoj – spojevi koji djeluju kao šarke i koji omogućavaju rotaciju drugih tijela oko određene točke.
- Prizmatični spoj – spojevi koji djeluju kao klizači i koji omogućavaju klizanje tijela po danoj osi.
- Pulley spoj – spojevi koji djeluju tako da sažimanje udaljenosti od svakog objekta do točke na tlu ostaje konstantno.
- Zupčasti spoj – specijalan tip spoja koji povezuje dva spoja koji pokreću jedan drugoga.
- Mouse spoj – spojevi koji vuku tijela prema ciljanoj poziciji.
- Kotač spoj – jednostavan spoj koji simulira kretanju kotača.
- Zavarni spoj – spojevi koji čvrsto povezuju dva tijela u jedan objekt
- Konop spoj – jednostavan spoj koji simulira užad koja povezuje dva tijela.
- Frikcijski spoj – spojevi koji simuliraju trenje na Z osi usporavajući relativnu brzinu između dva tijela
- Motorni spoj – spojevi koji vuku tijela prema ciljanoj poziciji.

4.4.10. Tween

Tween ponašanje omogućava nam animiranje svojstva nekog objekta kroz određeni vremenski period. Ovo ponašanje dolazi s različitim efektima ublažavanja koji se koriste pri interpoliranju između početne i krajnje vrijednosti za stvaranje vizualno dojmljivih efekata

poput tresenja gumba korisničkog sučelja ili efekata nad kolekcionarskim predmetima koji izbljede nakon što ih pokupimo u igri. [13]



Slika 54 Kreiranje Tween ponašanja za objekt

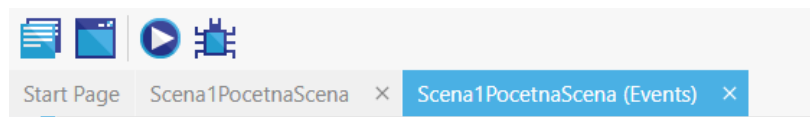
Tween ponašanja se mogu reproducirati u određenom redoslijedu stvarajući složene animacije i ponašanja objekta. Taj način reproduciranja nazivamo vezivanje ili *chaining*. Prema primjeru na slici (Slika 55) objekt započinje reprodukciju animacije „Tween and Scale“ koja bi smanjila objekt nakon klika lijeve tipke na mišu. Po završetka tog tween ponašanja, pokreće se nova animacija da se objekt ponovo poveća. To stvara petlju, zbog koje će objekt konstantno pulsirati. Varijacija toga također se može koristiti za stvaranje platformi koje se kreću gore-dolje ili za neprijatelje koji patroliraju i kreću se između putnih točaka odnosno *waypoints* točaka. [13]

Add condition	Add action
Left mouse button was released	Tween the scale of <code>ScaleTweenSprite</code> to X-scale: 0, Y-scale: 0 with easing "easeInOutExpo" over 1000ms as "ScaleTweenSmall"
The cursor/touch is on <code>ScaleTweenSprite</code>	Add action
Add condition	Add condition
Tween "ScaleTweenSmall" on <code>ScaleTweenSprite</code> has finished playing	Remove the tween "ScaleTweenSmall" from <code>ScaleTweenSprite</code>
Trigger once	Tween the scale of <code>ScaleTweenSprite</code> to X-scale: 1, Y-scale: 1 with easing "easeInOutExpo" over 1000ms as "ScaleTweenBig"
Add condition	Add action
Tween "ScaleTweenBig" on <code>ScaleTweenSprite</code> has finished playing	Remove the tween "ScaleTweenBig" from <code>ScaleTweenSprite</code>
Trigger once	Tween the scale of <code>ScaleTweenSprite</code> to X-scale: 0, Y-scale: 0 with easing "easeInOutExpo" over 1000ms as "ScaleTweenSmall"
Add condition	Add action

Slika 55 Korištenje Tween ponašanje u editoru događaja

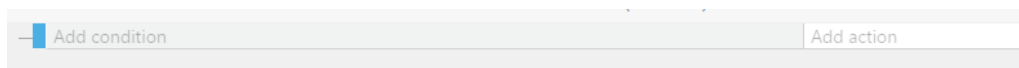
4.5. Događaji

Događaji ili *Events* GDevelop programa su logika koja pokreće i upravlja igru. Događaji se slažu i kombiniraju te u neku ruku programiraju u editoru događaja ali bez velikih znanja o programskim jezicima i njihovom korištenju. Događaji se slažu i pišu na brz i efikasan način, te je logika lagana za čitanje i pisanje. To omogućava autorima, dizajnerima i drugim korisnicima brzo razumijevanje svih pravila uz mogućnost stalnog ažuriranja logike igre. [13]



Slika 56 Editor događaja na sučelju

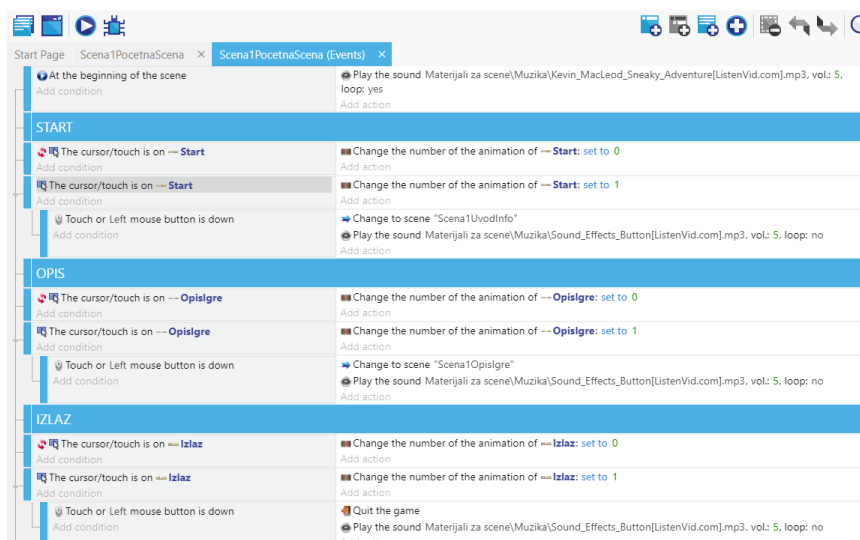
Prilikom kreiranja scene otvaraju se dvije kartice, editor scene i editor događaja. Kako bi kreirali događaje za našu scenu moramo prvo otvoriti prozor editora događaja klikom na navedenu karticu. Ovisno o sceni prozor koji se otvori je prazan s početnim poljima za unos, ili može već sadržavati prijašnje kreirane događaje s uvjetima i akcijama koje možemo doraditi. Kao što smo već spomenuli glavna procedura kreiranja događaja je određivanje uvjeta pod kojim se neki događaj treba dogoditi, te akcije koja obavlja zadanu funkciju događaja. [13]



Slika 57 Prazni prozor editora događaja s poljima za unos uvjeta i akcije



Slika 58 Alati za kreiranje događaja

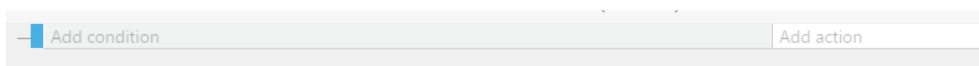


Slika 59 Editor Događaja sa zadanim uvjetima i akcijama

U prijašnjem pod-poglavlju o Editoru Događaja opisali smo sve pojedinosti sučelja editora i osnovne funkcije, stoga ćemo u ovom poglavlju proučiti različite tipove događaja koje nam GDevelop pruža, te načine organiziranja. [13]

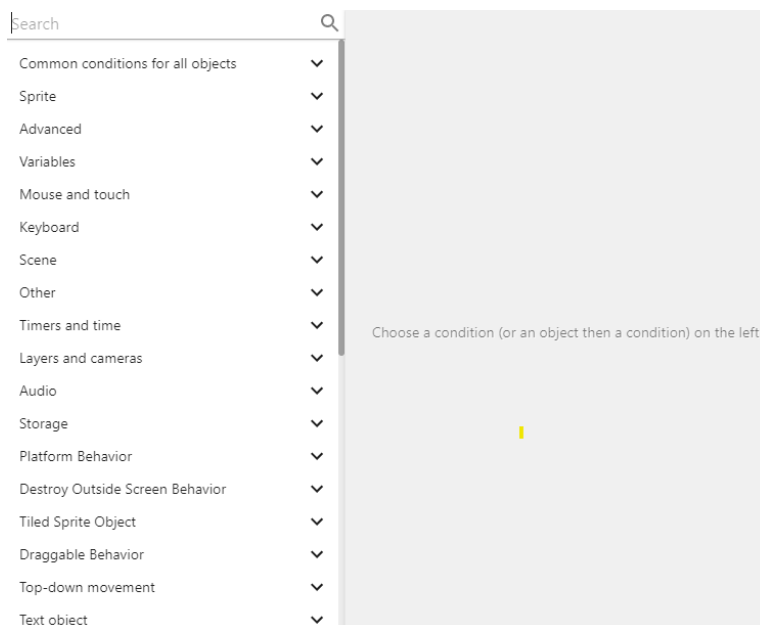
4.5.1. Standardni događaj

Najčešće korišteni događaji u GDevelop programu su standardni događaji. U editoru događaja na alatnoj traci kliknemo na gumb za kreiranje događaja što stvara praznu tablicu u koju se dodaju uvjeti i akcije. Svaki događaj sastoji se od uvjeta koji se nalazi na lijevom polju tablice, i akcije koja se nalazi na desnoj strani. [13]



Slika 60 Prazna tablica gdje se dodaju uvjeti (lijevo) i akcije (desno)

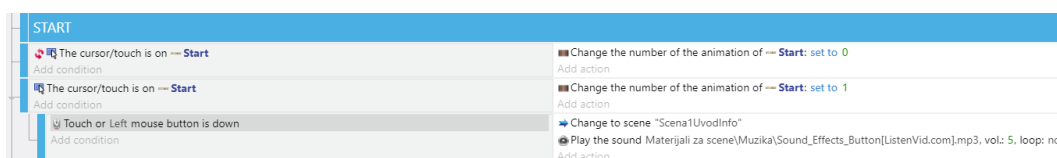
Klikom na gumb za uvođenje uvjeta *Add condition*, otvara se prozor s listom različitih uvjeta koji će utjecati na željeni objekt. Lista sadrži uvjete za sve vrste objekata na sceni i izvan scene, te za sve vrste i načine upravljanja i djelovanja na objekt od strane igrača. [13]



Slika 61 Lista uvjeta

Kako bi uvjet dobio neki rezultat moraju se dodati i akcije. Kao što smo spomenuli, akcije se dodaju na desnoj strani tablice klikom na gumb *Add action*. Isto kao i kod uvjeta, otvara se prozor s listom svih mogućnih akcija koje objekt može obaviti. Na taj jednostavan način stvara se osnovni događaj koji će obaviti neku specifičnu radnju tijekom igranja igre. Kada je zadani uvjet ispunjen, odnosno kada je *true*, aktivira se zadana akcija tog uvjeta. [13]

Jedan od jednostavnijih primjera događaja je kada odredimo da se klikom na neki objekt dogodi neka određena radnja. Za uvjet postavi se da strelica miša mora dirati objekt i mora se kliknuti lijevom tipkom, onda i samo onda dogodi se akcija koju će taj objekt izvršiti npr. klikom na dugme START na početnoj sceni igre započeti će drugu scenu. [13]



Slika 62 Korištenje uvjeta i akcije za mijenjanje scene nakon klika START

4.5.2. For Each događaj

For Each događaj, ili u prijevodu „Za sve“, odnosi se na tip događaja koji uzima objekt kao parametar. Tijekom igre, za svaku instancu objekta ponovit će se svi zadani uvjeti i akcije tog objekta. For Each događaji se ne koriste često zato što uvjeti i akcije automatski obuhvaćaju sve objekte. Međutim, u nekim slučajevima događaj može biti koristan za pokretanje akcije koja ovisi o svakoj instanci objekta. [13]

4.5.3. Repeat i while događaji

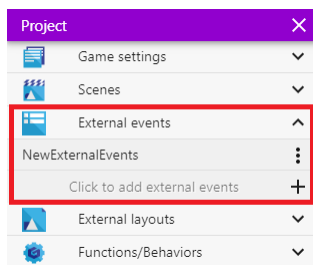
Događaj ponavljanja ili *Repeat* događaj radi na principu standardnog događaja ali s dodatkom da se taj događaj ponovi više puta zaredom prije nastavka na sljedeći događaj na listi u editoru. Događaji koji slijede se ne izvršavaju sve kod se Repeat događaj prije njih ne ponovi prema zadanom broju repeticije. [13]

While događaj je specijalan tip *Repeat* događaja koji se neprekidno ponavlja sve dok su specificirani uvjeti *true*. Kada uvjeti While događaja nisu više istiniti program prelazi na sljedeći događaj u listi. Isto kao u većini programskih jezika događaj radi na principu *while* petlje. [13]

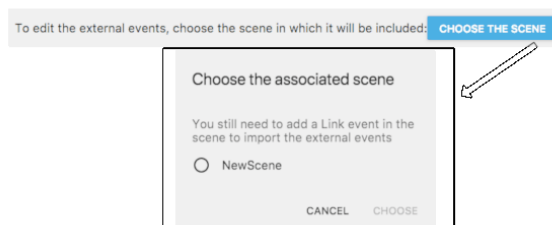
4.5.4. Povezani događaj

Povezani ili Link događaj odnosi se na poveznicu koja povezuje vanjske događaje te postavlja ih na rezervirano mjesto unutar liste u editoru događaja. Kao što smo prije spomenuli, vanjski događaji korisni su kada želimo odvojiti logiku igre na više dijelova što rezultira boljom preglednošću i organiziranosti u kompleksnijim igrama. Iste vanjske događaje može koristiti svaka scena, a to nam omogućava funkcija Link događaja. [13]

Vanjske događaje kreiramo klikom gumb *Click to add external events* na prozoru voditelja projekta. Nakon kreiranja vanjskog događaja u dodatnom prozoru možemo odabrati scenu s kojom će taj događaj biti povezan. [13]



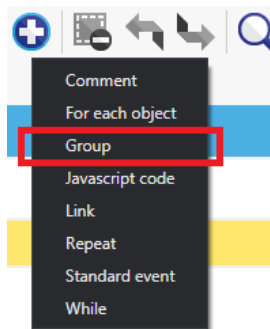
Slika 63 Kreiranje vanjskog događaja



Slika 64 Kreiranje Link događaja - Povezivanje vanjskog događaja sa scenom

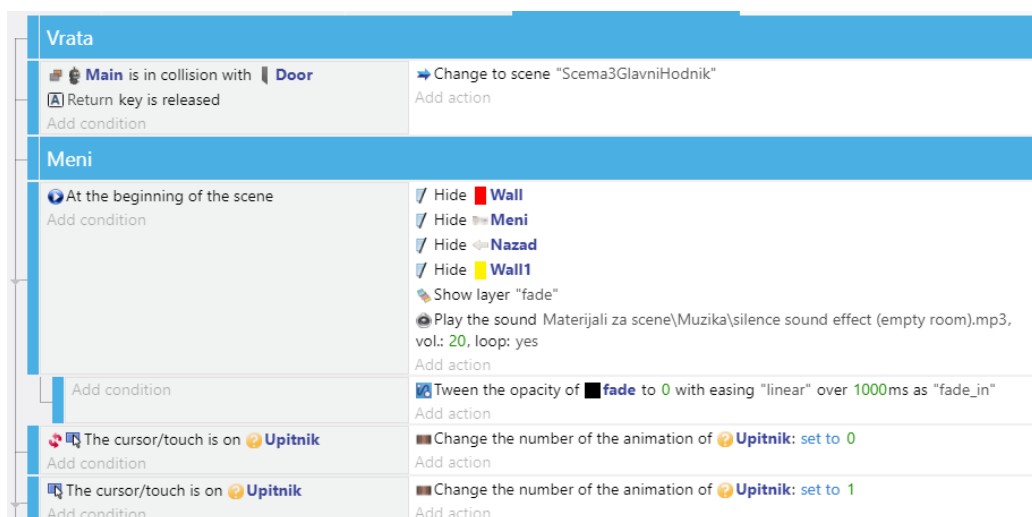
4.5.5. Grupiranje događaja

Grupiranje događaja je jednostavna funkcija koja organizira i strukturira događaje u određene grupe. Klikom na gumb u obliku velikog plavog plusa na alatnoj traci, otvara se lista s opcijama gdje se može odabrati funkcija grupiranja. [13]



Slika 65 Grupiranje događaja

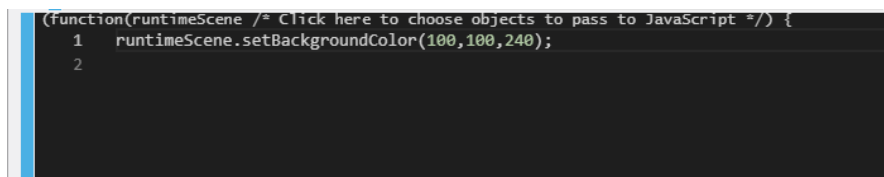
Događaje koje grupiramo su povezani tako da izvršavaju funkcije koje su namijenjene za jednu određenu svrhu. Stoga sve događaje koji utječu na interakciju s nekim objektom grupiramo u grupu događaja pod nazivom „Interakcija s objektima“. Tako je lista događaja znatno preglednija i organiziranija u vrlo kompleksnim igrama koja ima velik broj događaja i pod-događaja. [13]



Slika 66 Grupiranje događaja u grupe Vrata i Meni

4.5.6. JavaScript Code događaj

Iako smo naveli da GDevelop ne zahtijeva napredno znanje programiranja, program svejedno nudi korištenje programskog jezika za napredne korisnike. JavaScript Code događaj je *event* koji nam omogućava dodavanje prilagođenog JavaScript koda u logiku naše igre. Klikom na veliki plavi plus na alatnoj traci otvara se lista na kojoj je opcija *Javascript code*. Odabirom te opcije otvara se terminal unutar prozora editora događaja gdje nam je omogućeno pisanje JavaScript koda. [13]



```
(function(runtimeScene /* Click here to choose objects to pass to JavaScript */) {  
  1  runtimeScene.setBackgroundColor(100,100,240);  
  2
```

Slika 67 JavaScript Code terminal u editoru

4.6. Ostali dodaci programa

Osim dodataka poput objekata i ponašanja te svih omogućenih svojstva, GDevelop pruža i nekolicinu drugih zanimljivih i korisnih dodataka. Ti dodaci mogu se koristiti izravno pomoću akcija, uvjeta ili izraza, te se dio njih kategoriziraju i kao ekstenzije kojima nije potreban objekt na sceni. [13]

4.6.1. Napredni uvjeti

Kada smo govorili o različitim vrstama događaja spomenuli smo događaje koji koriste petlje poput *For* i *While*. Osim tih osnovnih programskih petlji, u programu možemo koristiti i jednostavne logičke operacije poput: [13]

- Or logička operacija – kada uvjet ima zadanu *Or* operaciju, akcija će se izvršiti kada se izvrši jedan ili više uvjeta ili pod-uvjeta.
- Not logička operacija – kada uvjet ima zadanu *Not* operaciju, događaj vraća pod-uvjet sa suprotnom vrijednošću (ako je pod-uvjet *true*, onda vraća *false*).
- Trigger once – kada je uvjet ispunjen, akcija za taj događaj se aktivira samo jednom.

- Compare two numbers – uvjet koji uspoređuje dva broja korištenjem znakovima nejednakosti.
- Compare two strings – uvjet koji uspoređuje dva stringa, odnosno izraza korištenjem znakovima nejednakosti.
- Always true/false – uvjet koji uvijek vraća ili *true* ili *false*.

4.6.2. Audio

Važan dio svake igre su muzika i zvukovi. Muzika stvara dodatni ambijent i osjećaj privlačnosti prema likovima, priči i samoj igri. GDevelop program omogućava uvođenje različitih uvjeta i akcija namijenjenih za audio datoteke. Tijekom kreiranja događaja omogućeni su uvjeti i akcije *Play a sound* ili *Play a music file* kojim odabiremo vlastiti audio zapis koji će ozvučiti pozadinu ili određenu akciju objekta. [13]

Postoji razlika između uvođenja zvuka (*sound*) i muzike (*music*). Zvukovi su audio zapisi koji traju samo nekoliko sekundi i oni se uvode za akcije koje određeni objekt izvrši, dok je muzika duži audio zapis koji se obično koristi za pozadinski ambijent. GDevelop podržava dva audio formata: [13]

- wav – format namijenjen za zvukove
- mp3 – format namijenjen za muziku

4.6.3. Inventura

Inventure su memorijske strukture koja omogućavaju praćenje broja objekta pobranih od strane igrača tijekom igre. Predmeti inventure se identificiraju po prema zadanim imenima, te može ih biti beskrajno. Inventure su prvotno prazna memorijska skladišta u koja možemo spremati predmete naredbom *Add an item*. Isto tako program omogućava i brisanje predmeta iz inventure naredbom *Remove an item*. Inventure se kreiraju pomoću editora događaja gdje se specificiraju svojstva inventure, način prikaza predmeta u inventuri te njihovo korištenje. [13]

4.6.4. Timer

Timer ili brojač je svojstvo koje se može dodati svakom objekt u sceni. Kao što smo već prije spomenuli, svaki *timer* je drugačiji za svaki instancu objekta na sceni. Oni su korisni kada određujemo kada instanca izvršava određenu akciju ili može se odrediti da instanca izvršava akciju svakih X sekundi. Razlikujemo dvije vrste brojača: [13]

- Timer scene – brojač koji se koristi na samoj sceni igre. Uvjeti i akcije koje se odnose na scenu izvršavaju se kada brojač dosegne određenu brojku.
- Timer objekta – brojač koji se koristi na pojedinom objektu odnosno instanci objekta na sceni. Svaka instanca može držati vlastiti brojač koji je neovisan od brojača drugih instanci objekta.

4.6.5. Varijable

Kao što smo već prije spomenuli varijable su spremnici u koji se mogu pohraniti određene numeričkih ili tekstualnih vrijednosti. Korištenjem varijabli u igri se mogu pohraniti različiti tipovi podataka poput imena igrača, broja predmeta u inventuri, *high score* igre i slično. Različiti tipovi vrijednosti moraju se pohraniti u različite tipove varijabli, a GDevelop podržava tri tipa varijabli: broj, tekst ili string, strukturu (varijabla koja sadrži druge varijable). [13]

Kad se pohrani neka vrijednost treba se također razmotriti na koji način će se ta vrijednost kasnije upotrijebiti. Stoga dolazimo do opsega varijabli koji određuje gdje i kada će se pristupiti varijabli, te koliko dugo se varijabla s vrijednošću može zadržati u memoriji. GDevelop omogućava tri opsega varijable: [13]

- Globalna varijabla – varijable koje su spremljene u memoriji dokle god igra traje, te se u njih spremaju podaci koje želimo dijeliti između svih scena u cijeloj igri. Globalna varijabla se izbriše nakon završetka igre. Primjer globalne varijable je konačna *high score* vrijednost igre koju igrač prikupi tijekom igre.
- Varijabla scene – varijable koje su spremljene u memoriji dokle god je određena scena aktivna, te se u njih spremaju podaci koje ne želimo dijeliti

između scena. Primjer korištenja varijable scene je spremanje određenog broja neprijatelja na jednoj sceni.

- Varijabla objekta – varijable koje se definiraju za svaku instancu objekta individualno i spremljene su u memoriju dokle god se instanca nalazi u sceni. Primjer varijable pojedinog objekta je „život“ ili vrijednost koja individualno omogućava svakoj instanci objekta da se zadrži na sceni. Kada instanci nestane život ona se briše iz scene, a tako se briše i varijabla iz memorije.

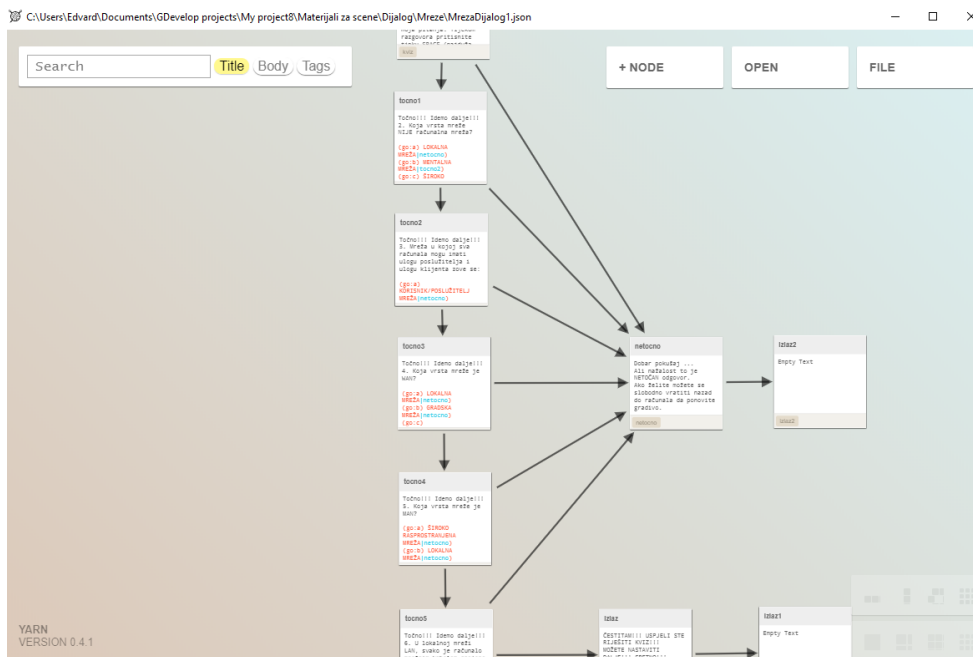
Sve vrste ili opsezi varijabli definiraju se različitim postupcima. Globalne varijable kreiramo na voditelju projekta, dok varijable scene i varijable objekta kreiramo u editoru scene i editoru objekta. Desnim klikom miša na željenu scenu otvara se opcija *properties* gdje se, uz ostala svojstva, može definirati varijabla scene. Za varijablu objekta kliknemo desnim klikom miša na objekt u editoru objekta i odaberemo opciju *Edit Object Variables*. [13]

4.6.6. Dijalog stablo ekstenzija

Dijalog stablo ekstenzija u GDevelop programu koristi se za brzo stvaranje dinamičkog ponašanja za dijalog između dviju instanci koje su u većini slučajeva prezentirani kao likovi. Uvođenje dijaloga zahtijeva velik broj grupiranih događaja će izvršavati funkcije ovisno o odabranoj opciji na stablu dijaloga, te isto tako potreban je vanjski program zvan YARN u kojem se kreira stablo. Dijalog je srž svake priče i prikupljanja informacija od NPC likova, te je jedan od najvažnijih dijelova RPG igara. Uvođenjem dijalog stabla u GDevelop, igra postaje više fokusirana i interaktivna. [13]

Kao što smo naveli dijalog stablo se kreira pomoću programa Yarn. Program je vrlo jednostavan za korištenje i mogu se kreirati velika stabla dijaloga za bilo koju situaciju. Svaki tekst unutar dijaloga definiran je kao *node*. Svaki node sadrži jedan tekst te u njemu se definiraju opcije grananja dijaloga što rezultira stvaranje više nodova. Tako se može kreirati jednostavan kviz s višestrukim izborom opcija ili način proučavanja informacija i dobivanja odgovora. Nakon kreiranja svih nodova izrađen je jedan dijalog koji se mora spremiti kao datoteka u .json formatu koji GDevelop podržava i zahtijeva. Datoteka se sprema u folder gdje se nalaze resursi igre, te se zatim pomoću editora događaja dijalog

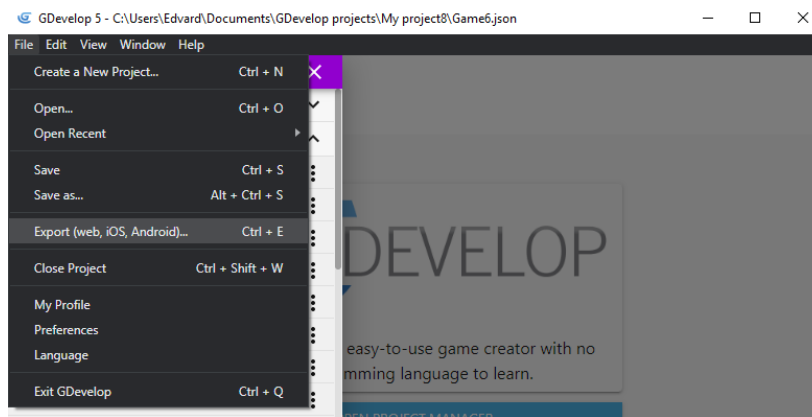
uvodi u scenu gdje se određuje uvjet kada se dijalog aktivira i akcija koja izvršava funkciju svake ponuđene opcije stabla dijaloga. [13]



Slika 68 Kreiranje dijalog stabla pomoću Yarn editora

4.7. Objavljivanje igre

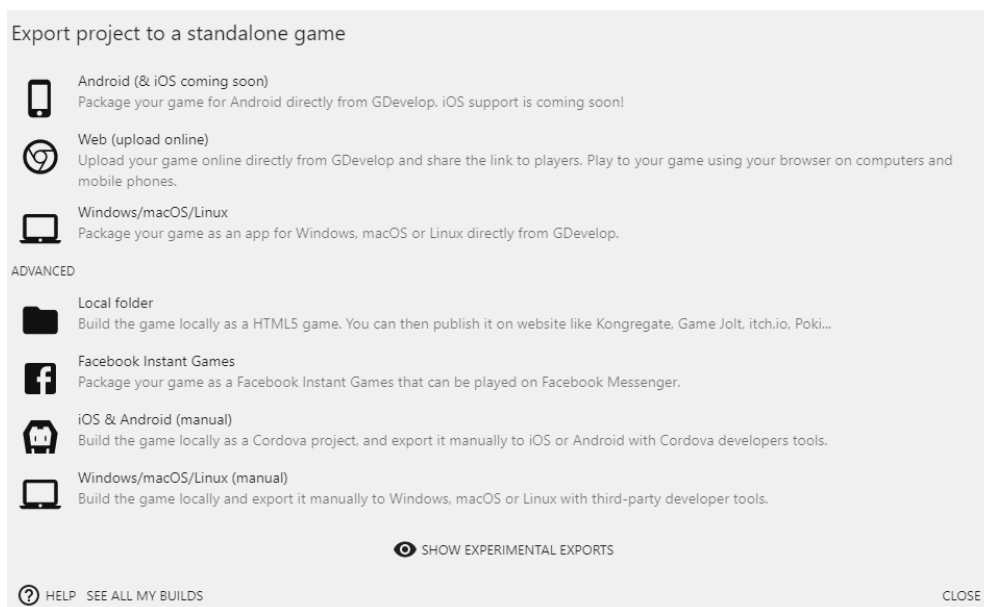
Posljednja faza svakog razvoja igre u bilo kojem programu je završno objavljivanje igre na željenim platformama. GDevelop omogućava objavljivanje kreiranih igara na svim platformama: desktop, web i mobilni uređaj. Nakon završavanja svih faza razvoja igre i kada potvrdimo da smo zadovoljni s igrom i da sve radi prema zadanom, ulazimo u fazu objavljivanja. Klikom na *File* u meniju programa, otvara se prozor gdje je ponuđena opcija *Export (web,iOS, Android)*. [13]



Slika 69 Export igre

Klikom na opciju izvoza (*Export*) otvara se novi prozor gdje možemo odabrati željenu platformu na kojoj će igra moći raditi. Kao što smo naveli od opcija su ponuđene: [13]

- Android – spremanje i objavljivanje igre za mobilne uređaje. Igra se kreira putem GDevelop online usluge koja šalje link pomoću kojeg se igra može instalirati na Android uređaju.
- Web – opcija koja besplatno sprema i učitava igru na GDevelop online server. Igru tada možemo dijeliti putem interneta
- Windows, macOS i Linux – spremanje i objavljivanje igre na operacijskim sustavima. Opcija sprema igru u paket kao aplikaciju putem GDevelop online usluge.



Slika 70 Platforme za objavljivanje igre

Osim tih opcija GDevelop također nudi opciju objavljivanja igre na Facebook društvenoj mreži. Igra se objavljuje na *Facebook Instant Games* dodatku. GDevelop nudi i izvoz igre za napredne korisnike tako da je omogućeno ručno objavljivanje u određeni lokalni folder, ili objavljivanje igre za projekte poput Cordova ili Electron u kojima se ručno izgrađuje igra za Windows, macOS i Linux, ili za mobilne uređaje. [13]

Objavljivanjem igre završava se proces kreiranja igre na GDevelop programu. Program još nudi nekolicinu drugih dodataka i ekstenzija za znatno napredne korisnike, poput kreiranja vlastitih ponašanja objekta, funkcije događaja, te vlastitih ekstenzija koje se mogu dodati u sustav programa. U suštini, za većinu korisnika programa koji žele kreirati jednostavnu igru s nekom svrhom, najvažnije je savladati glavne dijelove sustava, a to su kreiranje i manipuliranje objekta, znanje korištenja događaja za upravljanje logike igre, te sama vizija i ideje developera igre. [13]

5. Proces izrade igre prema SADDIE modelu

Svaka igra ima svoj proces izrade i metodologiju razvoja. U našem slučaju za izradu obrazovne, odnosno didaktičke igre pomoću GDevelop programa korišten je SADDIE (Specification, Analysis, Design, Development, Implementation, Evaluation) model koji određuje šest ključnih faza procesa izrade. Ključne faze prema SADDIE modelu su: Specifikacija, Analiza, Dizajn, Razvoj, Implementacija i Testiranje. [10]

U prvoj fazi izrade didaktičke igre, fazi specifikacije, određuje se tema igre, priča, likovi, nastavna jedinica koju će igra obuhvatiti, te koja će biti svrha ili cilj igre, odnosno što se želi postići igrom i koja će se temeljna znanja učeniku (igraču) prezentirati. Definišu se ishodi učenja koji određuju što će učenik biti u stanju razumijeti nakon završetka didaktičke igre. U drugoj fazi izrade, odnosno fazi analize, trebaju se pripremiti sve potrebne informacije o igri kao što su likovi, predmeti, scene i priča, te trebaju pripremiti svi resursi potrebni za implementaciju u igru. Određuje se skupina korisnika kojoj će igra biti namijenjena, informacije i resursi se usklađuju s kurikulumom, specificiraju se općenite ideje igre i na koji će način biti povezane sa željenom nastavnom jedinicom, provjerava se dostupnost razvojne okoline i dostupnost softverskih i hardverskih resursa. [10]

Nakon faze analize slijedi faza dizajna u kojoj se specificira i dizajnira svijet igre, a to su likovi, objekti, scene, odnosi između likova i objekta, pravila igre, i drugi elementi koji laze u igru. Tijekom faze dizajna vrlo je korisno pripremiti i modelirati sve navedene resurse koji će biti potrebni prilikom kreiranja igre u programu. [10] Iz vlastitog iskustva, najvažniji program za pripremu resursa i njihovo dizajniranje je *Adobe Photoshop*.

U sljedećoj fazi, u fazi razvoja, započinje se s kreiranjem same igre pomoću željenog programa za izradu igre. U našem slučaju igra će se kreirati pomoću programa GDevelop 5. U fazi razvoja koriste se svi pripremljeni i prikupljeni resursi i informacije, te se programiranjem igre u navedenom programu izrađuje igra. Cilj faze razvoja je gotova igra koju u sljedećim fazama implementiramo i evaluiramo. [10]

Nakon kreiranja igre, u fazi implementacije igra se integrira u proces učenja i testira na grupi korisnika kojima je namijenjena. Nakon toga ulazimo u posljednju fazu evaluacije gdje te ocjenjuju prikupljeni rezultati testiranja igre. Tako možemo utvrditi pozitivne i negativne strane igre, te nedostatke na kojima se treba još raditi. [10]

6. Projektna dokumentacija igre

6.1. Kratak pregled igre: INFO – Igra za Uvod u Informatiku

INFO je igra napravljena pomoću GDevelop 5 programa u svrhu izrade edukativne didaktičke igre za mlađe razrede osnovne škole. Igra obuhvaća gradivo Informatike „Osnove računala“ i specifično je namijenjena za 5. razred. [3]

U igri upravljamo robota zvanog INFO koji se našao u svijetu Informatike. Kako bi izašao iz tog bizarnog svijeta INFO mora naučiti gradivo Osnove računala koje obuhvaća sklopovlje računala i povezivanje u mrežu. Svako gradivo se nalazi u jednoj od prostorija koje INFO mora osvojiti. Onda i samo onda otvorit će se izlaz kojim naš robot izlazi iz igre.

Uz GDevelop kao primarni program, za kreiranje igre korišteno je još nekoliko drugih programa. Najzastupljeniji program izrade je Adobe Photoshop koji se koristio za kreiranje gotovo svih vizualnih resursa prisutnih u igri, te za 3D modeliranje i animiranje svih dinamičkih objekata koji izvršavaju određenu animaciju u sceni. Osim toga koristio se Yarn editor za izradu dijaloga u igri, te Audacity za modificiranje i prilagođavanje audio zapisa korištenih u igri. Svi vizualni materijali preuzeti su s web stranica, te prilagođeni po potrebi igre.

Animiranje likova je važan dio pripreme tijekom kreiranja igre. Davanjem likovima određenu animaciju znatno pridonosi šarmu i privlačnosti igre. Scene postaju više aktivnije te likovi prikazuju da su živi i da mogu reagirati u određenim situacijama (interakcija, skakanje, hodanje). Osim animiranja, za određen dio igre korišteno je i 3D modeliranje. Kako bi se na privlačan način prikazalo Top-down ponašanje likova iz ptičje perspektive u izometričnoj sceni, pomoću Photoshopa 2D model glavnog like pretvoren je u 3D model.

Iako je GDevelop vrlo lagan i brz program za izradu igara, dodatni program za izradu vizualnih resursa poput Photoshopa je gotovo neophodan dio kreiranja i dizajniranja.

6.2. Specifikacija

6.2.1. Nastavni sadržaj

Nastavna tema odabrana za didaktičku igre: Osnove računala

Nastavne jedinice:

- Sklopovlje računala
- Povezivanje u mrežu

6.2.2. Učenici kojima je nastavni sadržaj namijenjen

Nastavni sadržaj didaktičke igre obuhvaća prvo gradivo Informatike „Osnove računala“ i namijenjen je za 5. razred osnovne škole.

6.2.3. Svrha i ishodi učenja

Svrha odnosno cilj učenja je upoznati učenika s osnovom informatike koja počinje upoznavanjem računala, njegovom namjenom, specifikacijama i njegovom podjelom na sklopovlje i programsku podršku. Nadalje cilj je upoznati učenike s osnovnim i dodatnim dijelovima sklopovlja računala, s njihovom podjelom na ulazne i izlazne jedinice te središnju jedinicu i upoznati funkcije svakog dijela sklopovlja.

Cilj druge nastavne jedinice „Povezivanje u mrežu“ je upoznati učenike s računalnom mrežom, vrstama računalnih mreža s obzirom na njihovu veličinu te vrstama računalnih mreža prema njihovoj ulozi. Također cilj je upoznati učenike s osnovnim pojmovima poput LAN, WAN, MAN, Internet, Mrežna kartica, Mreža računala, te na jednostavan način objasniti i opisati njihovo značenje i funkcije.

Uz cilj, određeni su i očekivani ishodi navedene nastavne teme, odnosno što će učenici biti u stanju razumjeti i pokazati nakon procesa učenja kroz igru. Nakon završetka nastavne jedinice „Sklopovlje računala“ u igri, učenici će moći:

- opisati i objasniti što je računalo i čemu služi

- objasniti što je računalni sustav
- navesti dva glavna podsustava računalnog sustava
- objasniti što je sklopovlje računala
- nabrojati osnovne dijelove sklopovlja
- opisati čemu svaki dio sklopovlja služi
- odrediti izlazne i ulazne jedinice
- odrediti dijelove središnje jedinice

Nakon završetka nastavne jedinice „Povezivanje u mrežu“ u igri, učenici će moći:

- objasniti što je Mreža računala
- nabrojati neke prednosti računalne mreže
- nabrojati vrste računalne mreže po veličini
- objasniti što je lokalna mreža
- objasniti što je gradska mreža
- objasniti što je široko rasprostranjena mreža
- nabrojati vrste računalne mreže po ulozi
- objasniti što je Korisnik/poslužitelj mreža
- objasniti što je mreža ravnopravnih korisnika

6.2.4. Oblik i metode rada

Prema obliku rada igra je namijenjena za individualno igranje. Svaki učenik individualno pristupa igri i uz pomoć informacija rasprostranjenih po igri rješava zadatke. Metode rada koje se koriste tijekom igranja igre su:

- Metoda čitanja i rada na tekstu – razumijevanje dobivenih informacija je nužno za rješavanje zadataka u igri
- Metoda razgovora (Katehetički razgovor) – dio zadataka u igri su kvizovi koji kratkim pitanjima ispituju prikupljeno znanje na temelju informacija koje učenik tijekom igre prikupi.
- Dokumentacijska metoda – u igri učenicima su predstavljene informacije kao važan izvor znanja

- Metoda usmenog izlaganja – nastavnik učenicima objašnjava igru, koje gradivo obuhvaća, i što se očekuje tijekom igranja.
- Operacijske metode – rješavanje zadataka koji su predstavljeni u igri.

6.2.5. Opis priče kojom se prezentira gradivo

Naslov igre: INFO – Igra za Uvod u Informatiku

Glavni lik: Robot INFO

Ostali likovi:

- DIO – vodič u svijetu Informatike koji se nalazi u centralnom hodniku
- Vitez Kule – vodič u Kuli Sklopovlja koji
- Čuvar Kule – Zli robot na vrhu Kule Sklopovlja
- KOR – vodič u Mrežnom Centru
- KIR – čuvar Mrežnog Centra

Tijek Igre:

Igra započinje s prvom scenom gdje je prikazan glavni meni s tri opcije: Nova Igra, Opis Igre i Izlaz. Klikom na opciju *Nova Igra* pokrećemo igru i prelazimo na uvodnu scenu s kratkim uputama o igri, te nakon toga otvaramo scenu koja opisuje priču igre i predstavlja glavnog lika robota INFO. Nakon tih uvodnih scena, započinjemo s igrom u početnoj sceni gdje se upoznajemo s glavnim likom i njegovim kontrolama, te drugim pojedinostima o igri koje su opisane na meni prozoru.

Nakon početne scene, ulazimo u lift koji nas vodi do glavne centralne prostorije gdje se igra pretvara iz 2D platformera u 3D izometričnu igru iz ptičje perspektive. U Centralnoj prostoriji nailazimo na prvi NPC (*Non-player character* ili sporedni lik kojeg ne upravljamo) zvan DIO koji predstavlja Svijet Informatike u kojem se nalazimo i detaljnije objašnjava što nas očekuje u igri, zadatke, te način završavanja igre. Centralna prostorija sastoji se od tri veće prostorije Kule Sklopovlja, Mrežnog Centra i prostorije za Mini Igre. Na početku

otvorena je samo Kula Sklopovlja koju moramo osvojiti da bi otvorili sljedeću prostoriju Mrežni Centar.

U Kuli Sklopovlja predstavljena je nastavna jedinica „Sklopovlje računala“ i izgled igre vraća se iz 3D izometrične igre u 2D Platformer. Kula Sklopovlja sastoji se od dvije prostorije u kojima se nalaze informacije o sklopovlju. U prvoj prostoriji za analizu prezentirana je skica sklopovlja računala podijeljenih na ulazne i izlazne jedinice, te središnju jedinicu. U drugoj prostoriji nailazimo na NPC Vitez Kule koji objašnjava zadatak kule i kako je osvojiti. Kula Sklopovlja je dizajnirana kao vertikalna kula s platformama na kojima INFO može skakati. Dio tih platformi su i dijelovi računala na kojima možemo stajati i s kojima INFO može izvršavati interakciju. Dodirom dijela računala otvara se prozor s informacijama o tom dijelu. Nakon analize svih dijelova na vrhu kule se nalazi Čuvar Kule koji postavlja kviz o sklopovlju računala. Rješenjem kviza osvaja se kula i igra nas vraća u Centralnu prostoriju kraj ulaza u Kulu Sklopovlja.

Osvajanjem kule, otvara se nova prostorija Mrežni Centar te se aktivira jedan dio lampe u središtu Centralne prostorije. U Mrežnom Centru predstavljena je nastavna jedinica „Povezivanje u mrežu“ i isto kao i u kuli igra mijenja izgled u 2D platformer. Mrežni Centar sastoji se od tri prostorije u kojima se nalaze informacije o mreži računala. U prvoj prostoriji nailazimo na NPC KOR koji objašnjava pojedinosti Mrežnog Centra, kratak pregled nastavne jedinice te što nas očekuje u nastavku. U drugoj prostoriji nalazi se monitor kojeg INFO može aktivirati. Aktivacijom monitora scena se prebacuje na prezentaciju o Računalnoj mreži i povezivanju u mrežu. Nakon analize prezentacije u prostoriji s monitorom otvara se portal koji vodi u treću prostoriju gdje nailazimo na NPC KIR koji postavlja kviz o računalnoj mreži. Rješenjem kviza osvaja se Mrežni Centar i igra nas vraća u Centralnu prostoriju kraj ulaza u Mrežni Centar.

Osvajanjem Mrežnog Centra, otvara se posljednja prostorija Mini Igre te se aktivira drugi dio lampe u središtu Centralne prostorije. U prostoriji Mini Igre nalaze se tri vrata i svaka vrata vode u jednu mini igru: Parovi, Vježba s tipkovnicom, Ulazno Izlazne Jedinice. Rješavanjem navedenih mini igara, otvara se mogućnost povratka u centralnu prostoriju gdje su aktivirane obje lampe i iznad njih lebdeći natpis IZLAZ. Dodirom lampe INFO izlazi iz Svijeta Informatike i prikazuje se scena koja završava priču. Klikom na scenu igra završava i vraćamo se ponovno na prvu scenu s glavnim menijem.

6.3. Analiza

6.3.1. Okvirni vremenski plan izrade

Vremenski plan izrade igre:

- Odabir teme – jedan tjedan
- Razrada scenarija, određivanje glavnog lika i sporednih likova, određivanje tijeka igre, izrada nacрта svih prostorija, određivanje zadataka i uklapanje tih zadataka u scenarij – tri tjedna
- Crtanje, modeliranje i animiranje svih likova, prostorija, objekata – tri tjedna
- Izrada igre u GDevelop 5 programu – dva tjedna
- Testiranje, evaluacija i ispravak igre – jedan tjedan

6.3.2. Okolina za izradu igre i ostala tehnologija

Programska oprema korištena za pripremu i izradu igre:

- GDevelop 5
- Adobe Photoshop
- Adobe Illustrator
- Paint
- Audicity
- Lunapic (online softver)
- Microsoft Word

6.3.3. Usklađenost obrazovne tematike s kurikulumom

Kao što je već navedeno, igra INFO obuhvaća uvodni dio gradiva Informatike za učenike 5. razreda osnovne škole (kurikulum: <https://sysprint.hr/eudzbenici/inf5u2019/>). [3] Igra je zamišljena kao dio nastavnog sata, te kao potencijalni alat za ponavljanje gradiva od kuće. Na uvodnom satu nastavnik ukratko predstavlja uvod u informatiku, uvod u računalo te određene pojedinosti uvodne tematike. Nakon toga nastavnik predstavlja igru i kratkom

demonstracijom prikazuje način igranja objašnjavajući što igra obuhvaća od gradiva i što se očekuje od učenika.

Učenici zatim individualno igraju igru i testiraju svoje znanje na zabavan način do kraja nastavnog sata. U igri su predstavljene sve važne informacije o uvodnom gradivu Informatike i rješavanjem zadataka učenici usvajaju predstavljene nastavne jedinice gradiva. U slučaju ako neki učenici ne završe igru, nastavnik šalje svakom učeniku instalacijski link igre kako bi dovršili igru, te kako bi dobili mogućnost ponavljanja gradiva na način igranja igre.

6.4. Scenarij

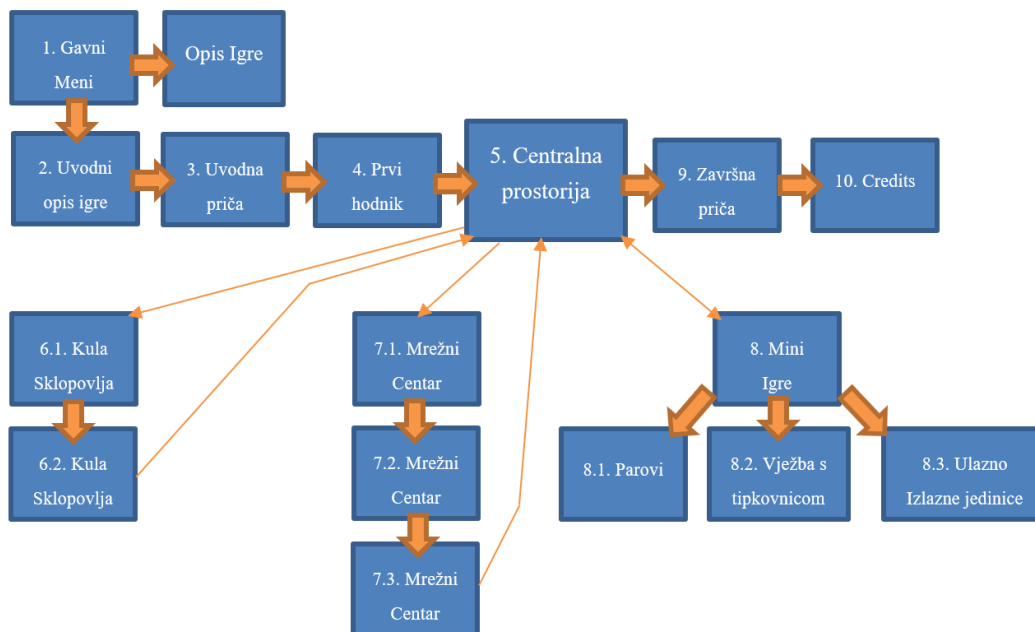
6.4.1. Popis i skica mape scena

Popis scena:

1. Glavni Meni
 - Opis Igre
2. Uvodni Opis Igre
3. Uvodna Priča
4. Prvi Hodnik
5. Centralna Prostorija
 - Centralna Prostorija nakon Kule Sklopovlja
 - Centralna Prostorija nakon Mrežnog Centra
 - Centralna Prostorija nakon Mini Igre
6. Kula Sklopovlja – prva prostorija
7. Kula Sklopovlja – druga prostorija
8. Mrežni Centar – prva prostorija
9. Mrežni Centar – druga prostorija
 - Prezentacija od 13 scena
10. Mrežni Centar – treća prostorija
11. Mini Igre
12. Parovi
13. Vježba s tipkovnicom

14. Ulazno Izlazne jedinice
15. Završna Scena
16. Credits

Skica mape scena:




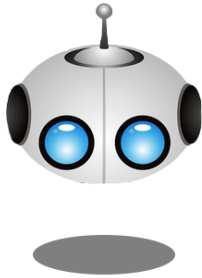


Slika 71 Skica mape scena


6.4.2. Popis likova

Popis likova:

Tablica 1 Popis likova

<p>Robot INFO: Glavni lik igre kojeg upravljamo. Pronašao vrata koja ga transportiraju u Svijet Informatike.</p>	
--	--

<p>Robot DIO: Vodič Svijeta Informatike. Nalazi se u Centralnoj prostoriji i pomaže nam u istraživanju Svijeta Informatike.</p>	
<p>Vitez Kule: Vodič Kule Sklopovlja. Nalazi se u drugoj prostoriji Kule Sklopovlja na samom dnu kule i objašnjava nam pojedinsti zadatka.</p>	
<p>Čuvar Kule: Čuvar Kule Sklopovlja. Nalazi se na vrhu kule gdje postavlja kviz koji obuhvaća gradivo nastavne jedinice „Sklopovlje računala“.</p>	
<p>Robot KOR: Vodič Mrežnog Centra. Nalazi se u prvoj prostoriji Mrežnog Centra i objašnjava nam pojedinsti zadatka.</p>	

<p>Robot KIR: Čuvar Mrežnog Centra. Nalazi se u posljednjoj prostoriji Mrežnog Centra gdje postavlja kviz koji obuhvaća gradivo nastavne jedinice „Povezivanje u mrežu“.</p>	
--	---

6.4.3. Razrada scenarija

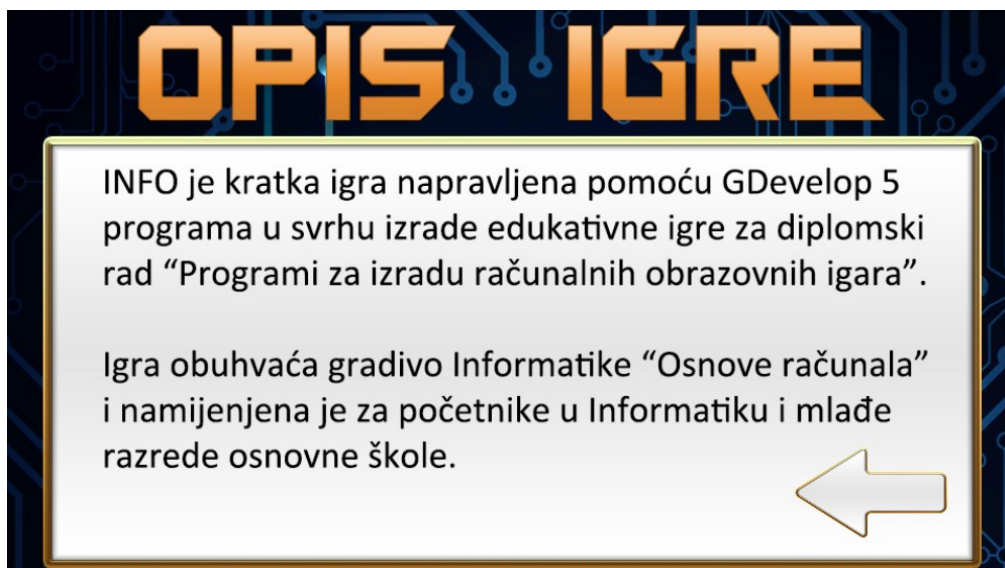
Scena 1 – Glavni Meni

Pokretanjem igre prva scena koja se prikazuje je Glavni Meni Igre koji ima tri gumba s određenim funkcijama: Nova Igra, Opis Igre i Izlaz. Iz glavnog menija možemo pokrenuti igru klikom na gumb Start ili prekinuti igru klikom na gumb Izlaz. Dodatno, ponuđena je opcija otvaranja scene s kratkim opisom igre.



Slika 72 Scena 1 - Glavni Meni Igre

Klikom na gumb Opis Igre otvaramo dodatnu scenu na kojoj je ukratko opisana igra i svrha njene izrade, te koje nastavno gradivo obuhvaća i kome je igra namijenjena.



Slika 73 Dodatna Scena - Opis Igre

Scena 2 – Uvodni Opis Igre

Klikom na gumb Start u Glavnom Meniju, otvara se nova scena na kojoj se ukratko opisuje svrha igre, kome je namijenjena i koje se nastavne jedinice obrađuju. Također opisano je na koji način možemo upravljati glavnog lika, i na koji način dobiti pomoć i upute.



Slika 74 Scena 2 - Uvodni Opis Igre

Opisano je da klikom na tipku Backspace otvara se dodatna pomoć tijekom igranja. Klikom na zeleni gumb Start otvaramo sljedeću scenu.

Scena 3 - Uvodna Priča

Uvodna Priča ukratko objašnjava priču igre i kako je naš glavni lik Robot INFO pronašao Svijet Informatike.

Priča: Jednog dana robot zvan INFO istraživao je veliku livadu i naišao na vrata koja prema natpisu vode u SVIJET INFORMATIKE. Zbog velike znatiželje, INFO je odlučio otvoriti vrata i istražiti misteriozni svijet.



Slika 75 Scena 3 - Uvodna Priča

Klikom lijeve tipke računalnog miša ili tipkom ENTER na tipkovnici ulazimo u sljedeću scenu.

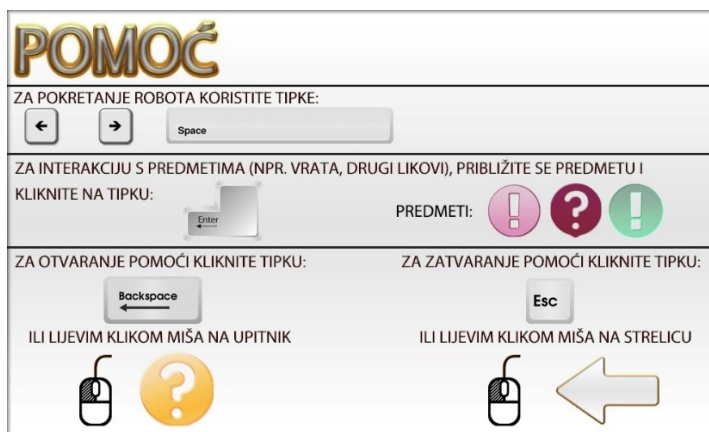
Scena 4 – Prvi Hodnik

U ovoj sceni glavni lik INFO nalazi se u ulaznom hodniku Svijeta Informatike i omogućena nam je kontrola. INFO-a kontroliramo pomoću strelica lijevo i desno na tipkovnici, te tipkom Space možemo izvršiti animaciju skakanja.



Slika 76 Scena 4 - Prvi Hodnik

U gornjem desnom kutu scene nalazi se žuti upitnik na kojeg možemo kliknuti lijevim klikom miša ili tipkom Backspace na tipkovnici. Klikom na žuti upitnik otvara se dodatni meni s uputama kako se kretati i kako izvršiti interakciju s objektima. Pokretanjem glavnog lika prema vratima gdje se nalazi zeleni simbol uskličnika prelazimo u sljedeću scenu.



Slika 77 Dodatni Meni za Pomoć

Scena 5 – Centralna Prostorija

Ulaskom u Centralnu Prostoriju igra se transformira iz 2D platformera u 3D izometričnu igru iz ptičje perspektive. Tom promjenom glavni lik dobiva nove kontrole

pomoću kojih se može kretati u svim smjerovima. Za upravljanje glavnog lika koriste se sve strelice na tipkovnici, dok je mogućnost skakanja u toj sekciji igre zabranjena.

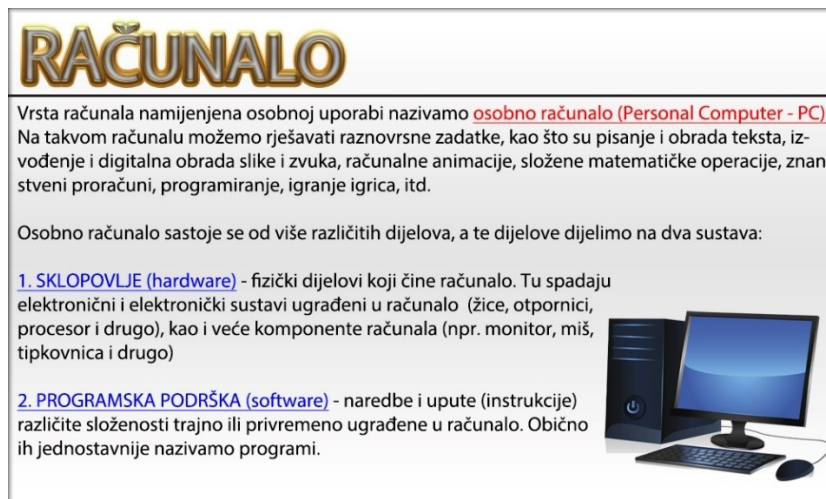
U Centralnoj Prostoriji nailazimo na NPC lika zvanog DIO s kojim možemo komunicirati kada mu priđemo i kliknemo tipku ENTER na tipkovnici. On nam objašnjava pojedinosti o Svijetu Informatike, te nas usmjerava na put i pomaže u istraživanju.

DIO: Dobrodošli u Svijet Informatike!!!! Ja sam DIO. Ovdje možete naučiti i savladati Uvodni dio Informatike: Sklopovlje i povezivanje na mrežu. Kada savladate obje teme otvorit će vam se izlaz iz ovog svijeta. Trenutno se nalazite u centralnoj prostoriji iz koje možete pristupiti prostorijama: Kula Sklopovlja, Mrežni Centar i Mini igre. Svaka od navedenih prostorija obuhvaća određeno gradivo Uvoda u Informatiku. Trenutno vam je otvorena samo Kula Sklopovlja. Kako bi otvorili druge prostorije morate prvo osvojiti Kulu. SRETNO!!!!



Slika 78 Scena 5 - Centralna Prostorija

Osim DIO-a, na sceni se nalazi računalo kao objekt s kojim možemo izvršiti interakciju. Približavanjem računalu i klikom na tipku ENTER otvara se prozor koji sadrži informacije o računalu i njegovoj namjeni i podijeli.



Slika 79 Prozor s informacijama o računalu

U sceni su prikazane tri prostorije, Kula Sklopovlja na lijevoj strani, Mrežni Centar u desnom gornjem kutu i Mini Igre u desnom donjem kutu. Svaka od prostorija obrađuje određeno gradivo i prema vodiču otvorena je samo Kula Sklopovlja. Kako bi otvorili sljedeću prostoriju moramo prvo osvojiti kulu.

Scena 6 – Kula Sklopovlja: Prva prostorija

Ulaskom u Kulu Sklopovlja igra se transformira nazad u 2D platformer. U prvoj prostoriji nalazi se crtež s popisom dijelova sklopovlja računala i podjelom na ulazne i izlazne jedinice, te središnju jedinicu.



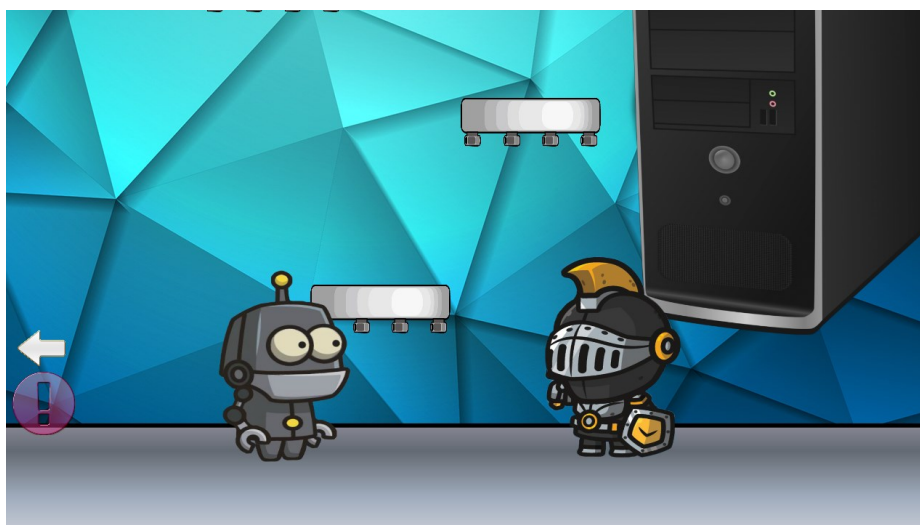
Slika 80 Scena 6 - Kula Sklopovlja: Prva prostorija

Na desnoj strani scene nalazi se strelica te simbol uskličnika. Dodirom uskličnika i klikom na tipku ENTER prelazimo u drugu prostoriju Kule Sklopovlja.

Scena 7 – Kula Sklopovlja: Druga prostorija

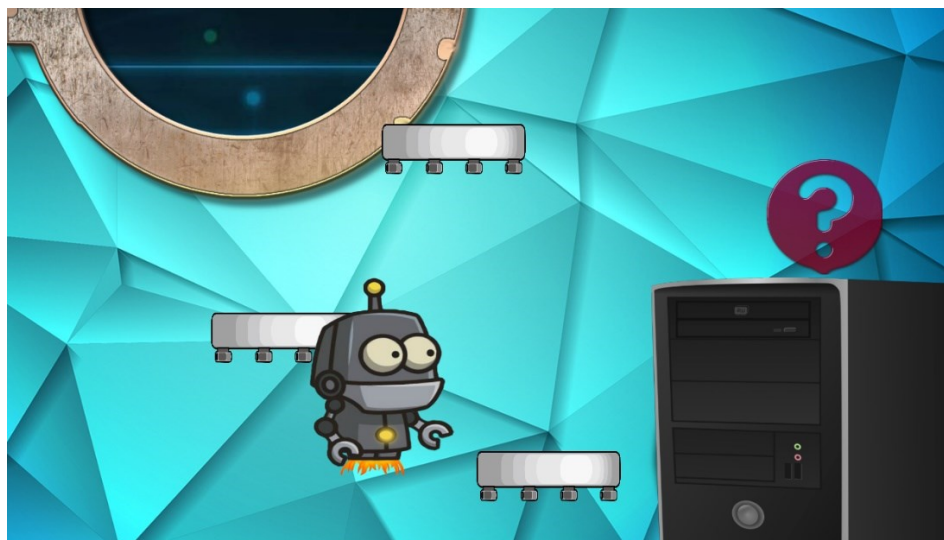
U drugoj prostoriji nailazimo na vodiča Kule Sklopovlja zvanog Vitez Kule. Interakcijom s njim, on nam objašnjava pojedinosti zadatka koji treba riješiti kako bi se osvojila kula.

Vitez Kule: Dobrodošli u KULU SKLOPOVLJA! Ja sam VITEZ KULE. Ovdje možete naučiti o fizičkim dijelovima koji čine računalo, odnosno sklopovlju računala ili hardveru (hardware na engleskom). Sklopovlje računala možemo podijeliti u tri kategorije: SREDIŠNJU JEDINICU, ULAZNE JEDINICE i IZLAZNE JEDINICE. Kako bi osvojili kulu i izašli iz nje, morate proučiti sve dijelove računala i točno odgovoriti na pitanja koja će vam na vrhu kule postaviti ČUVAR KULE.



Slika 81 Scena 7 - Kula Sklopovlja: Druga prostorija

Scena je napravljena kao visoka prostorija s razbacanim platformama na koje možemo skakati i time se penjati do vrha kule. Tijekom penjanja nailazimo na dijelove sklopovlja računala i potrebno je svakom dijelu pristupiti kako bi prikupili dovoljno informacija za završni kviz na vrhu kule. Svaki dio sklopovlja je platforma na koju možemo skočiti te dodirom upitnika i klikom na tipku ENTER otvara se prozor s informacijama o tom dijelu.



Slika 82 Penjanje do vrha kule i istraživanje sklopovlja računala

SREDIŠNJA JEDINICA

Središnja jedinica računala je kućište sa svim dijelovima koji se u njemu nalaze i na koje se svi izlazni i ulazni dijelovi spajaju. Središnja jedinica sastoji se od više dijelova koji omogućavaju rad računala:

 <p>a) Kućište: Metalna kutija koja čini oklop centralne jedinice. U kućištu se nalaze ostali dijelovi središnje jedinice</p>	 <p>b) Matična ploča: Dio središnje jedinice koji se nalazi unutar kućišta i na koji se spajaju svi dijelovi računala</p>	 <p>c) Procesor: Središnja jedinica za obradu podataka koja se sastoji od: 1. upravljačke jedinice - upravlja dijelovima računala 2. aritmetičke jedinice - izvršava obradu podataka</p>	 <p>d) Glavna memorija: U glavnu ili unutrašnju memoriju ubrajamo dvije vrste memorije: 1. ROM - trajno pohranjeni podaci važni za rad 2. RAM - memorija u kojoj su privremeno učitavaju programi i datoteke</p>
---	---	---	--

Slika 83 Prozor s informacijama koji se otvori interakcijom s određenim dijelom sklopovlja

Na vrhu kule nailazimo na NPC lik zvan Čuvar Kule koji daje kratki kviz koji obuhvaća sve što smo naučili interakcijom sa sklopovljem računala.



Slika 84 Čuvar Kule

Kviz je napravljen pomoću Yarn Editora u kojem se može kreirati dijalog s jednostavnim pitanjima i odgovorom u obliku višestrukog izbora gdje su ponuđeno više odgovora od kojih je samo jedan točan.

Čuvar Kule (Kviz):

Ha Ha Ha! Ja sam ČUVAR KULE!

Ako želite izaći iz ove kule morate točno odgovoriti na moja pitanja. Tijekom razgovora pritisnite tipku SPACE (najduža tipka na tipkovnici) kako bi nastavili razgovor.

1. Koji od navedenih dijelova je fizički dio središnje jedinice računala? izaberite jedan od ponuđenih odgovora i opet pritisnite tipku SPACE.
 - a) MONITOR → netočno
 - b) TIPKOVNICA → netočno
 - c) PROCESOR → točno
 - d) OPERACIJSKI SUSTAV → netočno

Točno!!! Idemo dalje!!!

2. Koji od navedenih dijelova NIJE Izlazna Jedinica?

- a) ZVUČNIK → netočno
- b) PISAČ → netočno
- c) MIŠ → točno
- d) MONITOR → netočno

Točno!!! Idemo dalje!!!

3. Koji od navedenih dijelova je Izlazna Jedinica?

- a) SLUŠALICE → točno
- b) TIPKOVNICA → netočno
- c) MIŠ → netočno
- d) KUĆIŠTE → netočno

Točno!!! Idemo dalje!!!

4. Zvučna kartica je zadužena za obradu zvučnih signala!

- a) TOČNO → točno
- b) NETOČNO → netočno

Točno!!! Idemo dalje!!!

5. Dvije glavne vrste pisača su: TINTNI i LASERSKI!

- a) TOČNO → točno

b) NETOČNO → netočno

Točno!!! Idemo dalje!!!

6. Koliko iznosi Jedan INČ?

a) 3,52 → netočno

b) 2,54 → točno

c) 4,32 → netočno

d) 1,52 → netočno

Točno!!! Idemo dalje!!!

7. RAM je memorija u kojoj su trajno pohranjeni podaci!

a) TOČNO → netočno

b) NETOČNO → točno

ČESTITAM!!! USPJELI STE RIJEŠITI KVIZ!!!

MOŽETE IZAĆI IZ KULE!!! HA HA HA HA HAA

Nakon uspješno riješenog kviza kula je osvojena i igra nas prebacuje nazad u Centralnu Prostoriju koja je za nijansu izmijenjena. Osvajanjem kule otvorio se Mrežni Centar, ali isto tako upalio se jedan dio centralne žute lampe u središtu prostorije. Pokraj lampe nailazimo na vodiča koji nam čestita na uspjehu osvajanja kule. Isto tako igra je ponovno u 3D izometričnom obliku.

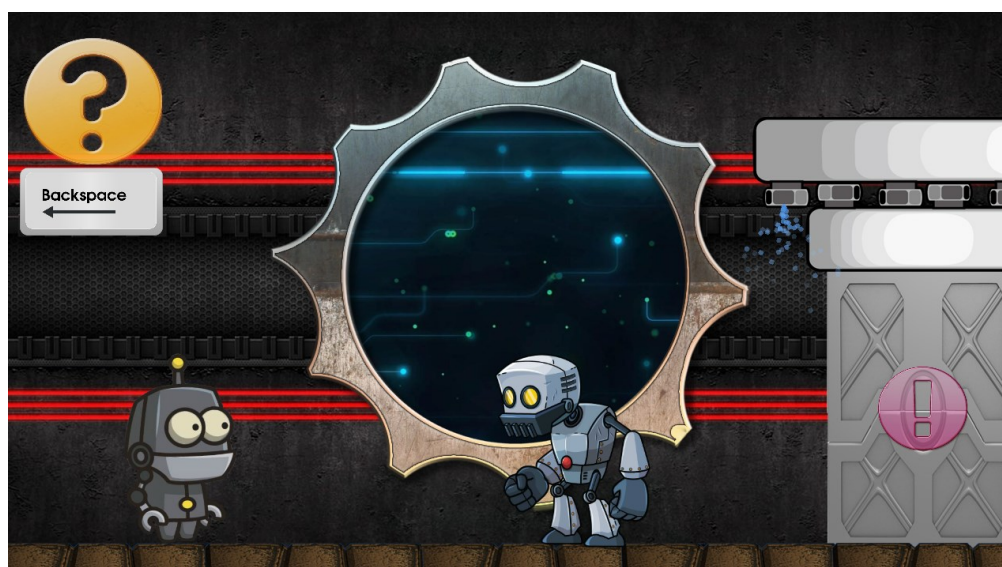
DIO: Čestitam! Osvojili ste Kulu Sklopovlja i time savladali gradivo sklopovlja računala. Otvoren vam je put u Mrežni Centar, ali ako želite možete ponovno ući u Kulu i ponoviti gradivo.



Slika 85 Centralna prostorija nakon Kule Sklopovlja

Scena 8 – Mrežni Centar: Prva prostorija

Ulaskom u novo otvorenu prostoriju prelazimo u novu scenu. U prvoj prostoriji Mrežnog Centra nalazimo na vodiča Mrežnog Centra zvanog KOR koji opisuje prostoriju u kojoj se nalazimo, te o pojedinostima računalne mreže i povezivanje u mrežu.



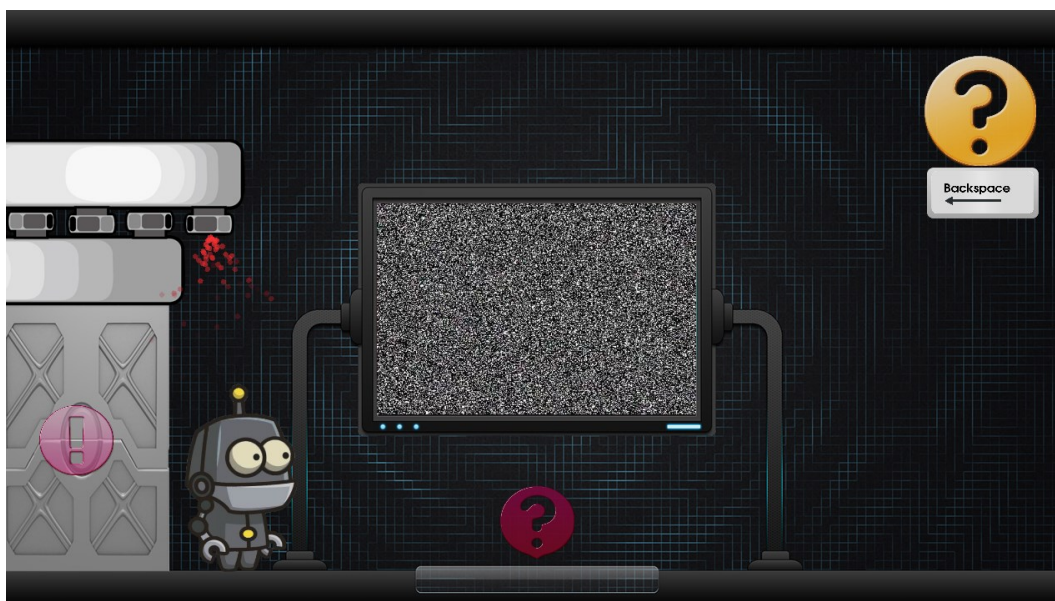
Slika 86 Scena 8 - Mrežni Centar: Prva Prostorija

KOR: Dobrodošli u MREŽNI CENTAR! Ja sam KOR! Ovdje možete naučiti o povezivanju računala u mrežu. Velika prednost računala je što se može povezivati s drugim digitalnim uređajima i tako s njima razmjenjivati podatke. Preko MREŽE računalo se može povezivati s drugim računalima, mobitelima, tabletima itd. Kako bi omogućili povezivanje između računala i digitalnih uređaja potrebno je izgraditi RAČUNALNU MREŽU.

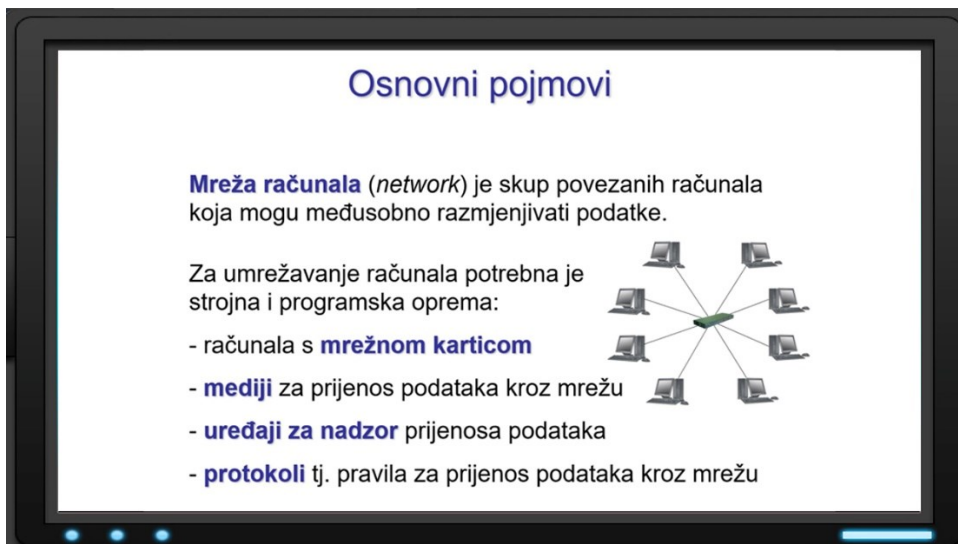
Nakon razgovora s vodičem na desnoj strani nalaze se vrata sa simbolom uskličnika koja vode u drugu prostoriju Mrežnog Centra.

Scena 9 – Mrežni Centar: Druga prostorija

U drugoj prostoriji nailazimo na Monitor koji možemo aktivirati. Dodirom simbola upitnika ispod monitora, aktiviramo ekran i prelazimo na dodatnu scenu prikazanu kao prezentacija od 13 slajda. Preko prezentacije obrađujemo nastavnu jedinicu računalne mreže i povezivanja u mrežu, i time se pripremamo za kviz.

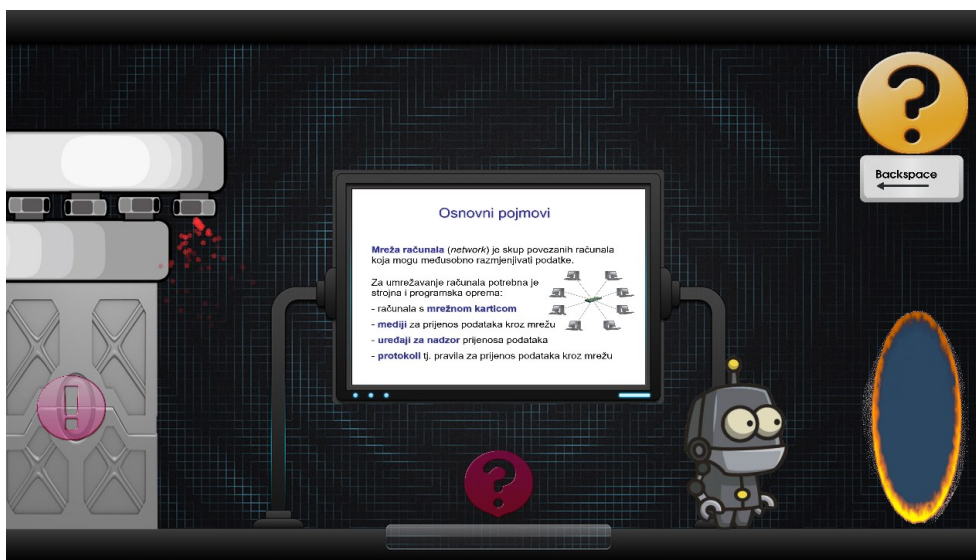


Slika 87 Scena 9 - Mrežni Centar: Druga prostorija



Slika 88 Dodatna scena – Presentacija

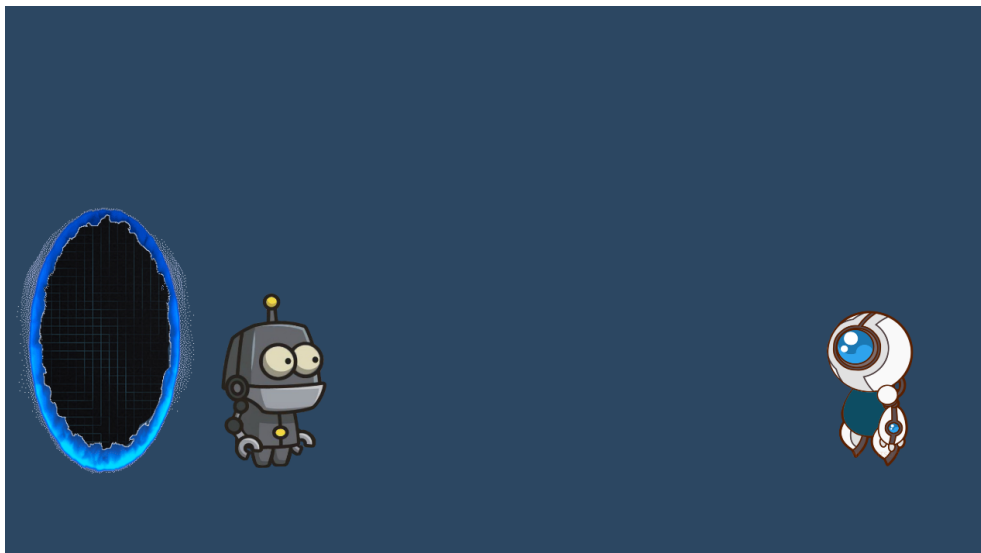
Nakon prezentacije igra nas vraća nazad u drugu prostoriju u kojoj se otvorio portal koji vodi u posljednju prostoriju Mrežnog Centra gdje na očekuje kviz.



Slika 89 Portal koji vodi u treću prostoriju

Scena 10 – Mrežni Centar: Treća prostorija

Ulaskom u portal prelazimo u treću i posljednju prostoriju Mrežnog Centra gdje nailazimo na čuvara Mrežnog Centra zvanog KIR. On nam daje kratki kviz koji obuhvaća sve što smo naučili s prezentacije u prijašnjoj prostoriji.



Slika 90 Scena 10 - Mrežni Centar: Treća prostorija

KIR (Kviz):

Pozdravljeni! Ja sam KIR!

Ako želite izaći iz mrežnog centra morate točno odgovoriti na moja pitanja. Tijekom razgovora pritisnite tipku SPACE (najduža tipka na tipkovnici) kako bi nastavili razgovor.

1. Mreža računala je skup povezanih računala koja mogu međusobno razmjenjivati podatke (izaberite jedan od ponuđenih odgovora i opet pritisnite tipku SPACE).
 - a) TOČNO → točno
 - b) NETOČNO → netočno

Točno!!! Idemo dalje!!!

2. Koja vrsta mreže NIJE računalna mreža?
 - a) LOKALNA MREŽA → netočno
 - b) MENTALNA MREŽA → točno
 - c) ŠIROKO RASPROSTRANJENA MREŽA → netočno

d) GRADSKA MREŽA → netočno

Točno!!! Idemo dalje!!!

3. Mreža u kojoj sva računala mogu imati ulogu poslužitelja i ulogu klijenta zove se:

a) KORISNIK/POSLUŽITELJ MREŽA → netočno

b) ŠIROKO RASPROSTRANJENA MREŽA → netočno

c) MREŽA RAVNOPRAVNIH KORISNIKA → točno

d) LOKALNA MREŽA → netočno

Točno!!! Idemo dalje!!!

4. koja vrsta mreže je WAN?

a) LOKALNA MREŽA → netočno

b) GRADSKA MREŽA → netočno

c) KORISNIK/POSLUŽITELJ MREŽA → netočno

d) ŠIROKO RASPROSTRANJENA MREŽA → točno

Točno!!! Idemo dalje!!!

5. Koja vrsta mreže je MAN?

a) ŠIROKO RASPROSTRANJENA MREŽA → netočno

b) LOKALNA MREŽA → netočno

c) GRADSKA MREŽA → točno

c) MREŽA RAVNOPRAVNIH KORISNIKA → netočno

Točno!!! Idemo dalje!!!

6. U lokalnoj mreži LAN, svako je računalo mrežnim kabelom spojeno s mrežnim preklopnikom.

a) TOČNO → točno

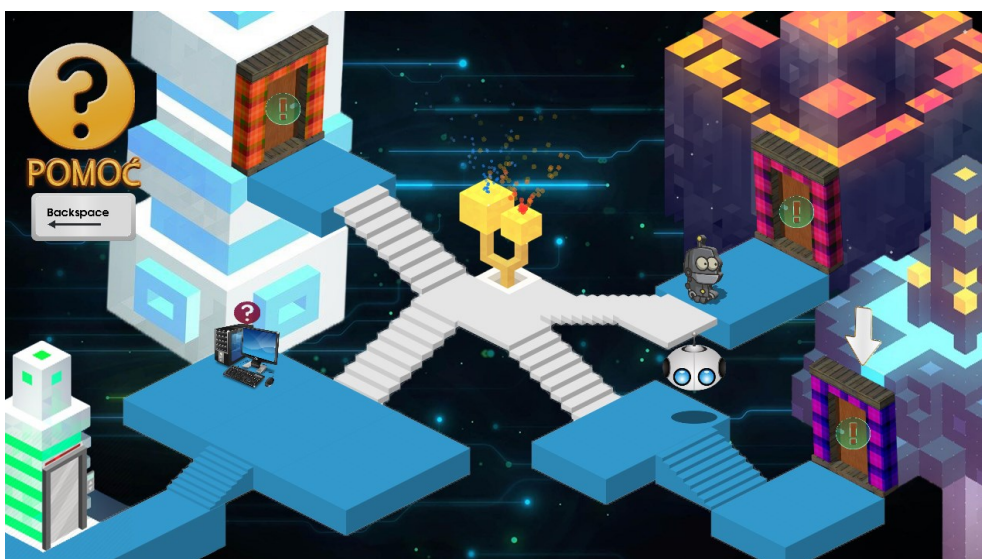
b) NETOČNO → netočno

ČESTITAM!!! USPJELI STE RIJEŠITI KVIZ!!!

MOŽETE NASTAVITI DALJE!!! SRETNO!!!

Nakon uspješno riješenog kviza Mrežni Centar je osvojen i igra nas prebacuje nazad u Centralnu Prostoriju koja je opet za nijansu izmijenjena. Osvajanjem Mrežnog Centra otvorila se posljednja prostorija za Mini Igre, ali isto tako upalio se i drugi dio centralne žute lampe u središtu prostorije. Pokraj treće prostorije nailazimo na vodiča koji nam čestita na uspjehu osvajanja Mrežnog Centra. Isto tako igra je ponovno u 3D izometričnom obliku.

DIO: Čestitam! Osvojili ste Mrežni Centar i time savladali gradivo povezivanja računala na mrežu. Otvoren vam je put u prostoriju mini igara. Kako bi se otvorio IZLAZ iz ovog svijeta, morate riješiti barem jednu od ponuđenih mini igara.



Slika 91 Centralna prostorija nakon Mrežnog Centra

Scena 11 – Mini Igre

Ulaskom u novo otvorenu prostoriju prelazimo u novu scenu zvanu Mini Igre. U prostoriji Mini Igre nailazimo na tri vrata i svaka su vrata ulaz u tri kratke interaktivne igre. Na početku otvorena su samo prva vrata s natpisom Parovi, dok su druga dva vrata zatvorena.



Slika 92 Scena 11 - Mini Igre

Scena 12 - Parovi

Ulaskom u prva vrata pokrećemo mini igru zvanu Parovi. Igra se odnosi na pogađanje parova gdje moramo spojiti sličicu sklopovlja računala s njegovim nazivom. Kada se svi parovi pronađu igra naš šalje nazad u hodnik gdje ponovno možemo odabrati željenu igru.

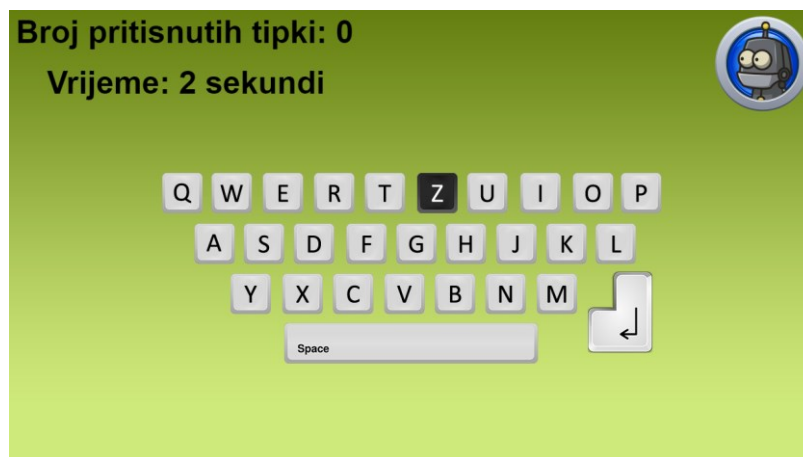


Slika 93 Mini igra - Parovi

Nakon uspješno riješene mini igre Parovi, igra nas vraća u prijašnju prostoriju ali ovoga puta sa svim otključanim vratima.

Scena 13 – Vježba s tipkovnicom

Ulaskom u druga vrata s natpisom Vježba s tipkovnicom prelazimo u drugu mini igru. Druga igra se odnosi na mjerenju brzine prstometata odnosno brzinu tipkanja tipki na tipkovnici. Scena prikazuje tipke na tipkovnici te označava koju tipku treba stisnuti. Zadatak igre je pritisnuti što veći broj tipki u 60 sekundi. Nakon što vrijeme istekne igra prikazuje korisnikov broj točno pritisnutih tipki te nas ponovno vraća u prijašnju prostoriju za odabir igre.



Slika 94 Mini igra - Vježba s tipkovnicom

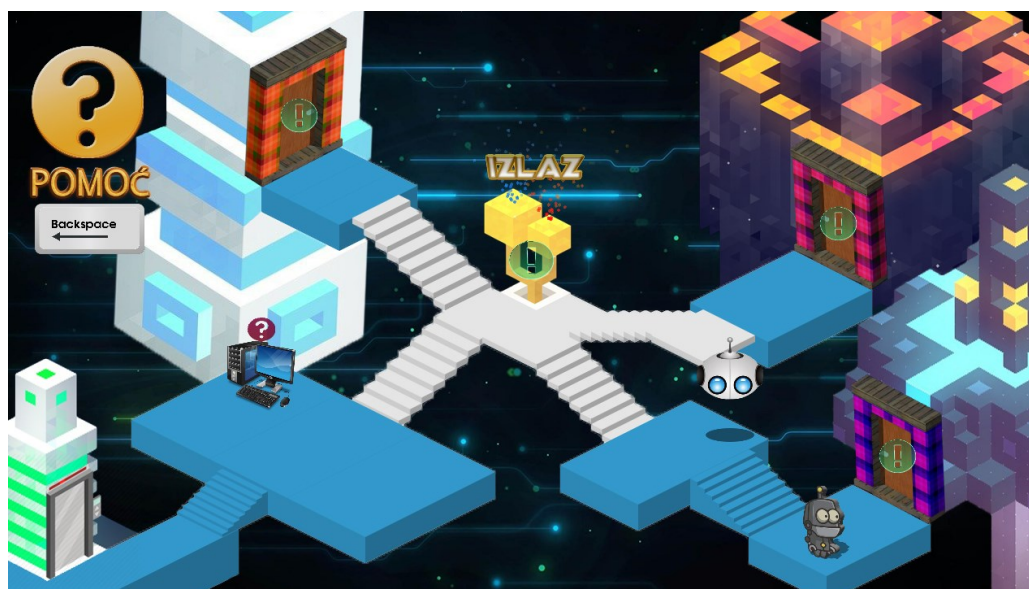
Scena 14 – Ulazno Izlazne jedinice

Treća i posljednja mini igra odnosi se na razvrstavanje sklopovlja računala u izlazne i ulazne jedinice. Na ekranu su prikazani dijelovi sklopovlja kao objekti koje možemo povlačiti u željeno područje. Navedena su dva područja „Izlazni uređaji“ i „Ulazni uređaji“ i u svako od područja moramo razvrstati odgovarajuće uređaje. Nakon uspješnog razvrstavanja igra nas vraća nazad u prostoriju za odabir igre.



Slika 95 Mini Igra - Ulazno Izlazne jedinice

Nakon povratka u prostoriju za odabir mini igre, na lijevoj strani se kreirao simbol uskličnika koji nas može odvesti nazad u Centralnu Prostoriju. Uspješnim rješenjem jedne ili više mini igra otvorio se IZLAZ iz Svijeta Informatike. Interakcijom s uskličnikom na lijevoj strani hodnika igra nas prebacuje nazad u Centralnu Prostoriju koja je opet za nijansu izmijenjena.



Slika 96 Centralna Prostorija nakon Mini Igre

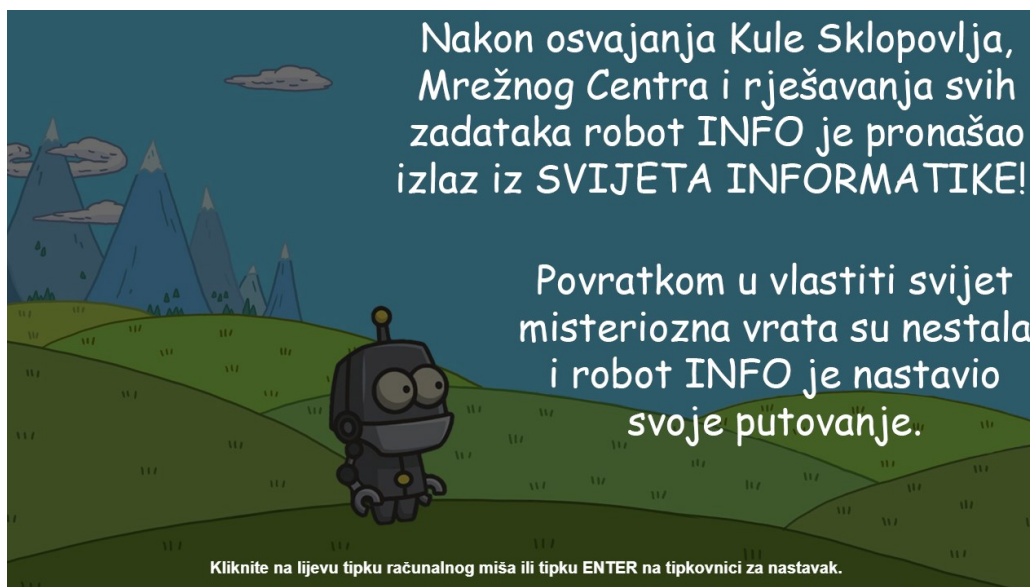
Rješavanjem Mini Igre aktivirala se žuta lampa u središtu prostorije koja sada vodi u izlaz iz Svijeta Informatike, a time i u kraj igre. Pokraj treće prostorije nailazimo i na vodiča koji nam čestita na našem uspjehu.

DIO: Čestitam! Riješili ste sve zadatke i otvorio vam se IZLAZ. Ako želite možete ponovno istražiti sve prostorije i ponoviti gradivo. Doviđenja!

Scena 15 – Završna Scena

Dodirom zelenog uskličnika na žutoj lampi prelazimo u završnu scenu koja nam prikazuje glavnog lika kako je izašao iz Svijeta Informatike.

Priča: Nakon osvajanja Kule Sklopovlja, Mrežnog Centra i rješavanja svih zadataka robot INFO je pronašao izlaz iz SVIJETA INFORMATIKE! Povratkom u vlastiti svijet misteriozna vrata su nestala i robot INFO je nastavio svoje putovanje.

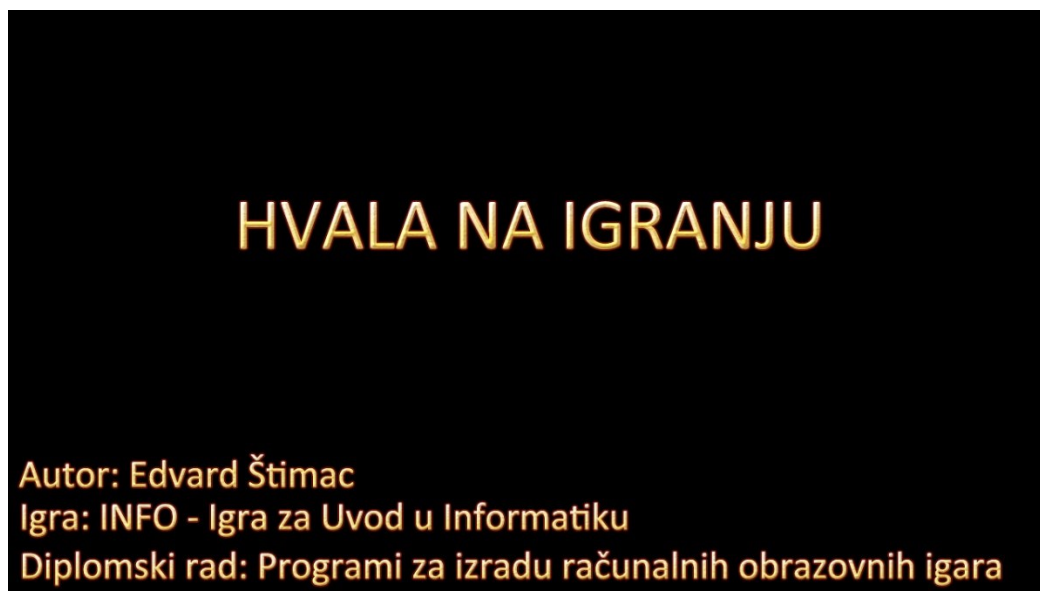


Slika 97 Scena 15 - Završna Scena

Klikom lijeve tipke računalnog miša ili tipkom ENTER na tipkovnici ulazimo u posljednju scenu.

Scena 16 – Credits

U posljednjoj sceni autor se zahvaljuje na igranju te se iznose dodatne informacije o autorovom imenu, nazivu igre, i nazivu diplomskog rada u kojem je igra dio projekta.



Slika 98 Scena 16 – Credits

7. Zaključak

Računalne igre dio su svakodnevnog života za mnoge ljude i predstavljaju aspekt zabave i opuštanja za sve naraštaje. No isto tako se računalne igre mogu potencijalno koristiti i u svrhu obrazovanja te, prema određenim istraživanjima iz područja *Game Based Learning*, imaju pozitivan učinak na razvoju vještina poput rješavanja problema, prikupljanja informacija, jačanju motivacije i aktivnijem sudjelovanju pri učenju.

Brzim razvojem tehnologije razvijaju se mnoge novije igre za različite svrhe uključujući i za obrazovanje. Isto tako razvijaju se i novi programi za izradu igara koji su sve više pristupačniji i lakši za korištenje. Time je velik broj programa danas gotovo svima dostupan za korištenje.

Jedan od najpristupačnijih, besplatnih programa GDevelop 5. Program je brz i fleksibilan i pomoću njega mogu se kreirati 2D video igre različitih žanrova bez dodatnog znanja programiranja. Osnovni koncept rada programa je jednostavan i bazira se na nekoliko važnijih čimbenika: objekti, događaji i ponašanja. Program ima pregledno grafičko sučelje koje znatno olakšava pri navigaciji i upravljanju editorima za izrađivanje scena igre, raznih objekta i njihovih ponašanja, te događaje. U usporedbi s ostalim besplatnim i jednostavnim programima koji su svima dostupni za korištenje, poput Construct, Game Maker Studio, eAdventure i slično, GDevelop je znatno fleksibilniji i slobodniji za manipuliranje. Svaki objekt u GDevelop programu može se manipulirati i njegovo ponašanje prilagoditi našim željama. Stoga GDevelop nema ograničenja poput programa kao što je eAdventure. Jedini minus GDevelop programa je uključivanje stabla dijaloga koji je vrlo ključan dodatak u kreiranju informativnih igara u koje spada i edukativna igra. GDevelop koristi vanjski softver za korištenje dijaloga, te je implementacija u igru znatno teža u usporedbi s programom poput eAdventure. Ipak, program je u konstantom razvoju i svakog mjeseca dobiva poboljšanja i nove dodatke od strane autora, što može rezultirati poboljšanjem implementacije dijaloga u igru.

Ovim diplomskim radom, pomoću analiziranog GDevelop programa, uspješno je izrađena edukativna igra „INFO – Igra za Uvod u Informatiku“ koja obrađuje uvodno nastavno gradivo „Osnove Računala“ za 5. razred osnovne škole te obuhvaća sve važnije čimbenike programa kao i dodatne mogućnosti ponuđene za bolji izgled.

8. Literatura

- [1] Beavis, C., O'Mara, J., McNeice, L., *Digital games: literacy in action*, AATE Interface Series, 2012.
- [2] De Freitas, S., *Learning in Immersive worlds: A review of game-based learning*, JISC, 2006.
- [3] Kniewald, I., Galešev, V., Sokol, G., Vlahović, V., Kager, D., Kovač, H., *Informatika+ 5 – Udžbenik iz informatike za 5. razred osnovne škole*, SysPrint, Zagreb, 2019., link web stranice: <https://sysprint.hr/eudzbenici/inf5u2019/>
- [4] Novaković, A., *Učenje uz pomoć didaktičkih računalnih igara*, Sveučilište u Rijeci – Odjel za informatiku, 2015.
- [5] Pavkov, S., *Usporedba programskih alata za izradu didaktičkih igara*, Sveučilište u Rijeci – Odjel za informatiku, 2015.
- [6] Perenič, S., Femc, I., Humar, K., Ferlin, S., *Projektna dokumentacija - Didaktična Računalniška Igra*, Univerza v Ljubljani Pedagoška fakulteta, 2013.
- [7] Perrotta, C., Featherstone, G., Aston, H. and Houghton, E., *Game-based Learning: Latest Evidence and Future Directions*, NFER Research Programme: Innovation in Education, 2013.
- [8] Pho, A., Dinscore, A., *Game-Based Learning*, Tips and Trends, 2015.
- [9] Poje, D., *Izrada didaktičkih računalnih igara u alatu eAdventure*, Sveučilište u Rijeci – Odjel za informatiku, 2016.
- [10] Rugelj, J., Zapušek, M., *Achieving teachers' competences in the serious game design process*, Proceedings of the 8th European Conference on Games Based Learning, Academic Conferences and Publishing International Limited, Berlin, 2014.
- [11] Vasiljevič, G., Štrukelj, T., Šušterič, M., *Dokumentacija projekta - Otov Indeks*, Univerza v Ljubljani Pedagoška fakulteta, 2013.

[12] Whitton, N., *Learning with Digital Games – A practical Guide to Engaging Students in Higher Education*, Routledge, 2010.

[13] GDevelop 5, GDEVELOP WIKI – OPEN-SOURCE, CONDINGLESS GAME CREATOR, 2019., link web stranice:

<http://wiki.compilgames.net/doku.php/gdevelop5/start>

9. Slike i tablice

Slike:

Slika 1 Početna stranica GDevelop 5	11
Slika 2 Kreiranje Nove Igre	12
Slika 3 Radno Sučelje	13
Slika 4 Alatna traka.....	13
Slika 5 Voditelj projekta (Project Manager)	14
Slika 6 Editor scena.....	15
Slika 7 Editor objekta.....	15
Slika 8 Panel za svojstva instance	16
Slika 9 Editor slojeva	17
Slika 10 Dodavanje efekta	17
Slika 11 Scena bez sloja s efektom	18
Slika 12 Scena s efektom na sloju.....	18
Slika 13 Akcija koja mijenja vidljivost efekta "Efekt" na jednom sloju.....	18
Slika 14 Kartica za Editor događaja.....	19
Slika 15 Editor događaja s novim funkcijama na alatnoj traci	19
Slika 16 Novi događaj.....	20
Slika 17 Novi Pod-događaj	20
Slika 18 Ubacivanje komentara u događaj.....	20
Slika 19 Grupiranje događaja.....	21
Slika 20 Debugger.....	21
Slika 21 Promjena određenih svojstva instanci korištenjem debuggera	22
Slika 22 Kreiranje GDevelop računa	22
Slika 23 Ubacivanje objekta u Editor objekta.....	23
Slika 24 Tipovi objekta	23

Slika 25 Instanca - Objekt prikazan u sceni	24
Slika 26 Definiranje varijable instance	25
Slika 27 Kreiranje Sprite objekta	26
Slika 28 Prozor za određivanje svojstva tijekom kreiranja sprite objekta	26
Slika 29 Animiranje objekta.....	27
Slika 30 Kreirani Sprite objekt na sceni.....	27
Slika 31 Kreiranje Tiled Sprite objekta.....	28
Slika 32 Tiled sprite objekt – jedna pločica (lijeva strana) i skup pločica (desna strana) koju dobijemo širenjem jedne pločice	28
Slika 33 Prozor za određivanje svojstva tijekom kreiranja tiled sprite objekta	28
Slika 34 Kreiranje emitera partikla	29
Slika 35 Prozor za određivanje svojstva tijekom kreiranja emitera partikla.....	29
Slika 36 Emiter čestica u sceni.....	30
Slika 37 Kreiranje Text objekta	30
Slika 38 Događaj koji mijenja veličinu fonta Text objekta "Score"	31
Slika 39 Kreiranje Panel Sprite objekta	31
Slika 40 Kreiranje objekta za unos teksta	31
Slika 41 Kreiranje objekta za crtanje oblika	32
Slika 42 Prozor za određivanje svojstva tijekom kreiranja Shape painter objekta	32
Slika 43 BBCode tagovi.....	33
Slika 44 Dodavanje ponašanja u objekt	33
Slika 45 Zadani tipovi ponašanja objekta	34
Slika 46 Kreiranje platformskog ponašanja	34
Slika 47 Kreiranje ponašanja platformera.....	35
Slika 48 Kreiranje ponašanja za uništavanje instance tijekom izlaska iz scene	36
Slika 49 Kreiranje ponašanja vučnog objekta.....	36

Slika 50 Kreiranje ponašanja za Top-down kretanje objekta.....	36
Slika 51 Kreiranje Anchor ponašanje za objekt.....	37
Slika 52 Kreiranje Pathfinding ponašanja za objekt	37
Slika 53 Kreiranje ponašanja Pathfinding prepreke za objekt	38
Slika 54 Kreiranje Tween ponašanja za objekt	41
Slika 55 Korištenje Tween ponašanje u editoru događaja	41
Slika 56 Editor događaja na sučelju	42
Slika 57 Prazni prozor editora događaja s poljima za unos uvjeta i akcije	42
Slika 58 Alati za kreiranje događaja	42
Slika 59 Editor Događaja sa zadanim uvjetima i akcijama.....	42
Slika 60 Prazna tablica gdje se dodaju uvjeti (lijevo) i akcije (desno)	43
Slika 61 Lista uvjeta.....	43
Slika 62 Korištenje uvjeta i akcije za mijenjanje scene nakon klika START.....	44
Slika 63 Kreiranje vanjskog događaja.....	45
Slika 64 Kreiranje Link događaja - Povezivanje vanjskog događaja sa scenom	45
Slika 65 Grupiranje događaja.....	46
Slika 66 Grupiranje događaja u grupe Vrata i Meni	46
Slika 67 JavaScript Code terminal u editoru.....	47
Slika 68 Kreiranje dijalog stabla pomoću Yarn editora.....	51
Slika 69 Export igre	52
Slika 70 Platforme za objavljivanje igre	52
Slika 71 Skica mape scena	62
Slika 72 Scena 1 - Glavni Meni Igre.....	64
Slika 73 Dodatna Scena - Opis Igre	65
Slika 74 Scena 2 - Uvodni Opis Igre.....	65
Slika 75 Scena 3 - Uvodna Priča.....	66

Slika 76 Scena 4 - Prvi Hodnik.....	67
Slika 77 Dodatni Meni za Pomoć.....	67
Slika 78 Scena 5 - Centralna Prostorija.....	68
Slika 79 Prozor s informacijama o računalu	69
Slika 80 Scena 6 - Kula Sklopovlja: Prva prostorija.....	69
Slika 81 Scena 7 - Kula Sklopovlja: Druga prostorija	70
Slika 82 Penjanje do vrha kule i istraživanje sklopovlja računala	71
Slika 83 Prozor s informacijama koji se otvori interakcijom s određenim dijelom sklopovlja	71
Slika 84 Čuvar Kule	72
Slika 85 Centralna prostorija nakon Kule Sklopovlja.....	75
Slika 86 Scena 8 - Mrežni Centar: Prva Prostorija	75
Slika 87 Scena 9 - Mrežni Centar: Druga prostorija	76
Slika 88 Dodatna scena – Presentacija.....	77
Slika 89 Portal koji vodi u treću prostoriju	77
Slika 90 Scena 10 - Mrežni Centar: Treća prostorija.....	78
Slika 91 Centralna prostorija nakon Mrežnog Centra.....	80
Slika 92 Scena 11 - Mini Igre	81
Slika 93 Mini igra - Parovi.....	81
Slika 94 Mini igra - Vježba s tipkovnicom	82
Slika 95 Mini Igra - Ulazno Izlazne jedinice	83
Slika 96 Centralna Prostorija nakon Mini Igre	83
Slika 97 Scena 15 - Završna Scena	84
Slika 98 Scena 16 – Credits	85

Tablice:

Tablica 1 Popis likova..... 62