Izrada 3D računalne igre u Unreal Engine 4 razvojnom okruženju

Pečatnik, Krunoslav-Stjepan

Undergraduate thesis / Završni rad

2018

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University North / Sveučilište Sjever**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:122:706151

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-04-19**

Repository / Repozitorij:

University North Digital Repository

Sveučilište
Sjever

Završni rad br. 574/MM/2018

Izrada 3D računalne igre u Unreal Engine 4 razvojnom okruženju

Krunoslav-Stjepan Pečatnik, 0833/336

Varaždin, rujan 2018. godine

Sažetak

U ovom radu opisan je kompletni postupak izrade 3D računalne igre korištenjem Unreal Engine 4 razvojnog okruženja. Opisan je postupak od prvog otvaranja programa pa do završnog pakiranja igre. Unreal Engine 4 omogućuje nam korištenje tehnika 3D modeliranja, animiranja, programiranja te spajanja svega u jedan finalni proizvod što rezultira računalnom igrom. Valja napomenuti da ovaj završni rad predstavlja postupak izrade računalne igre korištenjem blueprint vizualnog skriptiranja što znači da je igra napravljena bez ijedne linije pisanog koda. Također korišteni su i ostali alati kao što su Adobe Photoshop pomoću kojih su uređivane pojedine teksture i dizajnirane grafike korisničkog sučelja i glavnog menija igre.

Ključne riječi: animiranje, blueprint vizualno skriptiranje, modeliranje, Unreal Engine 4

Abstract

This work describes the complete process of making a 3D computer game using the Unreal Engine 4. The procedure is described from the first opening of the program to the final packaging of the game. Unreal Engine 4 allows us to use 3D modeling, animating, programming and merging techniques into one final product resulting in a computer game. It should be noted that this work represents the process of making a computer game using a blueprint visual scripting, meaning that the game is made without writing single line of code manually. Other tools, such as Adobe Photoshop, have been used to edit individual textures and to design user interface graphics under the game's main menu.

Keywords: animation, blueprint visual scripting, modeling, Unreal Engine 4

Sveučilište Sjever
Sveučilišni centar Varaždin
104. brigade 3, HR-42000 Varaždin

Prijava završnog rada

Definiranje teme završnog rada i povjerenstva

ODJEL: Odjel za multimediju, oblikovanje i primjenu

PRISTUPNIK: Krunoslav-Stjepan Pečatnik | MATIČNI BROJ: 0833/336

DATUM: 27.03.2018. | KOLEGIJ: 3D modeliranje

NASLOV RADA: Izrada 3D računalne igre u Unreal Engine 4 razvojnom okruženju

NASLOV RADA NA ENGL. JEZIKU: Creating a 3D computer game in Unreal Engine 4 environment

MENTOR: dr.sc. Andrija Bernik | ZVANJE: Predavač

ČLANOVI POVJERENSTVA:
1. mr.sc. Dragan Matković, v. pred. - predsjednik
2. doc.art. Robert Geček - član
3. dr.sc. Andrija Bernik, pred. - mentor
4. pred. Snježana Ivančić Valenko, dipl.ing. - zamjenski član
5.

Zadatak završnog rada

BROJ: 574/MM/2018

OPIS:

Tema završnog rada je izrada 3D računalne igre korištenjem Unreal Engine 4 razvojnog okruženja.
Uvod završnog rada opisuje povijest tvrtke Epic Games koji su ujedno i tvorci razvojnog okruženja Unreal Engine 4, značajne projekte i igre te korisničko sučelje razvojnog okruženja. Govori se o prednostima i manama Unreal Engine-a 4 i radi se usporedba s konkurentima. Važno je napomenuti kako će igra biti izrađena isključivo korištenjem alata za spajanje blokova koda pomoću sustava pod nazivom "Unreal Engine 4 Blueprint Visual Scripting System".

Praktični dio opisuje izradu 3D računalne igre koristeći Unreal Engine 4 razvojno okruženje. Cilj je napraviti igru otvorenog svijeta gdje igrač počinje s minimalnom opremom te mora prikupljati različite resurse kako bi preživio što je duže moguće.

U radu je potrebno:
- opisati Unreal Engine 4 razvojno okruženje
- opisati korisničko sučelje i korištene alate
- usporediti Unreal Engine 4 s konkurentima
- opisati postupak izrade igre

REPUBLIKA HRVATSKA
SVEUČILIŠTE SJEVER

ZADATAK URUČEN: 6.9.2018. | POTPIS MENTORA: Bernik

Sveučilište
Sjever

SVEUČILIŠTE
SJEVER

IZJAVA O AUTORSTVU
I
SUGLASNOST ZA JAVNU OBJAVU

Završni/diplomski rad isključivo je autorsko djelo studenta koji je isti izradio te student odgovara za istinitost, izvornost i ispravnost teksta rada. U radu se ne smiju koristiti dijelovi tuđih radova (knjiga, članaka, doktorskih disertacija, magistarskih radova, izvora s interneta, i drugih izvora) bez navođenja izvora i autora navedenih radova. Svi dijelovi tuđih radova moraju biti pravilno navedeni i citirani. Dijelovi tuđih radova koji nisu pravilno citirani, smatraju se plagijatom, odnosno nezakonitim prisvajanjem tuđeg znanstvenog ili stručnoga rada. Sukladno navedenom studenti su dužni potpisati izjavu o autorstvu rada.

Ja, Krunoslav-Stjepan Pečutnik *(ime i prezime)* pod punom moralnom, materijalnom i kaznenom odgovornošću, izjavljujem da sam isključivi autor/ica završnog/diplomskog (*obrisati nepotrebno*) rada pod naslovom Izrada 3D računalne igre u Unreal Engine 4 razvojnom okruženju *(upisati naslov)* te da u navedenom radu nisu na nedozvoljeni način (bez pravilnog citiranja) korišteni dijelovi tuđih radova.

Student/ica:
(upisati ime i prezime)

Pečutnik
(vlastoručni potpis)

Sukladno Zakonu o znanstvenoj djelatnost i visokom obrazovanju završne/diplomske radove sveučilišta su dužna trajno objaviti na javnoj internetskoj bazi sveučilišne knjižnice u sastavu sveučilišta te kopirati u javnu internetsku bazu završnih/diplomskih radova Nacionalne i sveučilišne knjižnice. Završni radovi istovrsnih umjetničkih studija koji se realiziraju kroz umjetnička ostvarenja objavljuju se na odgovarajući način.

Ja, Krunoslav-Stjepan Pečutnik *(ime i prezime)* neopozivo izjavljujem da sam suglasan/na s javnom objavom završnog/diplomskog (*obrisati nepotrebno*) rada pod naslovom Izrada 3D računalne igre u Unreal Engine 4 razvojnom okruženju *(upisati naslov)* čiji sam autor/ica.

Student/ica:
(upisati ime i prezime)

Pečutnik
(vlastoručni potpis)

Sveučilište
Sjever

Odjel za Multimediju, oblikovanje i primjenu

Završni rad br. 574/MM/2018

Izrada 3D računalne igre u Unreal Engine 4 razvojnom okruženju

Student

Krunoslav-Stjepan Pečatnik, 0833/336

Mentor

dr.sc. ,Andrija Bernik, pred.

Varaždin, rujan 2018. godine

Predgovor

Ideja za ovu temu bio je seminarski rad/prezentacija iz kolegija “Video animacija” na temu “Unreal Engine 4 vs Unity”. Također, tema je izabrana zbog mog osobnog interesa prema 3D računalnim igrama. Od malih nogu, igra koja mi je ostala u najdražem sjećanju je Unreal Tournament koja je proizvedena korištenjem Unreal Engine razvojnog okruženja. Ta zainteresiranost prema računalnim igrama motivirala me je da iskoristim ovu priliku i kao temu završnog rada odaberem izradu 3D računalne igre korištenjem Unreal Engine 4 razvojnog okruženja.

Želio bih se zahvaliti profesoru i mentoru dr.sc. Andriji Berniku na svemu što me naučio kroz kolegije “3D modeliranje” i “Video animacija” te na pomoći i savjetima koje mi je pružio pri izradi ovog završnog rada.

Također, zahvaljujem se obitelji i svima koji su me podržavali tijekom studiranja.

Popis korištenih kratica

3D	Three-dimensional
	Trodimenzionalno, tri dimenzije
EG	Epic Games
UE4	Unreal Engine 4
BVS	Blueprint visual scripting
	Blueprint vizualno skriptiranje
HUD	Heads Up Display
	Dinamično korisničko sučelje unutar igre (traka koja pokazuje zdravlje)
UI	User Interface
	Korisničko sučelje

Sadržaj

1. UVOD 1
2. EPIC GAMES, INC 2

2.1. Unreal Tournament 1999 GOTY 3

3. UNREAL ENGINE 4 4

3.1. Blueprint vizualno skriptiranje 5
3.2. Usporedba s konkurentom (UE4 i Unity) 6
3.3. Sučelje Unreal Engine 4 razvojnog okruženja 7
3.3.1. Traka kartica 8
3.3.2. Traka izbornika 8
3.3.3. Traka s alatima 9
3.3.4. Traka s modovima 10
3.3.5. Prozor 11
3.3.6. Traka s detaljima 12
3.3.7. Traka s aktorima 13

4. RAZVOJ RAČUNALNE IGRE 14

4.1. Definiranje tipa igre i ideje 14
4.2. Modeliranje razine/nivoa 16
4.3. Teksturiranje razine/nivoa 19
4.4. Postavljanje prirodnih elemenata (bilje, kamenje…) 22
4.5. Postavljanje rasvjete 25
4.6. Postavljanje HUD-a i UI-a 26
4.7. Animiranje 31
4.8. BVS (blueprint vizualno skriptiranje) 37
4.9. Detalji i popravljanje igre 41
4.10. Pakiranje igre 43
4.11. Pokretanje igre 45

5. ZAKLJUČAK 47
6. LITERATURA 48
7. POPIS SLIKA 49
8. PRILOG 51

1. Uvod

Ovaj rad prikazuje kompletan proces izrade 3D računalne igre koristeći Unreal Engine 4 razvojno okruženje. Neke od tehnika korištenih za izradu igre su 3D modeliranje, teksturiranje, animiranje i blueprint skriptiranje. Unreal Engine 4 nam omogućuje korištenje svih navedenih tehnika kako bi ih mogli spojiti u jednu cjelinu što na kraju rezultira računalnom igrom. Također, uz Unreal Engine 4, Epic Games nam nudi bezbroj besplatnih proizvoda koje možemo koristiti pri izgradnji našeg projekta. Neki od korištenih su "Open World Demo Collection" koji nam dopušta korištenje modela iz prirode kao što su kamenje, drveće i trava, te "Water Planes" koji nam omogućuje dodavanje fotorealistične plohe koja predstavlja vodenu površinu.

Epic Games nam besplatno nudi svoj Unreal Engine 4 koji je sam i više nego dovoljan da bi se napravila računalna igra. Naravno, neke teksture koje su bile korištene u izradi ovog rada UE4 nije sadržavao jer je tema bila specifična. Zbog toga te iste teksture preuzete su s interneta i preuređene po potrebi u programu Adobe Photoshop.

2. Epic Games, Inc

Epic Games je tvrtka koja se bavi proizvodnjom video igara sa sjedištem u Sjedinjenim Američkim Državama. Tvrtku je osnovao Tim Sweeney 1991. godine pod imenom Potomac Computer Systems. Godinu dana kasnije, tvrtka mijenja ime u Epic MegaGames te dovodi Marka Rein-a koji i danas ima ulogu potpredsjetnika kompanije. [2] [7]

Slika 2.1: Epic Games Logo

Epic Games je proizveo i koristi Unreal Engine koji je i ujedno glavno razvojno okruženje njihovih najpopularnijih igara kao što su Fortnite, Unreal Tournament i Gears of War.

2.1. Unreal Tournament 1999 GOTY

Unreal Tournament je pucačka igra iz prvog lica koja je jedna od najpoznatijih igara proizvedenih od strane Epic Games-a. U vrijeme izlaska igre, Unreal Tournament je dobivao razne pohvale glede grafike, dizajna i same igrivosti. Samim time, Epic Games je polako počeo sakupljati reputaciju kao jedan od ozbiljnih izdavača videoigara. Igra također pokazuje prve znakove odličnog funkcioniranja umjetne inteligencije tj. NPC-a (likova kojima ne upravlja igrač, eng. *non person character*). Cilj igre je penjanje po ljestvici turnira kako bi izazvali prvaka, Xan-a, tajanstvenog bića s iznimnom vještinom.

Također, igra je poznata diljem svijeta po lako prepoznatljivoj mapi "Facing Worlds".

Igra Unreal Tournament 99 bila je jedan od glavnih razloga odabira UE4 razvojnog okruženja za izradu računalne igre čiji je postupak prikazan kasnije u ovom radu.

Slika 2.2.: Unreal Tournament 1999 Game of the Year Edition

3. Unreal Engine 4

Iako je Unreal Engine 4 napravljen u svrhu razvijanja pucačkih igara iz prvog lica, nudi nam kompletan paket alata pomoću kojih smo u mogućnosti napraviti bilo kakvu računalnu igru, fotorealističnu scenu ili visokokvalitetni filmski sadržaj. Koristi C++ programski jezik no daje nam mogućnost korištenja BVS (blueprint vizualnog skriptiranja), što znači da smo u mogućnosti napraviti računalnu igru bez pisanja ijedne linije koda. [1][8]

Konferencija za proizvođače igara održana 19. ožujka 2014. godine značajna je po tome što je Epic Games izdao svoj Unreal Engine 4 kroz novi model licenciranja. Uzimanjem mjesečne pretplate, proizvođači igara dobili su u potpuno funkcionalnu verziju razvojnog okruženja. Izdavanje projekta proizvedenog u UE4 rezultiralo je naplatom od svega 5% bruto prihoda.

Epic Games je 4. rujna 2014. godine besplatno oslobodio svoj Unreal Engine 4 na korištenje školama i sveučilištima diljem svijeta, omogućivši korištenje svog razvojnog okruženja u potpunosti bez mjesečne naknade.

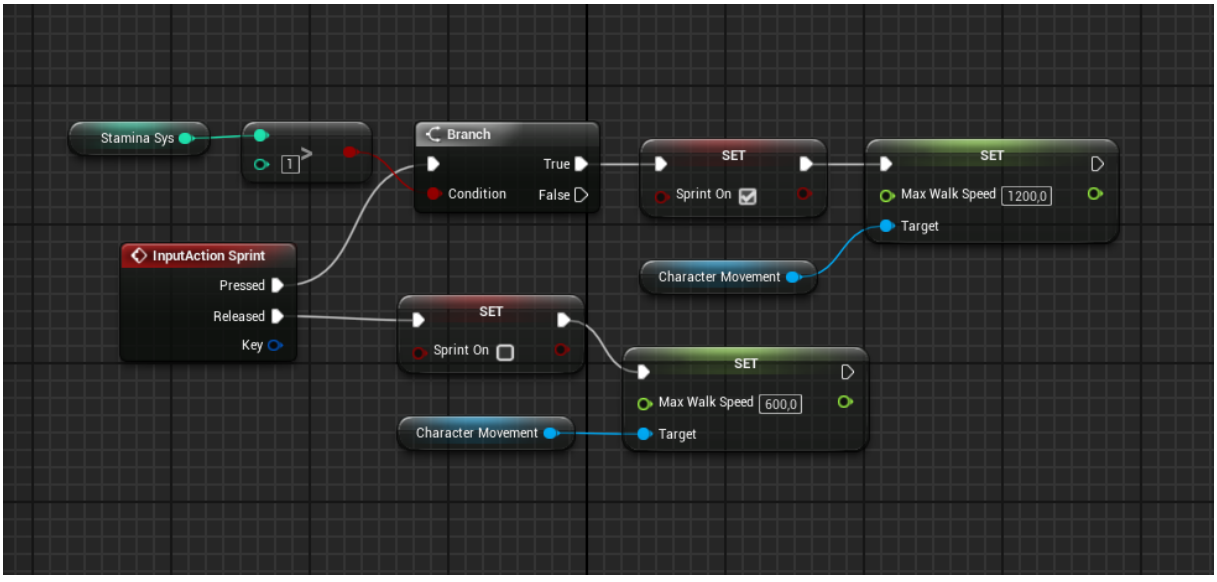
Tijekom konferencije proizvođača igara koja je održana u ožujku 2015. godine, Epic Games je objavio da će osloboditi Unreal Engine 4, sa svim budućim ažuriranjima svim korisnicima, potpuno besplatno. U zamjenu, Epic Games je postavio naknadu od 5% prihoda na proizvedeni sadržaji ukoliko je proizveo više od 3000 dolara prihoda. Na toj istoj konferenciji, Epic Games je najavio uspostavljanje Unreal Marketplace-a preko kojeg korisnici mogu kupiti i prodati sadržaj proizveden u Unreal Engine 4 pogonskom sklopu.

3.1. Blueprint vizualno skriptiranje

Blueprint vizualno skriptiranje je sustav ugrađen u Unreal Engine 4 koji nam omogućuje kompletan sustav skriptiranja igara. Temelji se na konceptu korištenja sučelja baziranog na čvoru (eng. *node-based interface*) kako bi se stvorili elementi igrivosti unutar UE4. Korištenjem ovog sistema nailazimo da se objekti definirani pomoću blueprint vizualnog skriptiranja nazivaju samo blueprinti (eng. *blueprints*).

Ovakav sustav proizveden je s namjenom kako bi i dizajneri koji ne posjeduju izvanredno znanje programskih jezika bili u mogućnosti samostalno proizvesti računalnu igru. Sustav je izuzetno fleksibilan jer omogućuje dizajnerima jednostavno korištenje koncepata i alata kojima svakodnevno barataju programerima.

Blueprinti rade po principu povezivanja čvorova, događaja i funkcija koji su specifični za svaku radnju u računalnoj igri. [1]



Slika 3.1: Primjer "Sprint" blueprinta igre

3.2. Usporedba s konkurentom (UE4 i Unity)

Unreal Engine 4 i Unity su danas dva najpopularnija razvojna okruženja za izradu igara. Ova usporedba služi kako bismo saznali koji pogonski slop je idealan za nas tj. za razvoj naše igre. Važno je napomenuti kako jedan i drugi imaju svoje prednosti i mane te odabir više ovisi o tipu igre koju želimo napraviti, ne o optimizaciji, težini rada u programu i otsalim sličnim parametrima.

Ako spomenemo igre za android uređaje, Unity je prvi koji dolazi u obzir. Također, ako želimo napraviti 2D ili indie igru, Unity je izvrstan odabir jer nam nudi korištenje odličnih 2D karakteristika i može uvelike pojednostaviti proces izrade 2D igre. Za izradu 3D igara Unity je također solidan izbor no grafički gledano nikako nije na istoj razini kao UE4. Jedna od glavnih prednosti Unity-a je jednostavan razvoj igara na skoro svim platformama (Windows, Mac OS, Linux…). Ako gledamo programski dio kod procesa proizvodnje igre, smatra se da je UE4 jednostavniji za savladati jer nudi BVS (blueprint vizualno skriptiranje) koji ustvari nudi mogućnost izrade računalne igre bez klasičnog pisanja koda. Naravno, ovaj sustav nije savršen, puno "teže" djeluje na učinak rada i brzinu računala nego klasično pisanje koda. Što se tiče dokumentacije, Unity je odličan za korisnike koji su tek počeli s radom u industriji računalnih igara. Unity također ima svoj asset store koji nam daje veći izbor proizvoda nego UE4 marketplace. [3]

Kao što je već spomenuto, oba razvojna okruženja su odlična i trenutno dominiraju na tržištu. Ako dođemo u situaciju da moramo izabrati jedan od navedenih pogonskih sklopova bitno je imati jasnu viziju kakvu igru želimo napraviti. Ako se odlučimo za indie tip igre ili pak 2D igru, Unity je bolje rješenje. Ako se odlučimo za vizualno moćnu, veću i na neki način fotorealističnu 3D igru, tada ni razvojno okruženje na tržištu nije na razini na kojoj se trenutno nalazi Unreal Engine 4.

3.3. Sučelje Unreal Engine 4 razvojnog okruženja

Unreal Engine 4 sastoji se od par ključnih elemenata sučelja koje koristimo pri izradi projekata. Glavno sučelje UE4 također možemo nazvati i urednik razine (engl. *level editor*).

Urednik razine pruža funkcije stvaranja razina/nivoa u UE4. Ovdje se stvaraju razine, pregledavaju i mijenjaju postavljanjem, preoblikovanjem i uređivanjem svojstava aktora. U UE4 razvojnom okruženju, scene koje stvaramo se općenito nazivaju razine. Razina je 3D okruženje u kojem postavljamo niz objekata i geometrije kako bi definirali svijet koji će igrači doživjeti igranjem same igre. Svaki objekt koji se nalazi u svijetu, bilo svjetlo, drvo ili glavni lik, naziva se aktorom. Tehnički gledano, aktor je programska klasa koja se koristi u UE4 za definiranje objekata koji imaju 3D položaj, rotaciju i podatke o skali. [6]

Slika 3.2: Level Editor

Urednik razine (eng. *Level Editor*) se sastoji od sljedećih elemenata:

1. Traka s karticama (eng. *Tab Bar*)
2. Traka s izbornicima (eng. *Menu Bar*)
3. Traka s alatima (eng. *Tool Bar*)
4. Traka s modovima (eng. *Modes*)
5. Prozor (eng. *Viewport*)
6. Traka s detaljima (eng. *Details*)
7. Traka s aktorima (eng. *World Outliner*)

3.3.1. Traka kartica

Urednik razine sadrži karticu koji daje naziv razine na kojem trenutno radimo. Kartice s drugih prozora za uređivanje mogu se priključiti uz ovu karticu za brzu i jednostavnu navigaciju. Također, s desne strane možemo vidjeti naziv projekta. [6]

Slika 3.3: Traka kartica

3.3.2. Traka izbornika

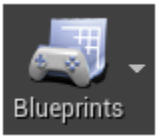
Traka izbornika poznata je svima koji su koristili Windows aplikacije. Omogućuje pristup općim alatima i naredbama koje se koriste pri radu s razinama u uredniku razine. [6]

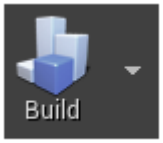
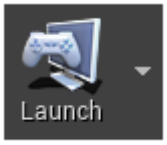
Slika 3.4: Traka izbornika

3.3.3. Traka s alatima

Traka s alatima, kao u većini aplikacija, skupina je naredbi koja omogućuje brzi pristup uobičajenim alatima i operacijama. [6]

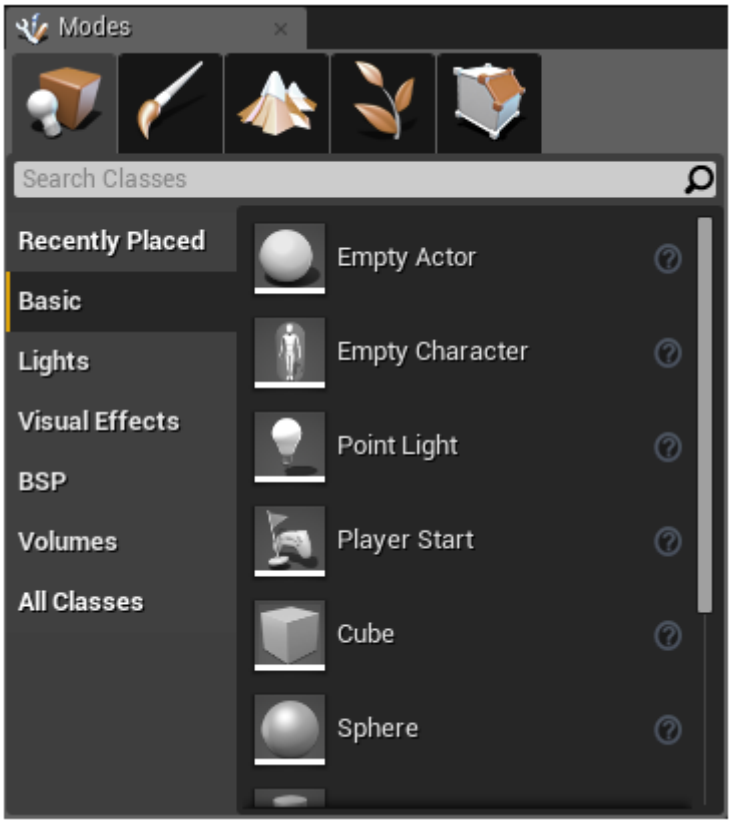
Slika 3.5: Traka s alatima

Naredba	Opis naredbe
	Sprema trenutnu razinu.
	Omogućuje povezivanje ili dodjeljivanje rješenja za upravljanje izvorom.
	Otvara preglednik sadržaja koji prikazuje sav sadržaj naše igre. Ovdje se uređuje sav sadržaj igre.
	Otvara Unreal Engine Launcher u odjeljak Marketplace.
	Postavke koje omogućuju jednostavan pristup najčešće korištenim opcijama kojima se upravlja u Level Editoru.
	**Omogućuje pristup stvaranju ili uređivanju blueprinta za trenutnu razinu.**
	Omogućuje stvaranje novog slijeda događaja. "Npr. dolazak na određeni dio mape aktivira požar."

	**Izvršava radnju gradnje na svim razinama u editor. Tijekom ovog procesa izračunavaju se statična rasvjeta i geometrija.**
	Pokreče igru u načinu normalnog igranja.
	**Pokreče trenutnu razinu na bilo kojoj od podržanih platform.**

3.3.4. Traka s modovima

Predstavlja različite načine uređivanja razine u Level Editoru. [6]



Slika 3.6: Traka s modovima

Alat	Funkcija alata
	Aktivira način rada za postavljanje i uređivanje aktora u sceni.
	Aktivira način rada za bojanje površina.
	Aktivira način rada za uređivanje i modeliranje krajolika.
	Aktivira način rada za postavljanje flore i faune.
	Aktivira način rada za uređivanje geometrije.

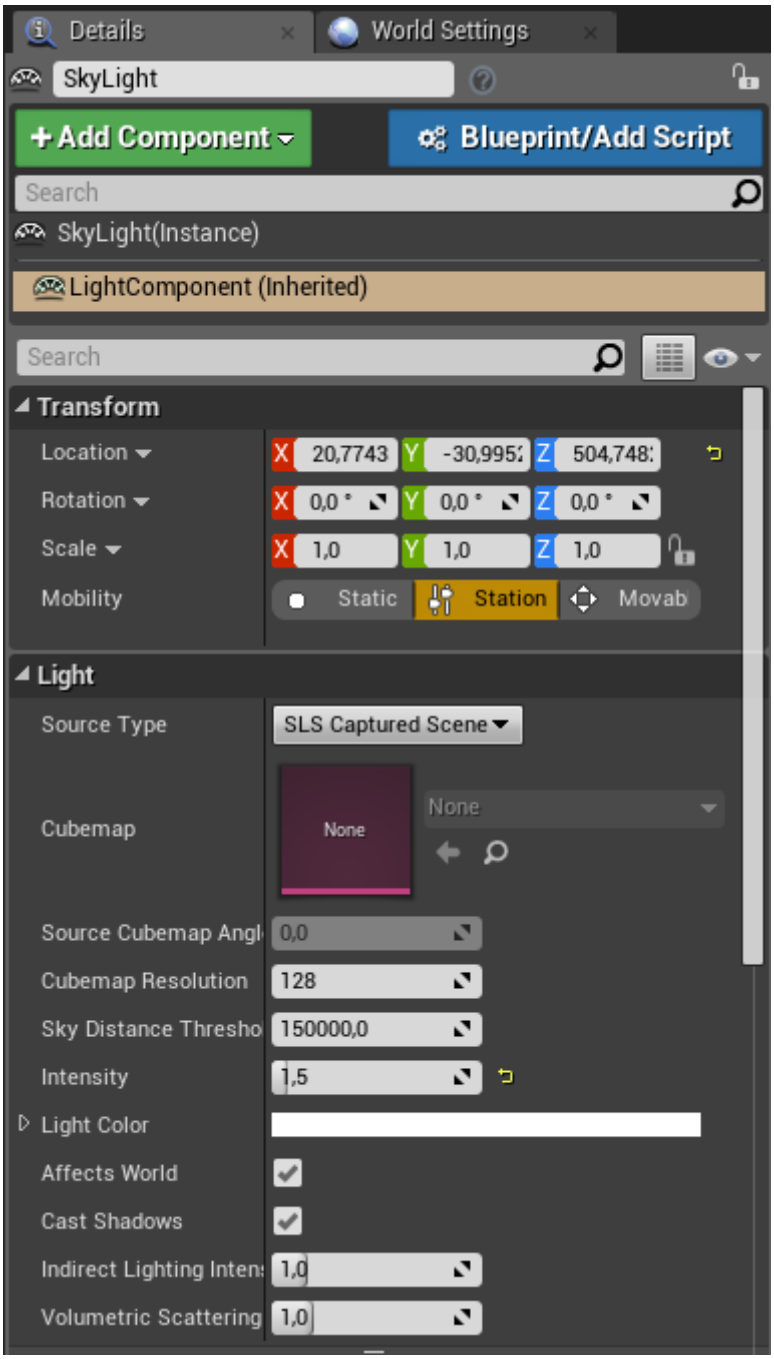
3.3.5. Prozor

Prozor (engl. *Viewport*) nam omogućuje pogled u svijet/nivo/razinu koju stvaramo u Unreal Editoru. [6]

Slika 3.7: Prozor

3.3.6. Traka s detaljima

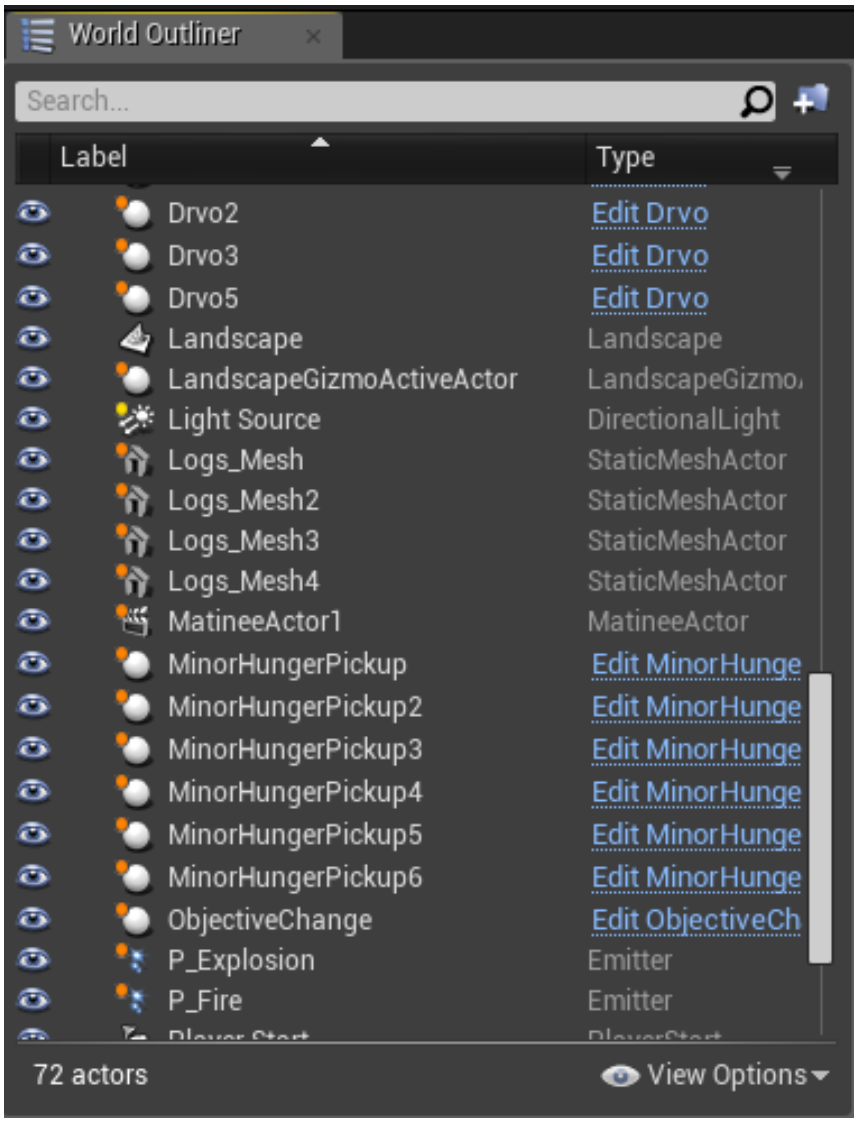
Traka s detaljima sadrži informacije koje su specifične za točno odabranog aktora. Daje mogućnost za uređivanje, kretanje, okretanje i skaliranje aktora, prikazuje entitete za odabrane aktore i omogućuje brz pristup dodatnim funkcijama uređivanja, ovisno o vrsti odabranog aktora. [6]



Slika 3.8: Traka s detaljima

3.3.7. Traka s aktorima

Traka s aktorima sadrži podatke o svim aktorima unutar scene u hijerarhijskom prikazu tablice. Aktori se mogu odabrati i mijenjati izravno iz ove trake. Također možemo koristiti padajući izbornik “Info” kako bismo uključili dodatni stupac koji prikazuje razine, slojeve ili ID imena.



Slika 3.9: Traka s aktorima

4. Razvoj računalne igre

Razvoj računalne igre podijeljen je u nekoliko faza:

- definiranje tipa igre i ideje
- modeliranje razine,
- teksturiranje razine,
- postavljanje prirodnih elemenata,
- postavljanje rasvjete,
- postavljanje HUD-a (Heads Up Display) i UI-a (User Interface)
- animiranje
- BVS (blueprint vizualno skriptiranje)
- detalji i popravljanje igre
- pakiranje igre

4.1. Definiranje tipa igre i ideje

Kako bismo uopće i započeli razvoj neke igre potrebno je imati jasnu ideju. Ideja ove računalne igre došla je prilikom pisanja seminara iz kolegija “Video animacija” na temu “Unreal Engine 4 vs Unity”. Prilikom pisanja spomenutog seminara zaključeno je kako je završni rad idealna prilika za razvijanje 3D računalne igre i učenje novog programa.

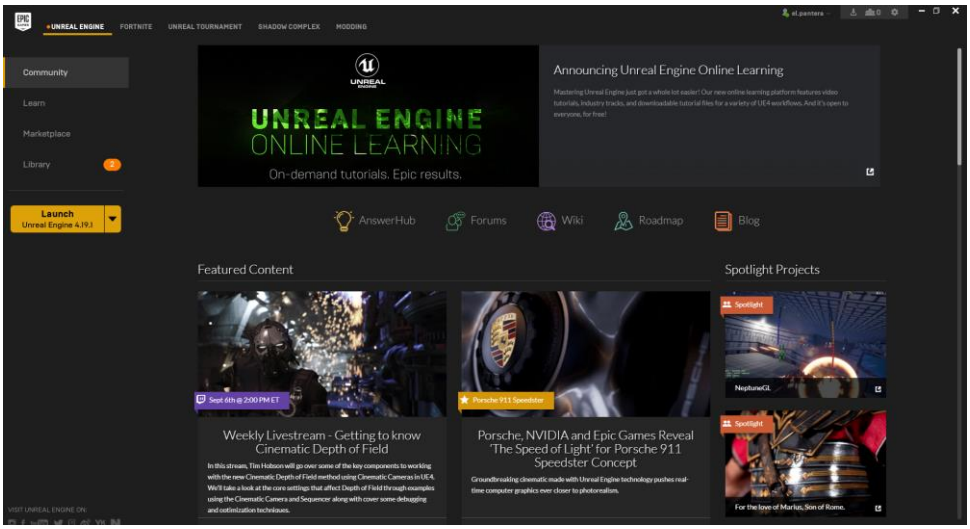
Slika 4.1: Ideja - igra "The Forest"

Ovaj primjer prikazuje igru “The Forest” koja je bila jedna od glavnih inspiracija prilikom odluke kakav tip igre napraviti. Igra napravljena i prikazana kroz ovaj rad je preživljavački “sandbox” što znači da igrač ima mogućnost nelinearnog kretanja po mapi sa zadatkom da preživi. Cilj je napraviti igru s mračnom atmosferom preživljavanja u šumi tijekom noći. Igrač će morati skupljati hranu kako bi preživio i ne smije dozvoliti da mu traka zdravlja padne na nulu, što rezultira umiranjem. Također, igrač koristi ručnu svjetiljku kako bi imao bolju vidljivost u mračnoj šumi. Ne smije dopustiti da se svjetiljka potroši zbog lose vidljivosti u noći i tako mora skupljati kutije u kojima se nalaze baterije kako bi napunio ručnu svjetiljku. Glavni zadatak u igri jest skupiti četiri komada drveta i zapaliti vatru na malom otoku usred jezera kako bi pozvao u pomoć.

Slika 4.2: Konačni izgled igre

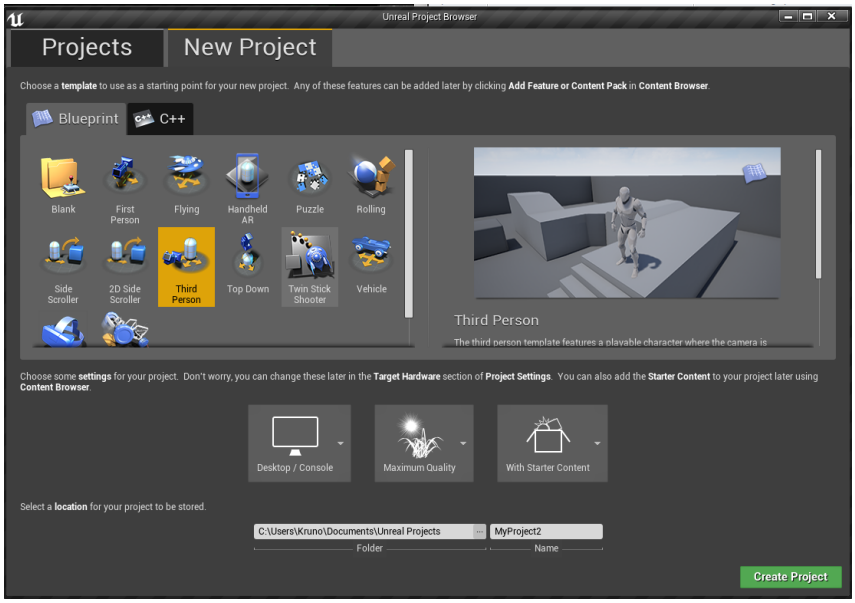
4.2. Modeliranje razine/nivoa

Da bi koristili Unreal Engine 4 potrebno je skinuti Epic Games Launcher koji se besplatno može preuzeti na službenoj stranici Epic Games-a. Nakon instalacije Epic Games Launcher-a odlazi se na sekciju Unreal Engine i otvara se UE4.



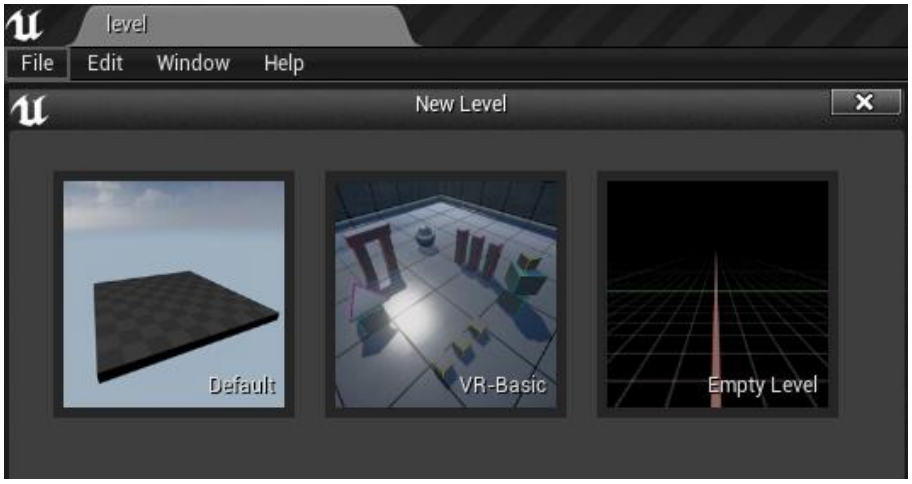
Slika 4.3: Epic Games Launcher

Pritiskom na tipku "*Launch*" UE4 nudi nekoliko opcija. U ovom primjeru, iako je cilj napraviti igru iz prvog lica, odabiremo "*3rd person template*" jer UE4 automatski stavlja pušku u ruke pri ulazku u "*1st person template*" . Kasnije ćemo kameru ubaciti u glavu našeg lika i isključiti vidljivost tijela.



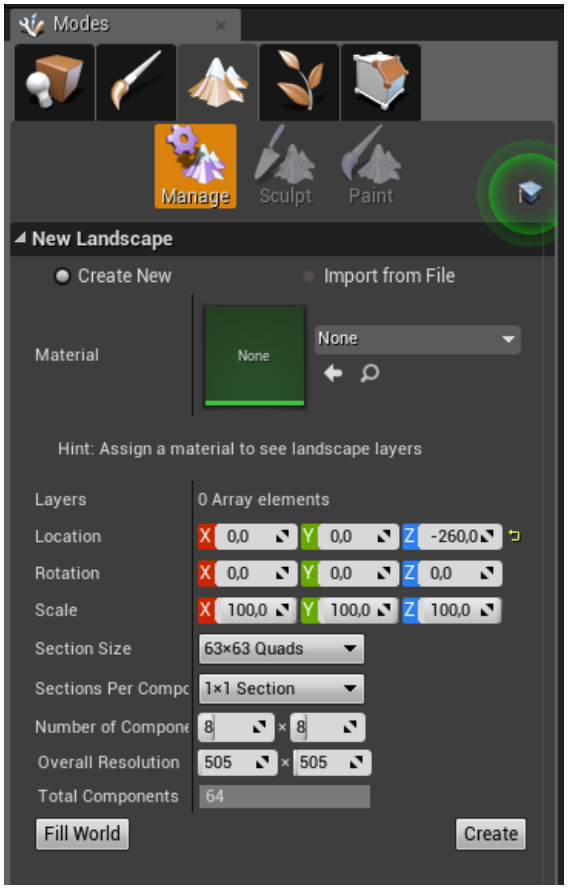
Slika 4.4: Odabir predloška

Po ulasku u program, potrebno je napraviti novu razinu. To se radi klikom na *File-New Level* i odabire se "*Default Level*".



Slika 4.5: Stvaranje nove razine

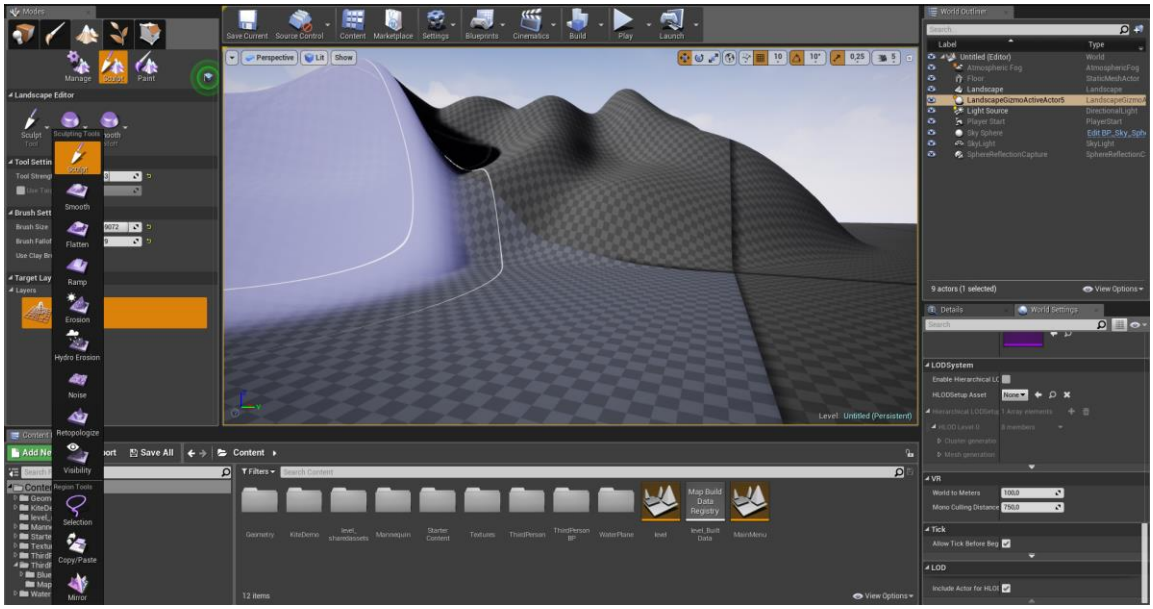
Kada se stvori nova razina, skalira se po potrebama koristeći "*Manage*" alat u traci modova i ažurira se pritiskom na tipku "*Create*".



Slika 4.6: Skaliranje nove razine

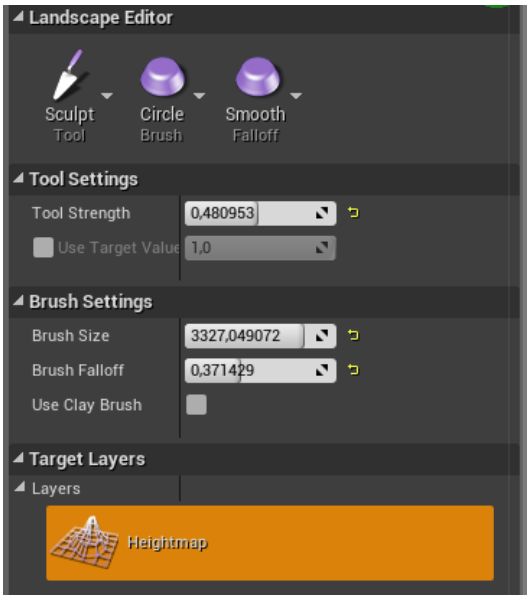
Modeliranje omogućava stvaranje trodimenzionalnih objekata na koje kasnije želimo primijeniti teksturu, animaciju ili simulaciju. [4][5]

Koristeći "*Sculpt*" alate u traci modova započinje se s modeliranjem razine. Svaki alat ima svoju funkciju. Na primjer, "*sculpt*" podiže krajolik, "*flatten*" ravna krajolik na mjestu na kojem smo počeli koristiti alat, "*erosion*" udubljuje krajolik, i tako dalje.



Slika 4.7: Modeliranje razine

Naravno za svaki alat moguće je podesiti parameter kao što su veličina kista i snaga djelovanja kista.

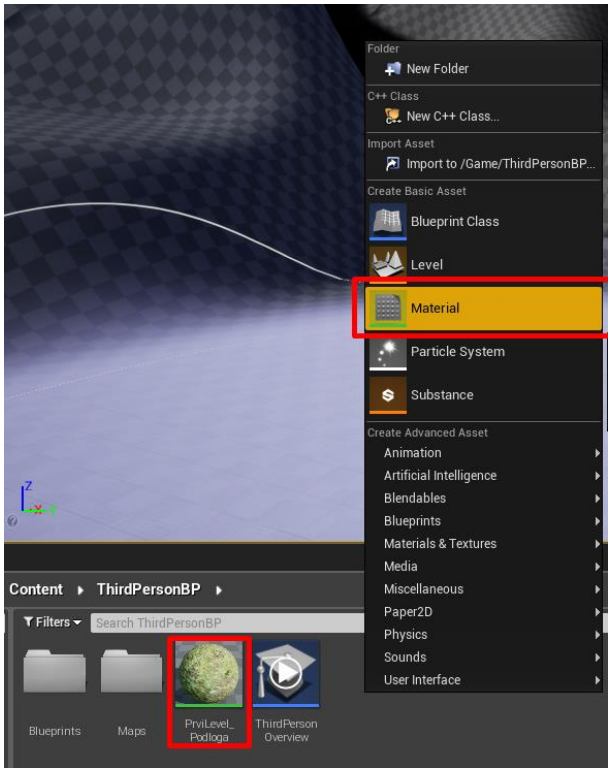


Slika 4.8: Parametri "sculpt" alata

4.3. Teksturiranje razine/nivoa

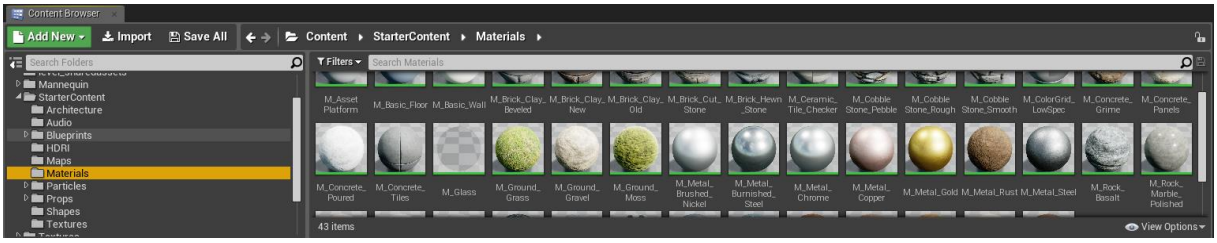
Nakon što razina poprimi željeni oblik, vrijeme je da se dodaju teksture. Svaki 3D model, u ovom slučaju reljef, može se i teksturirati, odnosno može se mu dodijeliti boja, materijal, svojstvo. Na samom početku teksturiranja odabire se tip materijala koji se primjenjuje na razini. [4][5] Sve teksture se već nalaze u Unreal Engine 4 razvojnom okruženju. Zadatak je da stvoriti tri tipa materijala: zemlja, trava, kamen. Naravno, kako tekstura ne bi izgledala kao list papira zalijepljen na plohu, potrebno je podesiti neke parametre.

U traci "Content Browser" pravi se novi materija (desni klik-new material).



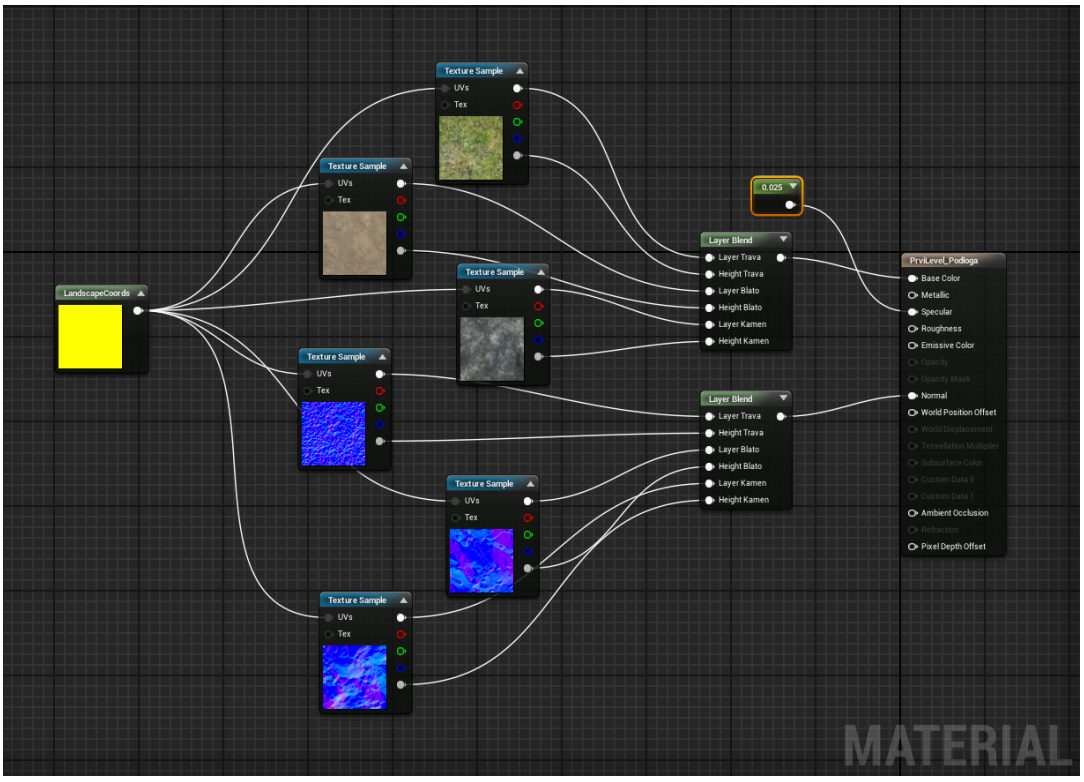
Slika 4.9: Stvaranje materijala

U traci "*Content Browser*" pronalaze se teksture za travu, blato i kamenje. Duplim klikom otvara se novi prozor materijala i tamo je moguće tehnikom povlačenja (engl *drag and drop*) ubaciti odabrane teksture.



Slika 4.10: Unreal Engine 4 teksture

Također, Unreal Engine 4 nudi svoje normale koje služe da tekstura ne bi izgledala kao papir nalijepljen na podlogu. Pronalaze se normale za svaku teksturu i vuku se u prozor gdje se nalaze postavke novog materijala. Ovaj proces podešavanja parametra materijala sličan je blueprint vizualnom skriptiranju. Potrebno je točno spojiti teksture i s lijeve strane u traci s detaljima podesiti željene parametre.



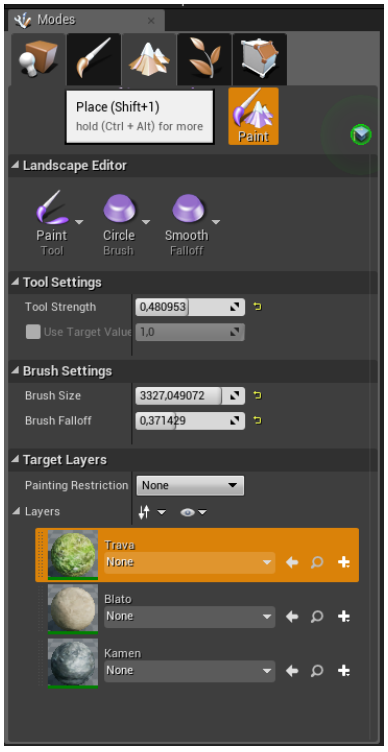
Slika 4.11: Materijal (blato, trava, kamen)

U ovom slučaju postavlja se i "*Specular*" čvor i podešava se na 0.025 koji sve tri teksture skalira na jednake proporcije.

Nakon podešavanja tekstura vrijeme je za teksturiranje terena. Prvo je potrebno označiti "*landscape*" u traci "*World Outliner*", zatim postaviti stvoreni materijal kao materijal koji se koristi pri teksturiranju terena. To se podešava u traci s detaljima. Potrebno je prvo označiti stvoreni materijal u traci s podacima i nakon toga ubaciti ga pod parametar "*Landscape Material*".

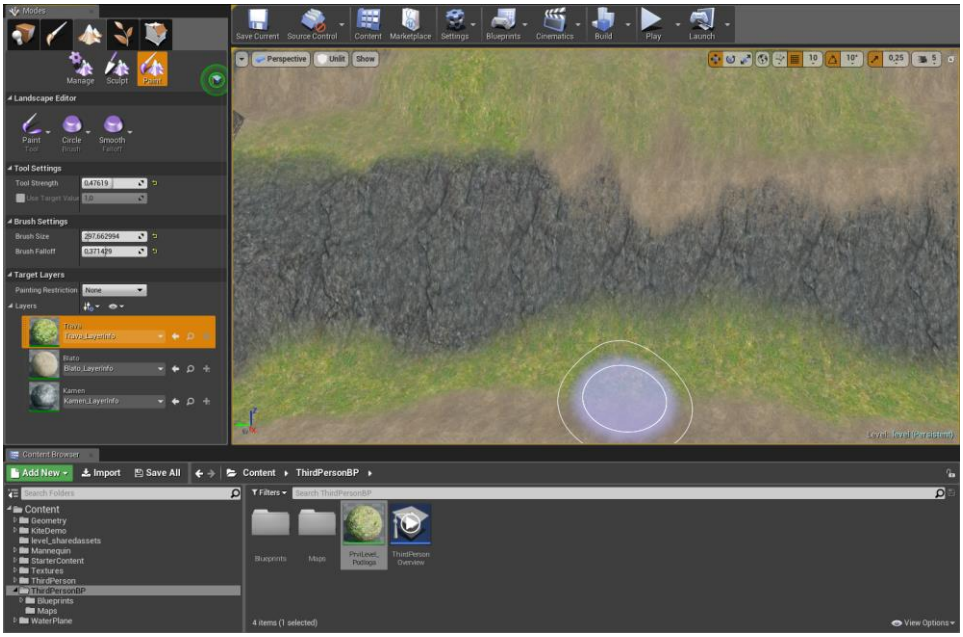
Slika 4.12: Postavljanje materijala

U traci s modovima pod trakom za bojanje pojavljuju se tri stvorena materijala (blato, trava, kamen). Koriste se na sličan način kao što i alati za stvaranje razine. Također, mogu se podešavati parametri kao što su veličina kista i snaga samog alata.



Slika 4.13: Alati za bojanje

Korištenjem tih tekstura postavlja se temeljna tekstura na koju tek kasnije dolazi bilje, kamenje i drveće.



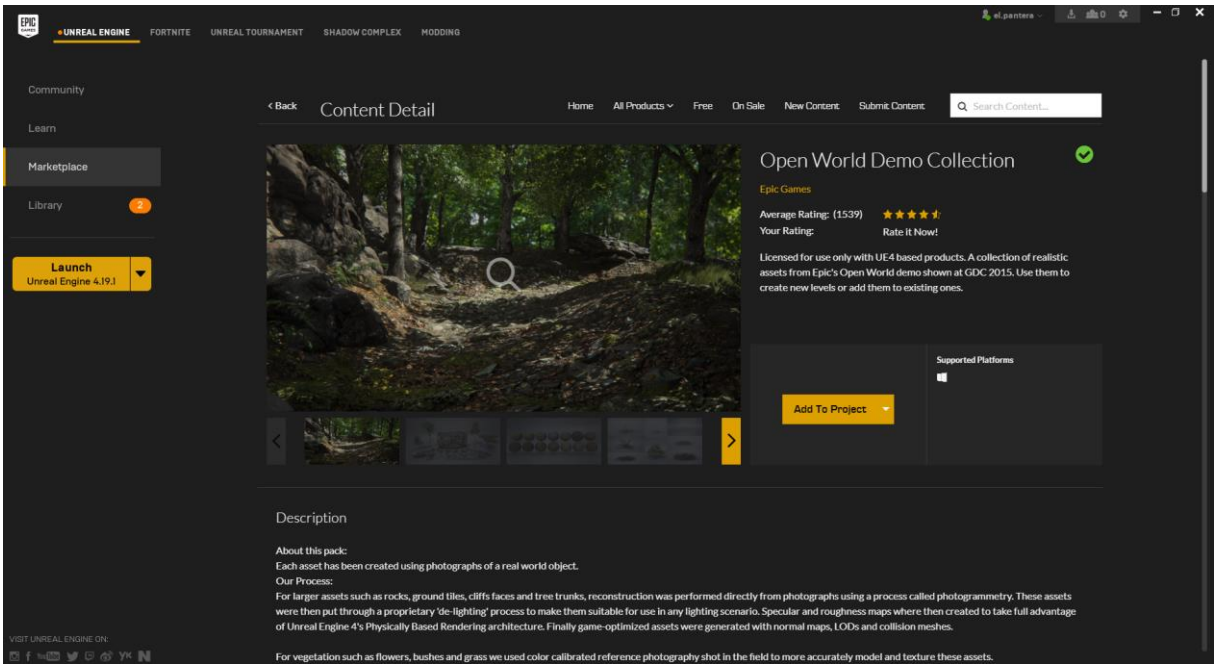
Slika 4.14: Teksturiranje razine

4.4. Postavljanje prirodnih elemenata (bilje, kamenje…)

Unreal Engine Marketplace nudi puno besplatnih elemenata koje možemo koristiti pri izradi projekta. Za ovu računalnu igru korištena je besplatna kolekcija bilja, kamenja i ostalih prirodnih modela i elemenata. Tu kolekciju moguće je besplatno preuzeti i uvesti u projekt preko Epic Games Launchera pod izbornikom "*Marketplace*".

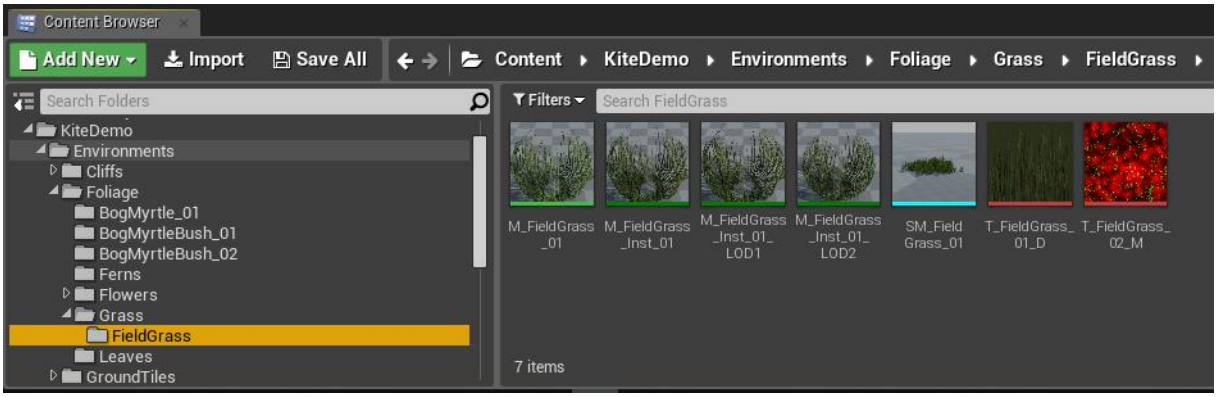
Slika 4.15: UE4 Marketplace

Klikom na „*Add to Project*“ dodaje se „Open World Demo Collection“ u projekt.



Slika 4.16: Dodavanje kolekcije prirodnih modela i elemenata

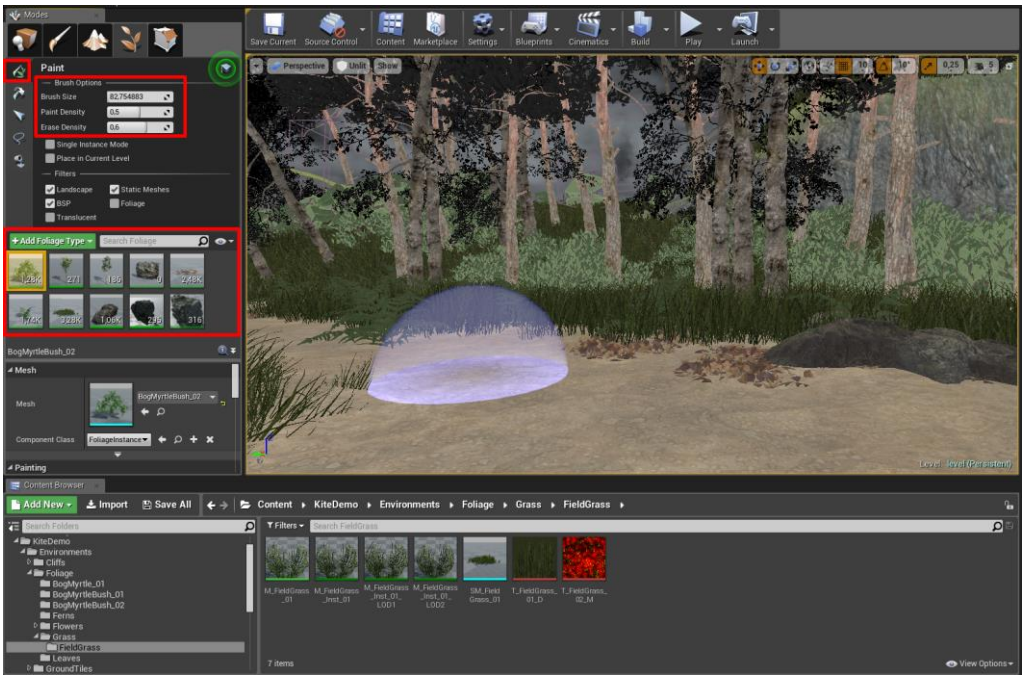
Kolekcija se sada nalazi u projektu i može ju se pronači u traci “*Content Browser*” pod imenom “KiteDemo”.



Slika 4.17: KiteDemo

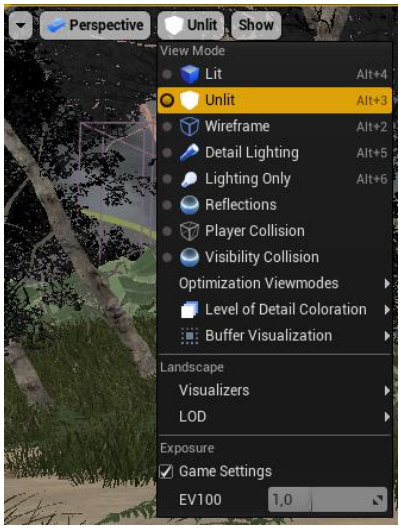
Kako bi se ovi modeli koristili, također koristi se traka s modovima no pod prozorom “*Foliage*”. U mapi “KiteDemo” pronalaze se željeni modeli i povlače se (eng. *drag and drop*) u “Foliage Editor”.

Također, moguće je i podešavati parametre modela. Neki od njih su veličina modela, gustoća modela, veličina kista i snaga kista. Ovaj princip funkcionira isto kao i teksturiranje razine. Nakon što se podese parametri modela, uzima se kist i pomoću njega se modeli postavljaju po podlozi.



Slika 4.18: Postavljanje trave, kanjema, drveća

Kako bi rad u UE4 razvojnom okruženju bio što lakši, kao pogled se preporučuje korištenje "Unlit" načina rada da se stabiliziraju parametri svijetla ako je razina presvijetla ili pretamna.



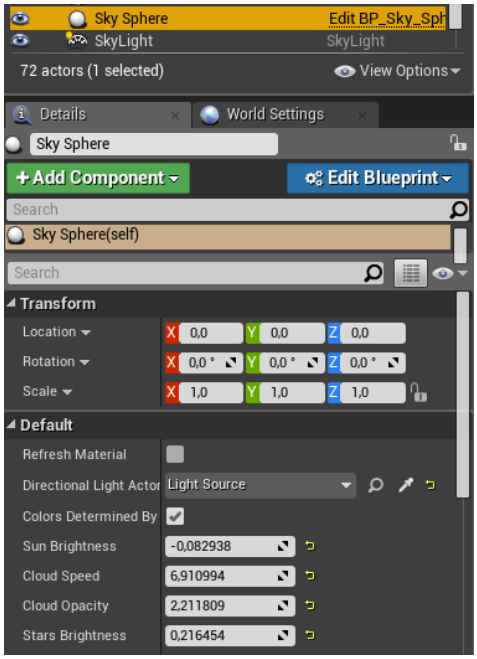
Slika 4.19: Unlit način rada

4.5. Postavljanje rasvjete

Pošto je Unreal Engine 4 već postavio rasvjetu u projekt, jedino što je preostalo je samo podesiti parametre rasvjete. U izborniku "*World Outliner*" potrebno je pronaći "*Skylight*" i rotirati ga. Ovo funkcionira tako da rotiramo sunce ispod vidljive razine kako bi se iznad razine postigla noć umjesto dana.

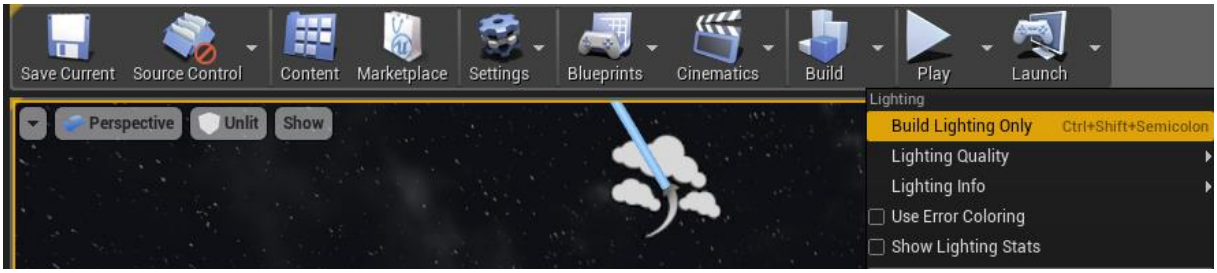
Slika 4.20: Rotiranje rasvjete

Nakon rotiranja rasvjete potrebno je podesiti parametre aktora "*SkySphere*" po želji. Moguće je postaviti gustoću oblaka, intenzitet vidljivosti zvijezda, brzinu kretanja oblaka po nebu, prozirnost oblaka itd.



Slika 4.21: Parametri "SkySphere"

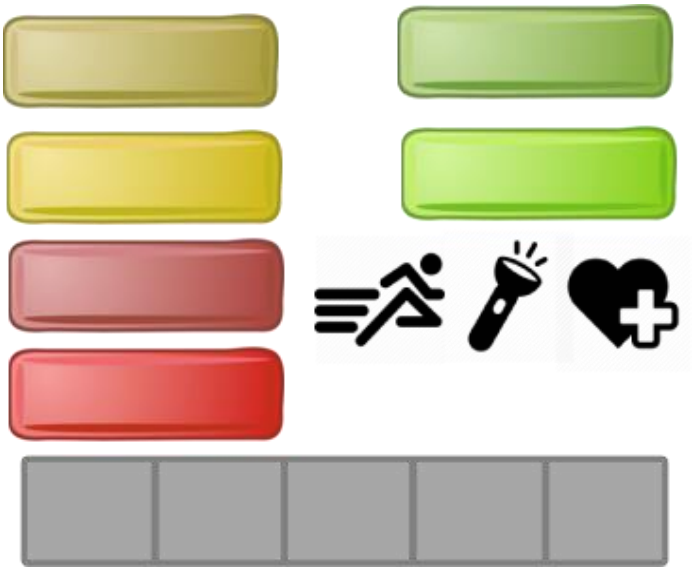
Nakon postavljanja rasvjete potrebno je izgraditi rasvjetu kako bi se sve promjene pravilno ažurirale. U traci s alatima, pritiskom na padajući izbornik pokraj alata "*Build*" odabire se naredba "*Build Lightning Only*".



Slika 4.22: Ažuriranje rasvjete

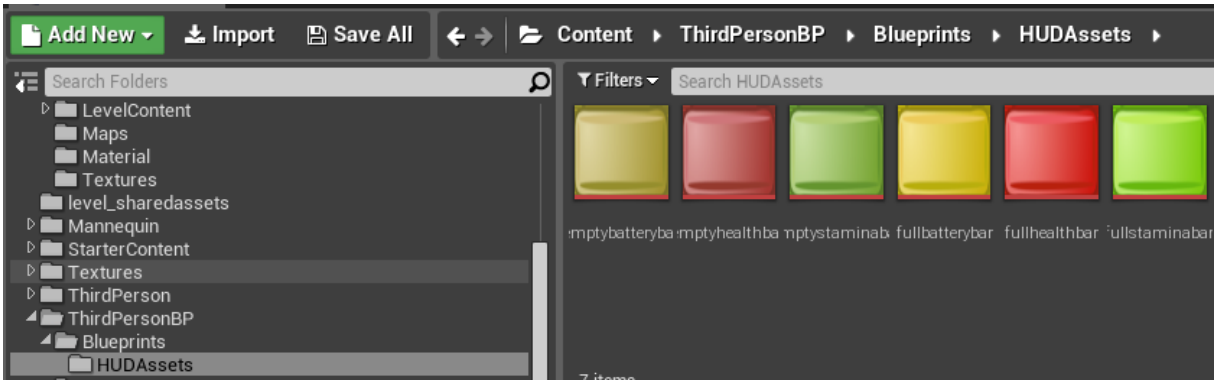
4.6. Postavljanje HUD-a i UI-a

Svaka igra mora imati svoje korisničko sučelje. Potrebno je napraviti traku sa zdravljem, kondicijom, ručnom svjetiljkom i ruksak. Materijale prikupljamo na internetu i u photoshopu radimo dvije verzije svake slike. Jednu tamniju i jednu svjetiljku. Kad će se traka prazniti postajat će sve svjetlija, a obrnuto kad će se puniti.



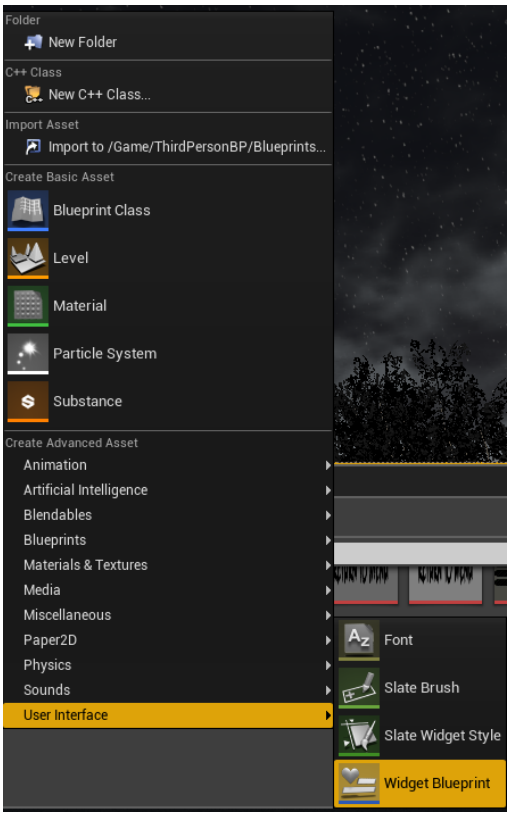
Slika 4.23: Elementi HUD-a

U traci "*Content Browser*" kreira se nova mapa i u nju se uvoze (pritiskom na tipku import) sljedeći elementi u PNG obliku.



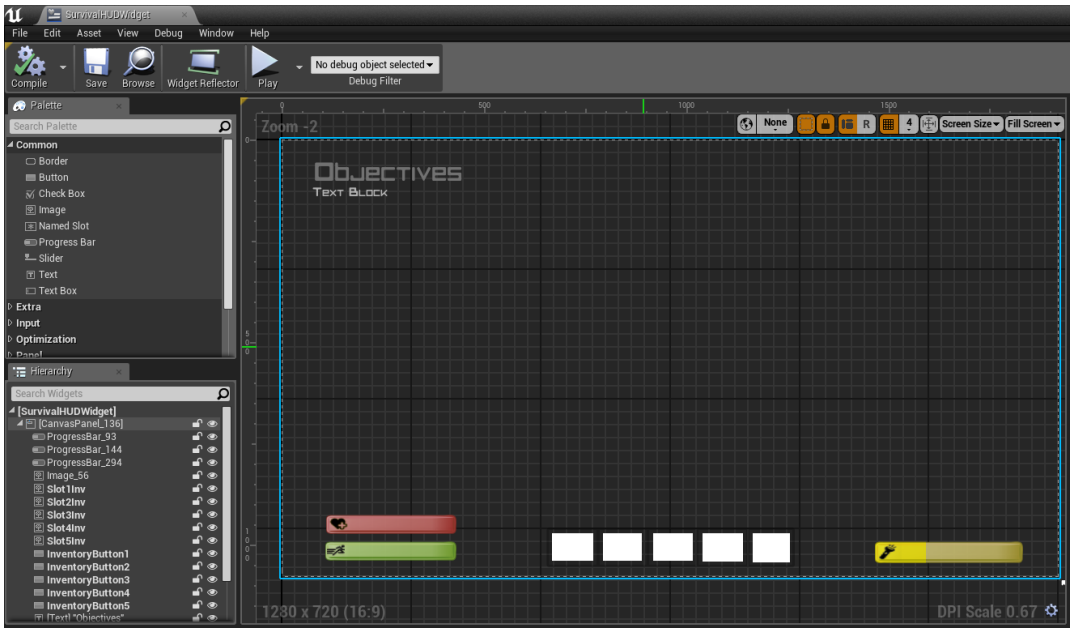
Slika 4.24: Uvoz elemenata HUD-a

Potrebno je napraviti novi Widget koji služi za postavljanje elemenata korisničkog sučelja. Kod stvaranja novih Widgeta i Blueprinta preporučuje se korištenje mape "Blueprints" kako bi projekt bio čim pregledniji. Novi Widget se stvara desnim klikom bilo gdje u "Content Browseru" – User Interface – Widget Blueprint.



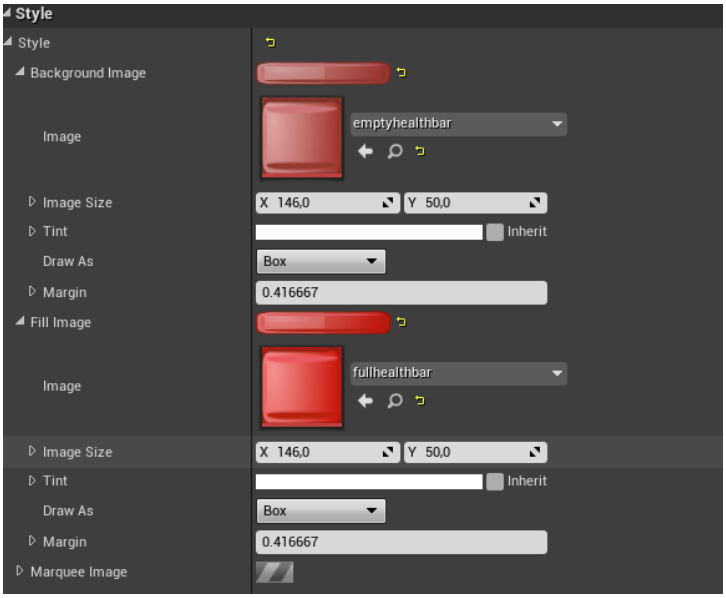
Slika 4.25: Stvaranje HUD Widgeta

Potrebno je otvoriti novi Widget i postaviti elemente korisničkog sučelja po želji.



Slika 4.26: Postavljanje HUD-a

U paleti s lijeve strane moguće je postaviti i tekst kao što je u primjeru "*Objectives*". Kasnije će taj tekst postati interaktivan uz pomoć blueprint vizualnog skriptiranja. Što se tiče traka zdravlja, kondicije i ručne svjetiljke potrebno je u parametrima postaviti i svijetlu i tamnu sliku kako bi se traka mijenjala ovisno o tome smanjuje li se zdravlje igrača, kondicija ili kapacitet baterije.

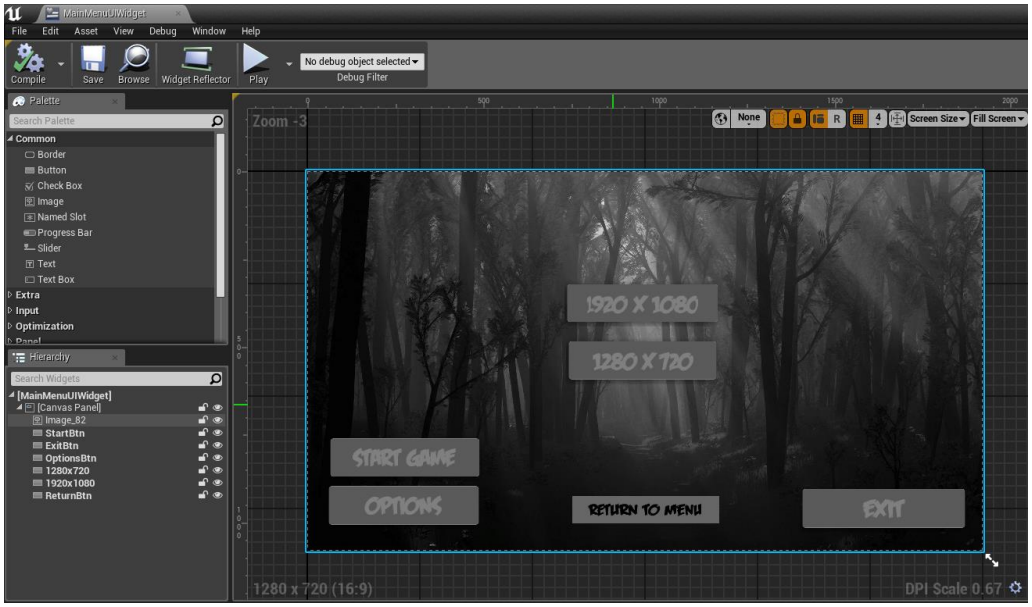


Slika 4.27: Parametri traka

Kod stvaranja korisničkog sučelja glavnog menija postupak je isti. Prvo što je potrebno je uvesti elemente u Unreal Engine 4.

Slika 4.28: Elementi korisničkog sučelja glavnog menija

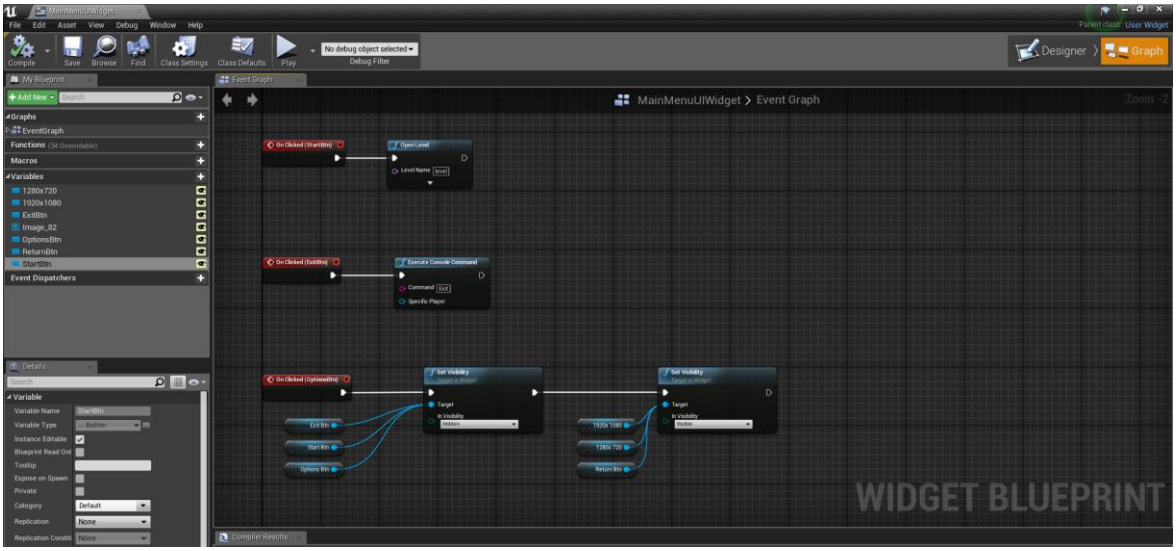
Za glavni meni potrebno je napraviti novu razinu, kod koje će se pritiskom na tipku "*Start Game"* pokrenuti već napravljena razina. Proces stvaranja korisničkog sučelja glavnog menija je isti kao i kod HUD-a. Potrebno je napraviti novi Widget i posložiti elemente sučelja po želji.



Slika 4.29: Dizajn korisničkog sučelja glavnog menija

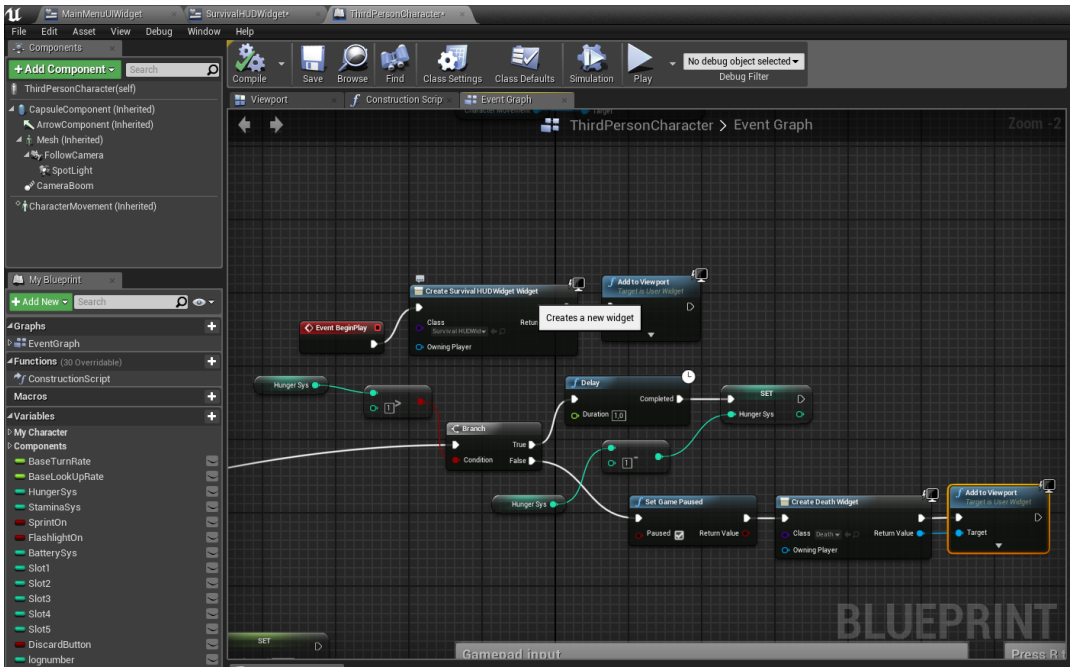
Naravno, potrebno je podesiti funkcionalnost menija kako bi se pritiskom na tipke postigao željeni rezultat (npr. pritiskom na tipku “Start Game otvara se glavna razina igre).

Funkcionalnost se postavlja u “Graph Editoru” i izgleda ovako.



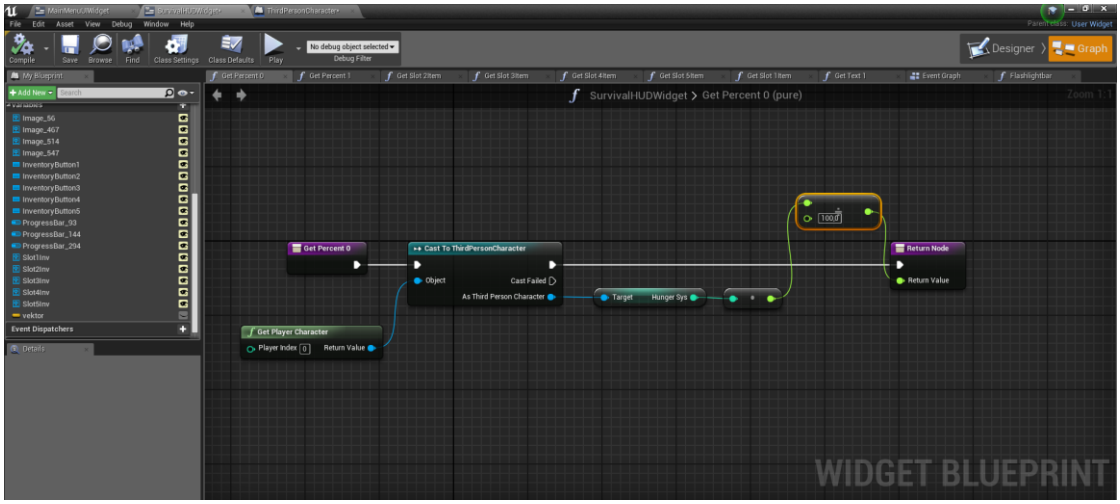
Slika 4.30: Funkcionalnost glavnog menija

Na isti način se postavlja funkcionalnost HUD-a. Sljedeći primjer pokazuje Blueprint sustava za zdravlje koji se kasnije spaja na traku zdravlja.



Slika 4.31: Blueprint sustava zdravlja

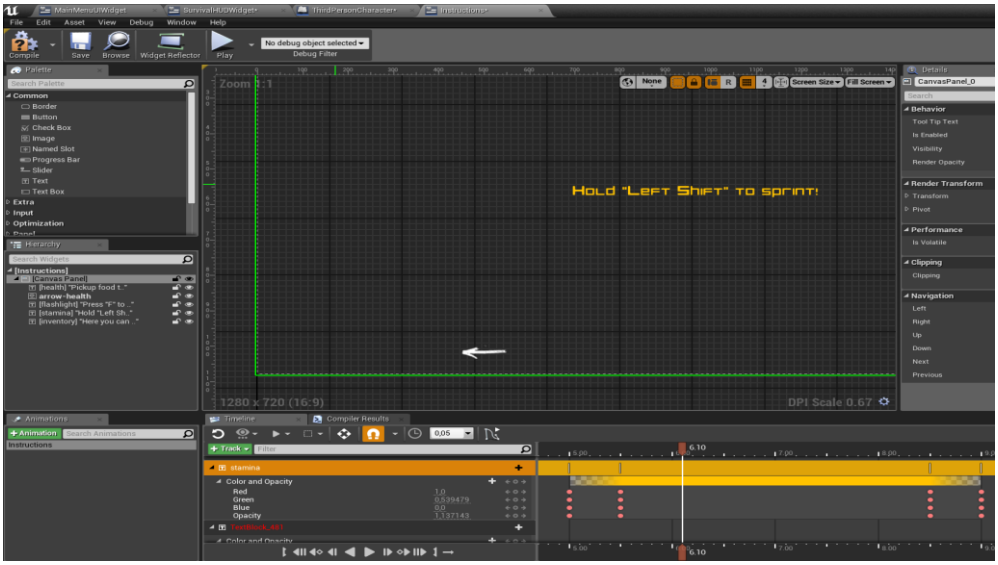
Ovaj primjer pokazuje spajanje sustava zdravlja na HUD traku zdravlja. Kako se postotak zdravlja prazni, traka se prazni ulijevo i postaje sve svjetlija. Ako zdravlje padne na nulu, igrač umire.



Slika 4.32: Sustav zdravlja povezan na HUD traku zdravlja

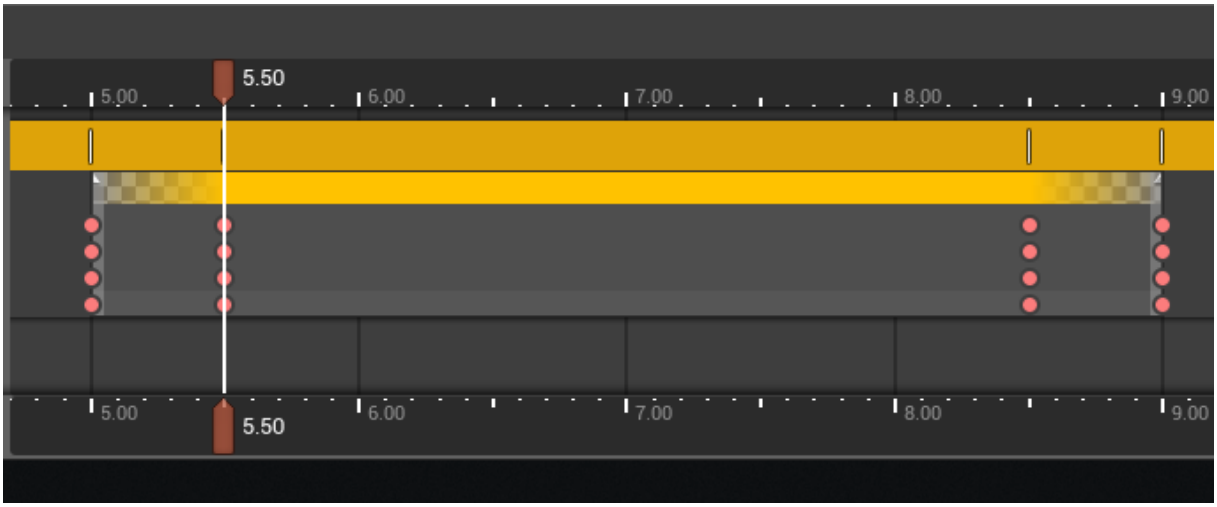
4.7. Animiranje

Pošto UE4 sam po sebi animira travu i drveće potrebno je samostalno animirati pojedine elemente korisničkog sučelja. Cilj je animirati grafike koje će igraču pri ulasku u igru objasniti čemu služe pojedini elementi HUD-a i pritiskom na koje tipke se izvršava određena radnja. Također, treba napraviti novi Widget Blueprint u kojem će se smjestiti i animirati grafike.



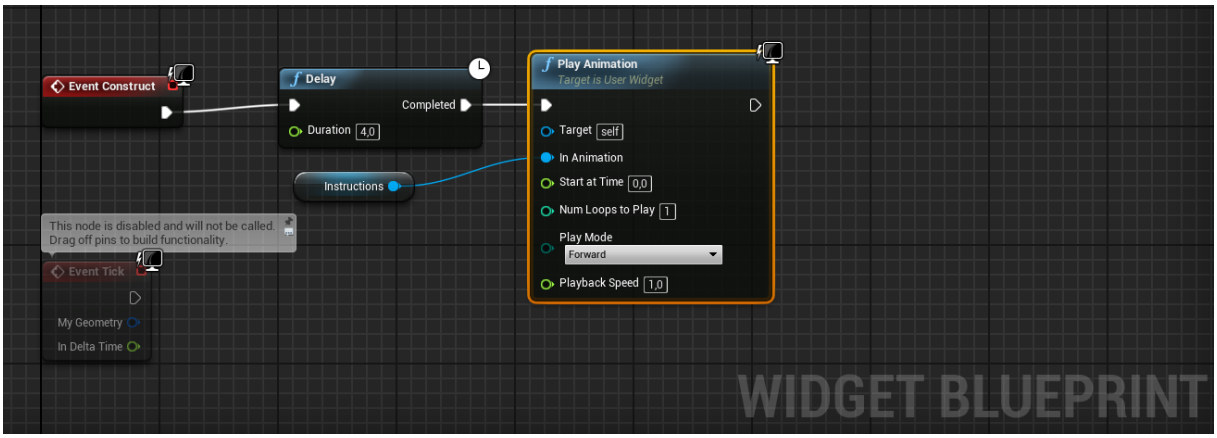
Slika 4.33: Prikaz animirane grafike HUD-a

Animiranje grafike funkcionira identično kao i u svakom programu za 3D modeliranje i animaciju. Uz pomoć sličica (keyframe-a), postavljamo parametre teksta koji se odnose na poziciju teksta, prozirnost teksta, i vrijeme za koje se tekst animira na ekranu.



Slika 4.34: Sličice animirane grafike

Nakon animiranja teksta, potrebno je podesiti da se animacija prikazuje pri ulasku u glavnu razinu igre.



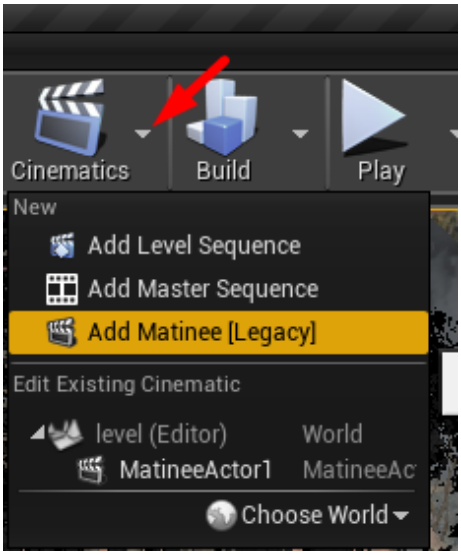
Slika 4.35: Blueprint animirane grafike

Graf prikazuje Blueprint animirane grafike. Postavljeno je kašnjenje animirane grafike od četiri sekunde ako igrač ima slabiju računalnu konfiguraciju i dođe do pada prikaza sličica u sekundi. Četiri sekunde su otprilike dovoljne da se igra stabilizira i u potpunosti učita na prosječnoj konfiguraciji.

Slika 4.36: Prikaz animirane grafike četiri sekunde nakon ulaska u igru

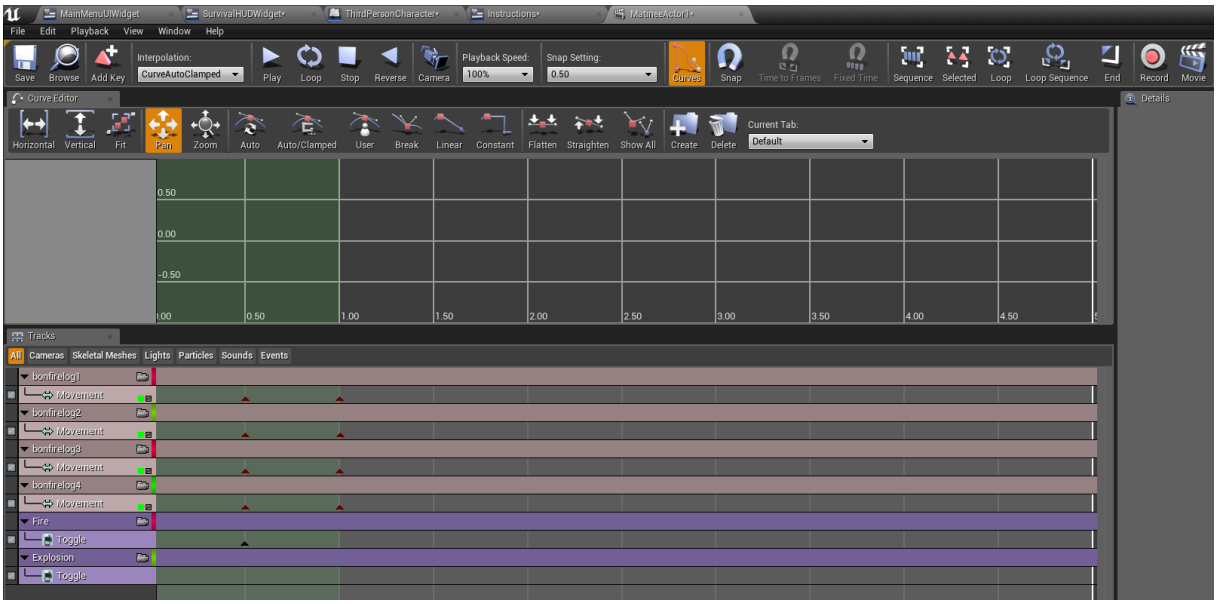
Također, potrebno je i animirati pojedine elemente u igri, u ovom slučaju vatru. Vatra će se moći zapaliti tek kada se skupi četiri komada drva. Prilikom dolaska igrača na sredinu jezera, igra će provjeriti ima li igrač potrebne resurse. Ako ima, mjesto odapinjanja (engl. *triggerbox*) će automatski zapaliti vatru i drvo će se automatski maknuti iz ruksaka i premjestiti u položaj u kojem će gorjeti.

Da bi se postigao takav efekt, koristi se "*Matinee Editor*".

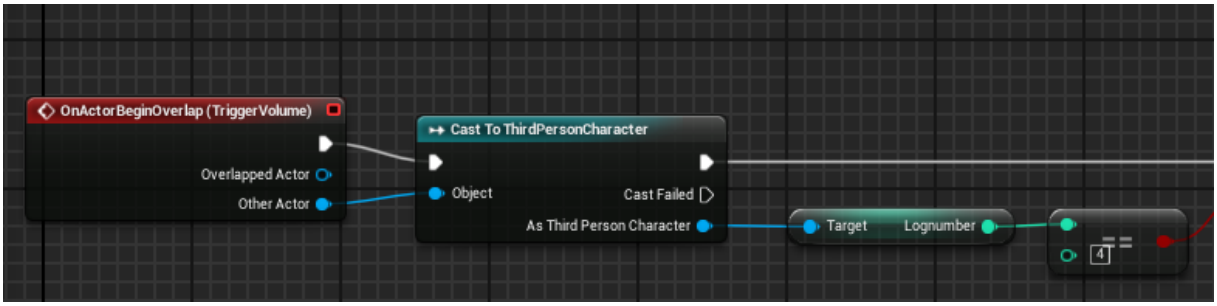


Slika 4.37: Otvaranje novog matine-a

Ovo funkcionira isto kao i animiranje grafike korisničko sučelja. Četiri komada drva postave se ispod površine zemlje. Dolaskom na mjesto odapinjanja sa četiri komada drva u ruksaku igrač aktivira matine. Pri aktivaciji, iz zemlje se podižu četiri komada drva na koje je stavljena animacija vatre koju nudi UE4, a drvo u ruksaku se uništava.



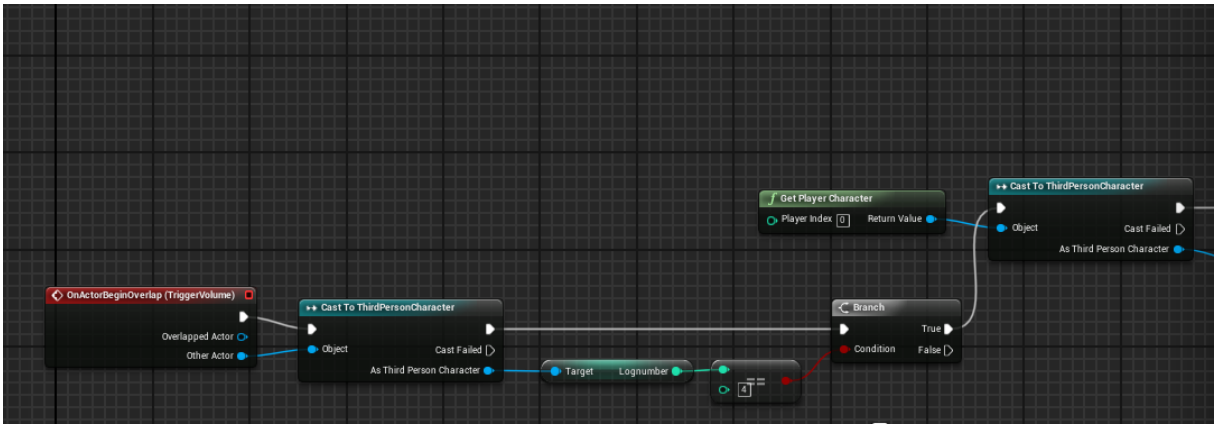
Slika 4.38: Animiranje matine-a



Slika 4.39: Blueprint za aktivaciju matine-a

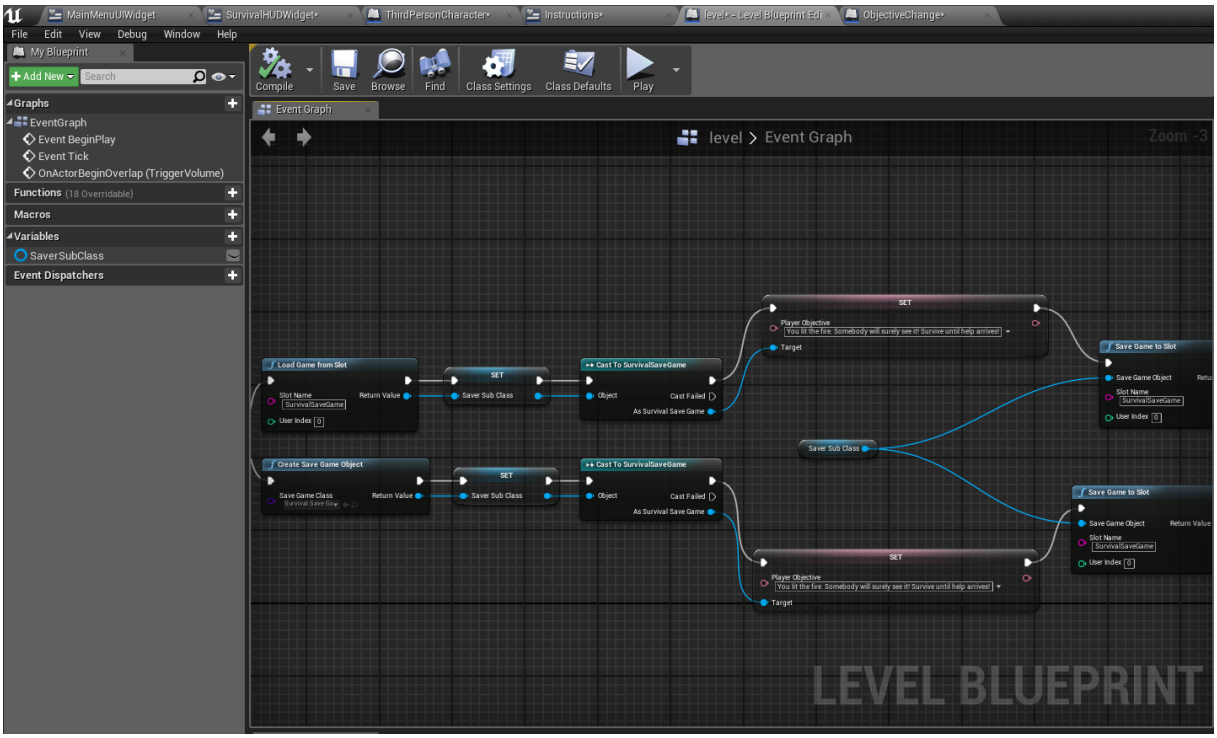
Također, u gornjem lijevom kutu mora se promijeniti i zadatak. Pri ulasku u igru zadatak glasi “Skupi 4 komada drveća kako bi zapalio signalnu vatru”. Nakon što je vatra zapaljena, zadatak se mora promijeniti. Cilj je promijeniti zadatak u “Zapalili ste vatru. Preživite toliko dugo dok ne stigne pomoć”.

Kako bi se to postiglo potrebno je napraviti novi Blueprint za promjenu zadatka. Taj Blueprint sastoji se od 3 dijela koji su međusobno povezani.



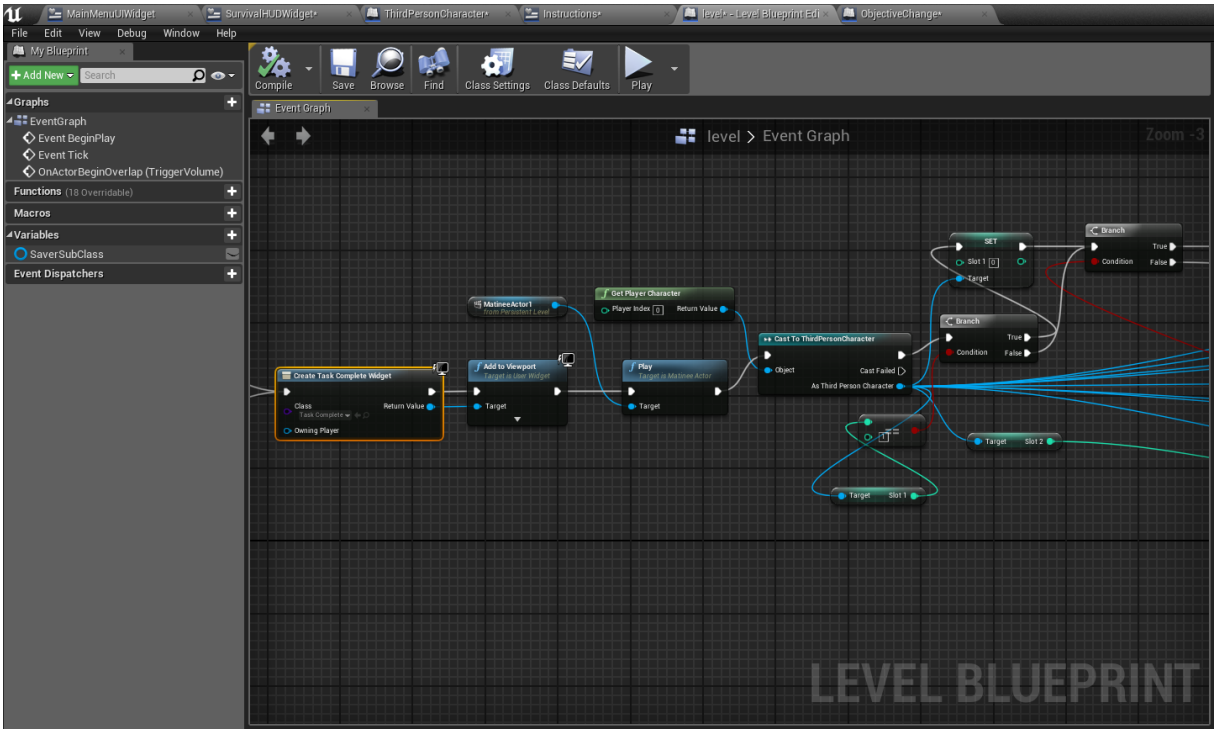
Slika 4.40: Blueprint za provjeru broja drva

Blueprint na slici 51. je podešen da igra provjerava dolazi li igrač na mjesto aktivacije s dovoljnim resursima. Ako je uvjet istinit tj. ako igrač dođe na mjesto aktivacije zadatak se mijenja u “Zapalili ste vatru. Preživite dok ne stigne pomoć”.



Slika 4.41: Blueprint za mijenjanje objektiva

Zadnji dio Blueprinta podešen je da se vatra aktivira dolaskom na predviđeno mjesto i da se drvo ukloni iz igračeva ruksaka.



Slika 4.42: Blueprint za aktivaciju vatre i uklanjanje drva iz ruksaka

Slika 4.43: Dolazak na mjesto aktivacije s resursima

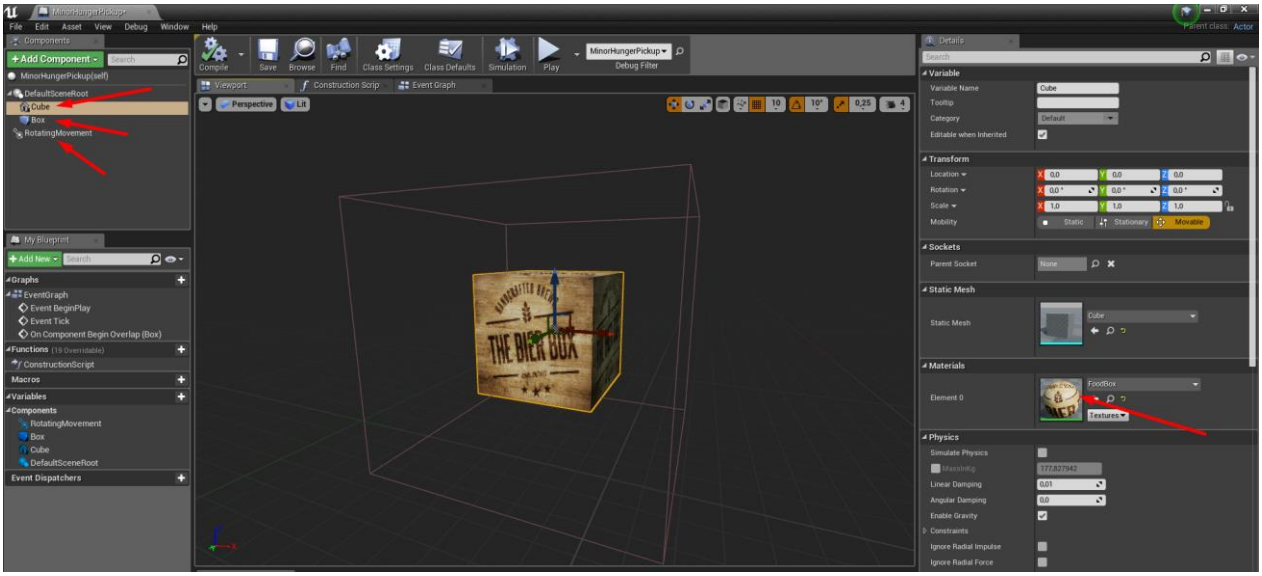
Slika 4.44: Aktivacija vatre i promjena zadatka

4.8. BVS (blueprint vizualno skriptiranje)

Iako je postupak blueprint vizualnog skriptiranja već objašnjen kroz par koraka izrade rada (Blueprinti za animaciju grafike, pokretanje vatre, provjere resursa), u ovom poglavlju prikazat će se Blueprinti pojedinih resursa koji pune trake zdravlja i ručne svjetiljke, funkcionalnosti sprinta/brzog trčanja, funkcionalnosti sustava zdravlja te blueprint za samu aktivaciju ručne svjetiljke.

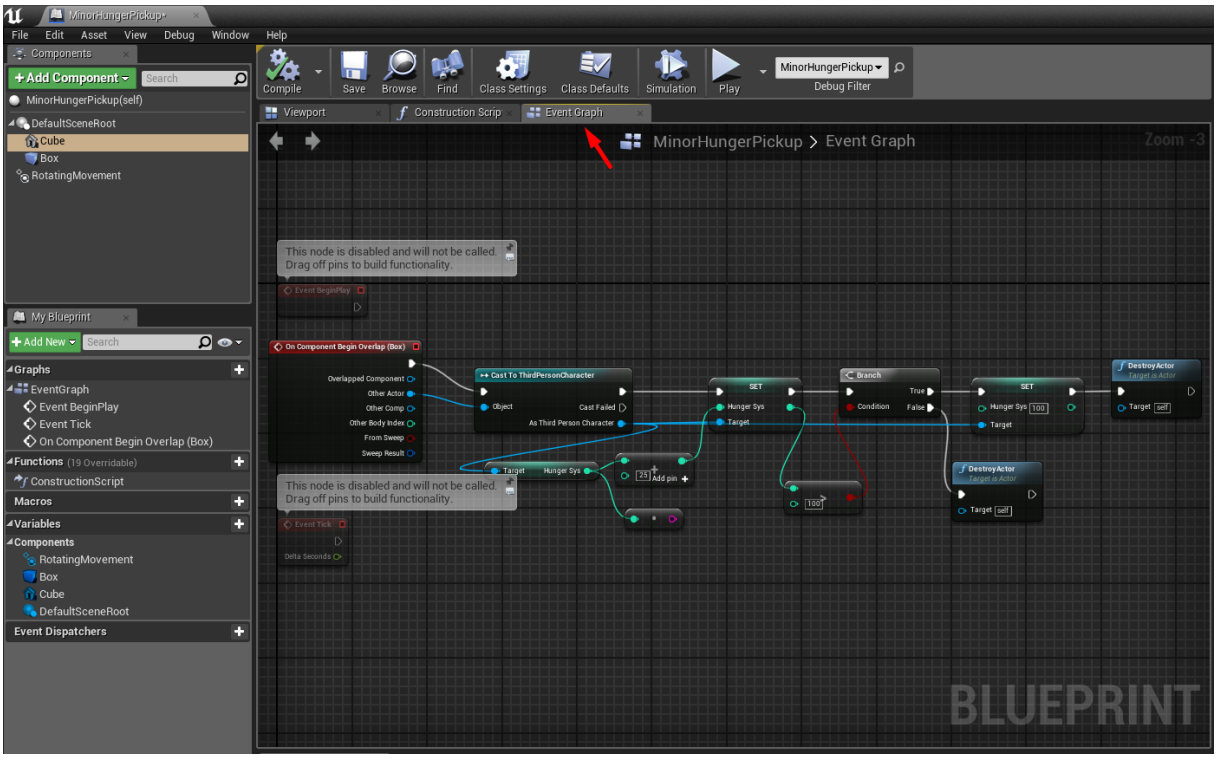
Kako bi igrač napunio svoje zdravlje/ručnu svjetiljku, potrebno je postaviti objekt/aktora u prirodi koji će prilikom dodira s igračem nestati i napuniti njegovo zdravlje ili ručnu svjetiljku.

U ovom primjeru potrebno je napraviti kutiju, povećati prostor aktivacije (engl. *hitbox*), dodati teksturu i rotaciju na sam objekt.



Slika 4.45: Dodavanje kutije, teksture, hitboxa i rotacije

Potrebno je i skriptirati da se igraču prilikom sakupljanja ove kutije zdravlje poveća za 25%. Taj Blueprint postavljamo u "*Event*" grafu ovog aktora i izgleda ovako.

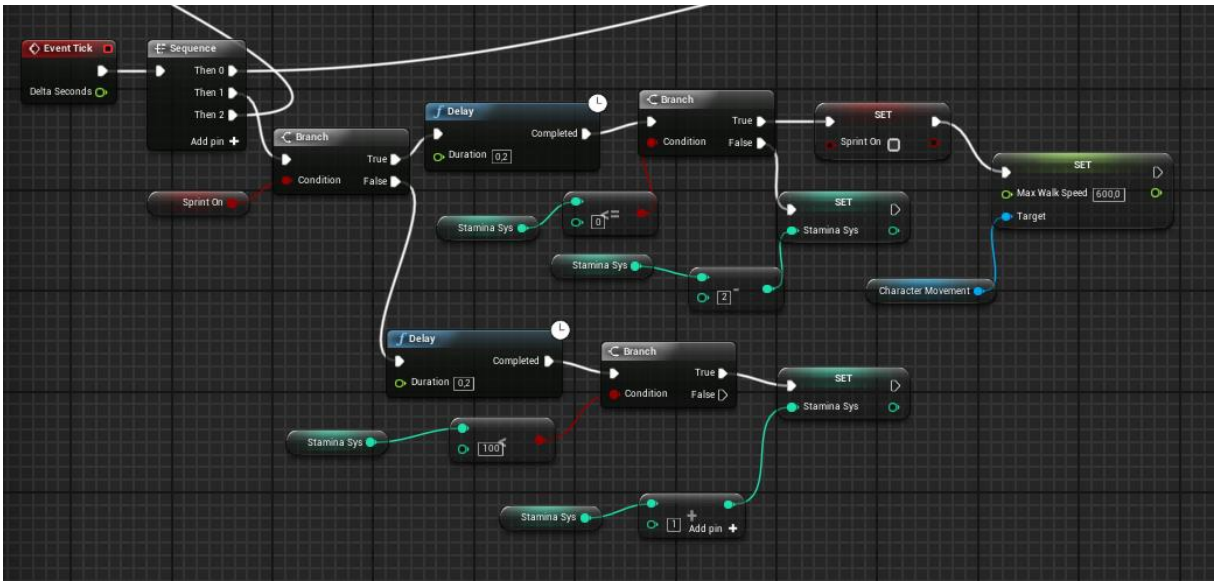


Slika 4.46: Blueprint funkcionalnosti kutije koja puni zdravlje

Na isti način stvara se kutija i Blueprint koji djeluje na ručnu svjetiljku, samo što se mora povezati na već prije stvoreni blueprint sustava ručne svjetiljke umjesto sustava zdravlja.

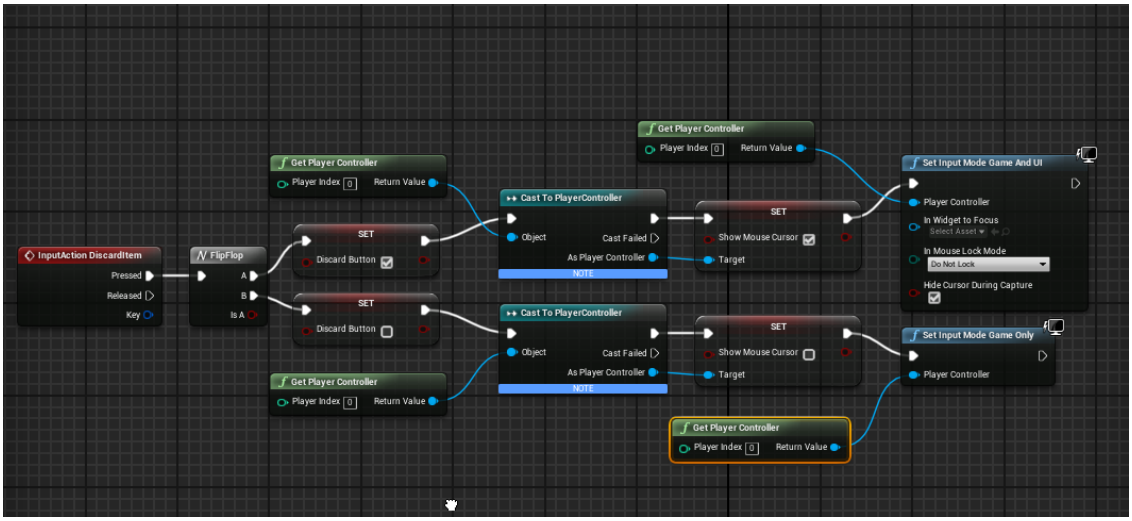
Slika 4.47: Izgled kutije u igri

Sljedeći Blueprint odnosi se na sustav brzog trčanja odnosno sprinta. Blueprint provjerava je li pritisnuta tipka "Shift". Ako je, igrač počinje brzo trčati što crpi kondiciju iz njegove trake. Kada se kondicija potroši do kraja igrač više ne može brzo trčati i počinje hodati. Pošto blueprint konstantno provjerava koristi li igrač brzo trčanja, ako opazi da igrač ne koristi, počinje automatski puniti kondiciju.



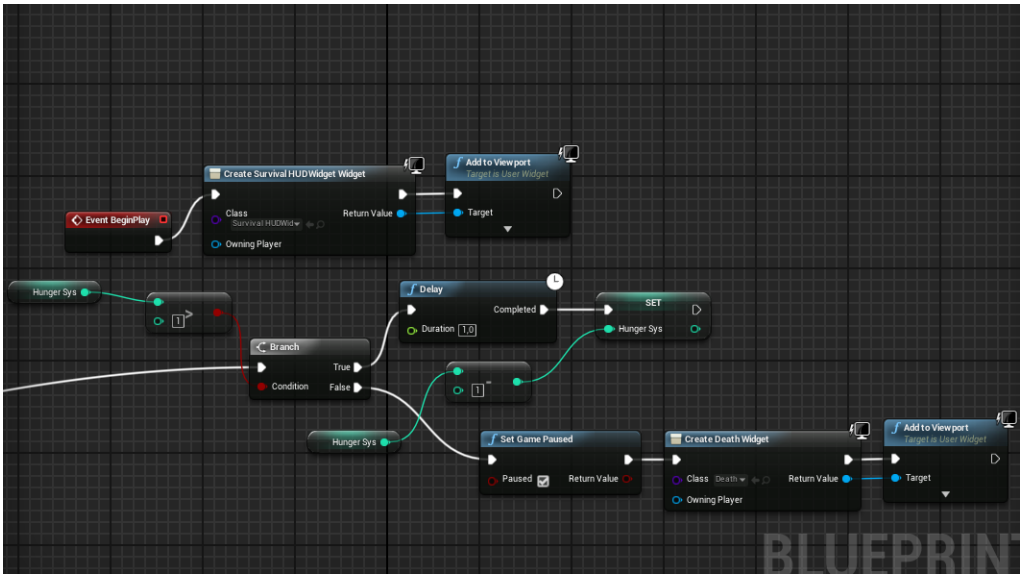
Slika 4.48: Sustav brzog trčanja

Na vrlo sličan način funkcionira i sistem ručne svjetiljke. Razlika je u tome što pritiskom na tipku jednom, igrač pali svjetiljku a ponovnim pritiskom ju gasi. Također, svjetiljka se gasi ako se isprazni na nulu no ne puni se automatski kao sprint, već je potrebno pobrati određene resurse koji ju pune. U ovom primjeru koristi se “Flip/Flop” prekidač koji služi da se pritiskom na istu tipku svjetiljka pali i gasi.



Slika 4.49: Pritisak na tipku "F" pali i gasi svjetiljku

Sustav zdravlja već je otprilike objašnjen na slici 42. i 43. no u ovom poglavlju se opisuje funkcionalnost kompletnog sustava. Sustav zdravlja prazni igračevo zdravlje za 1% svaku jednu sekundu. Igrač mora sakupljati kutije prikazane na slici 58. kako bi punio svoje zdravlje. Ako igrač ne uspije pokupiti kutiju na vrijeme, on umire.



Slika 4.50: Funkcionalnost sutava zdravlja

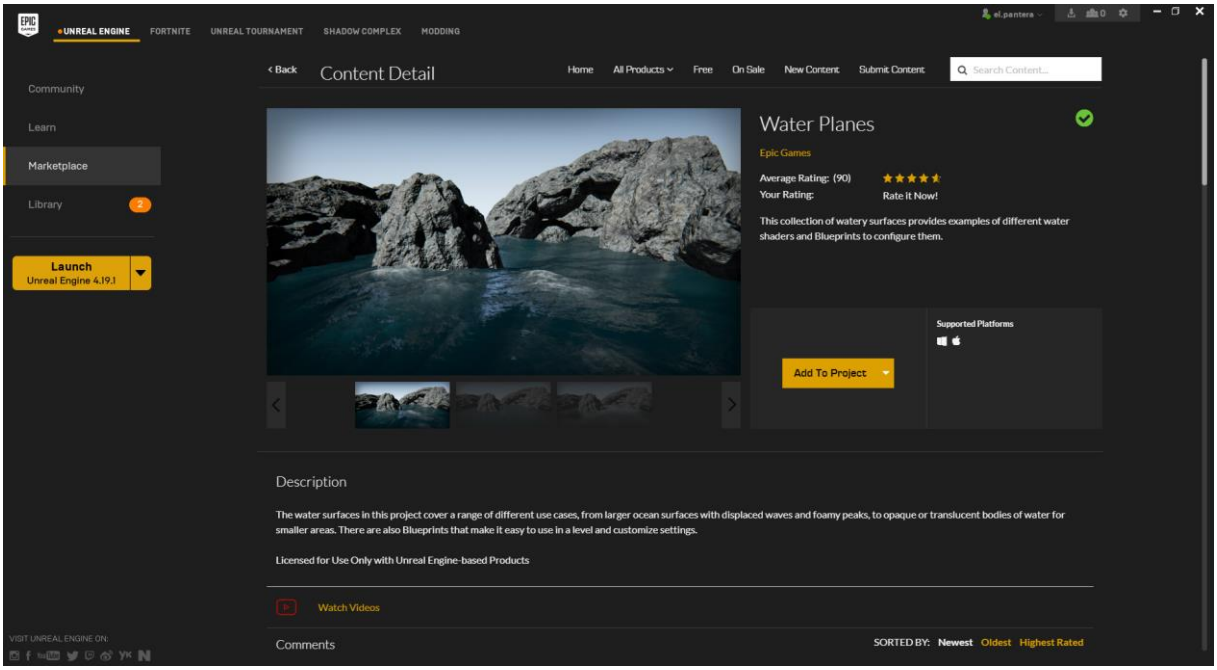
4.9. Detalji i popravljanje igre

U ovoj fazi igra je u potpunosti funkcionalna i već se sad može pokrenuti. Preostalo je još postaviti koliziju na objektima kako igrač neće moći prolaziti kroz drveće, kamenje i ostalo. Potrebno je dodati jezero i podesiti parametre da se uklopi sa scenom. Također, potrebno je i isključiti vidljivost ruku i nogu (engl. *mesha*) igrača, jer nismo modelirali i animirali lika već nam je UE4 dodijelio standardni "robotski mesh".

Kako bi se uključila kolizija, potrebno je u traci s modovima, pod postavkama kamenja i drveća u prirodi postaviti "*Collision Presets*" na "*Block All*". Tako igrač neće moći prolaziti kroz drveće i kamenje.

Slika 4.51: Podešavanje kolizije

Kako bi dodali jezero, u UE4 Marketplace-u je potrebno besplatno skinuti "*WaterPlanes*" model.

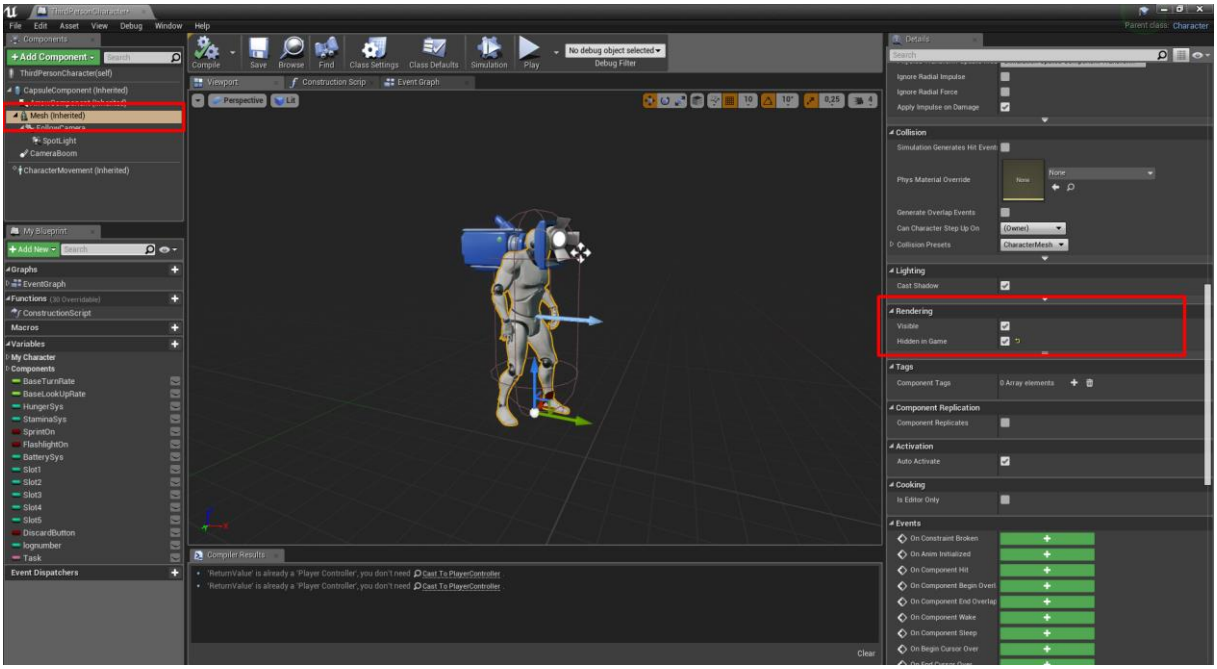


Slika 4.52: WaterPlanes model

Model se pojavljuje u prozoru "*Content Browser*". Potrebno ga je jednostavno povući na mjesto gdje je već prije napravljena predviđena udubina za vodu. Pod prozorom detalja podešavamo parametre kao što su veličina jezera, veličina valova, brzina valova, gustoća valova, površinska boja jezera i dubinska boja jezera.

Slika 4.53: Podešavanje jezera

Vidljivost glavnog lika može se isključiti odlaskom u Blueprint “ThirdPersonCharacter” označavanjem mesha i klikom na “*Hide in game*”.

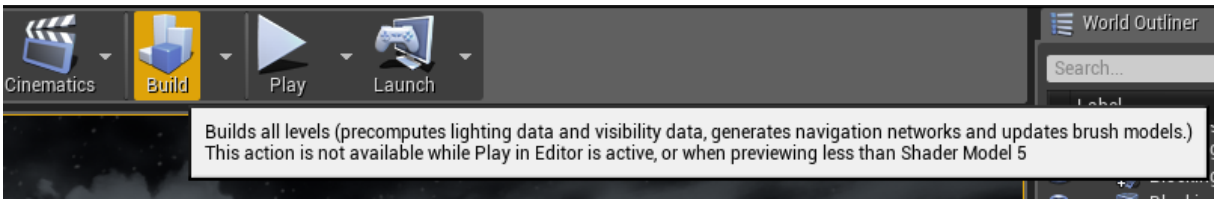


Slika 4.54: Sakrivanje mesha glavnog lika

Nakon popravka detalja u igri, ona je spremna da se pakira i publicira.

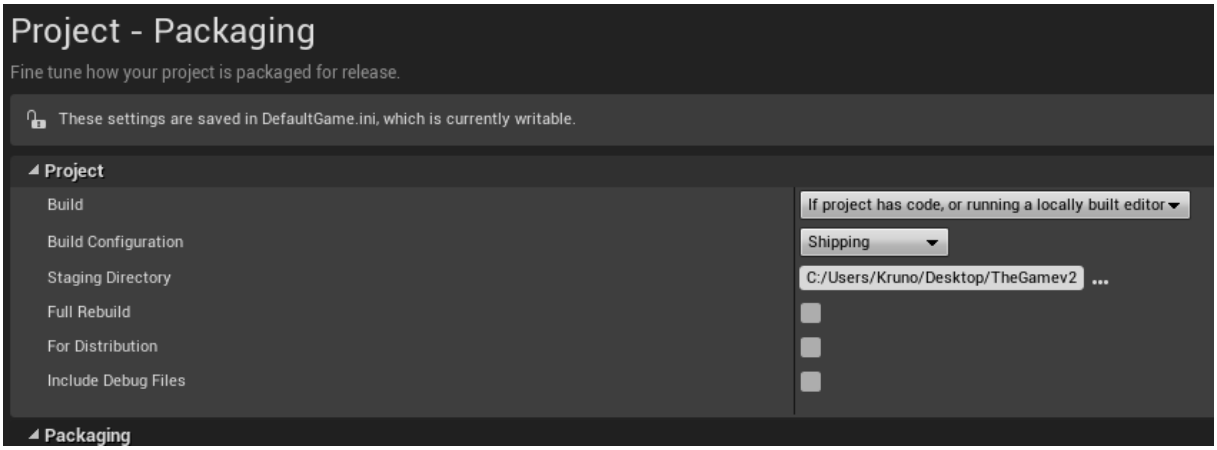
4.10. Pakiranje igre

Prije pakiranja igre potrebno je osvježiti rasvjetu kako igrač neće imati problema sa sjenama i teksturama. Taj proces može potrajati no potrebno ga je izvršiti prije pakiranja igre. Pokreće se pritiskom na alat “Build” u traci s alatima. Alat build također uz izgradnju svjetla pokreće i izgradnju refleksija i razine renderiranja detalja (LOD-Level of Detail).



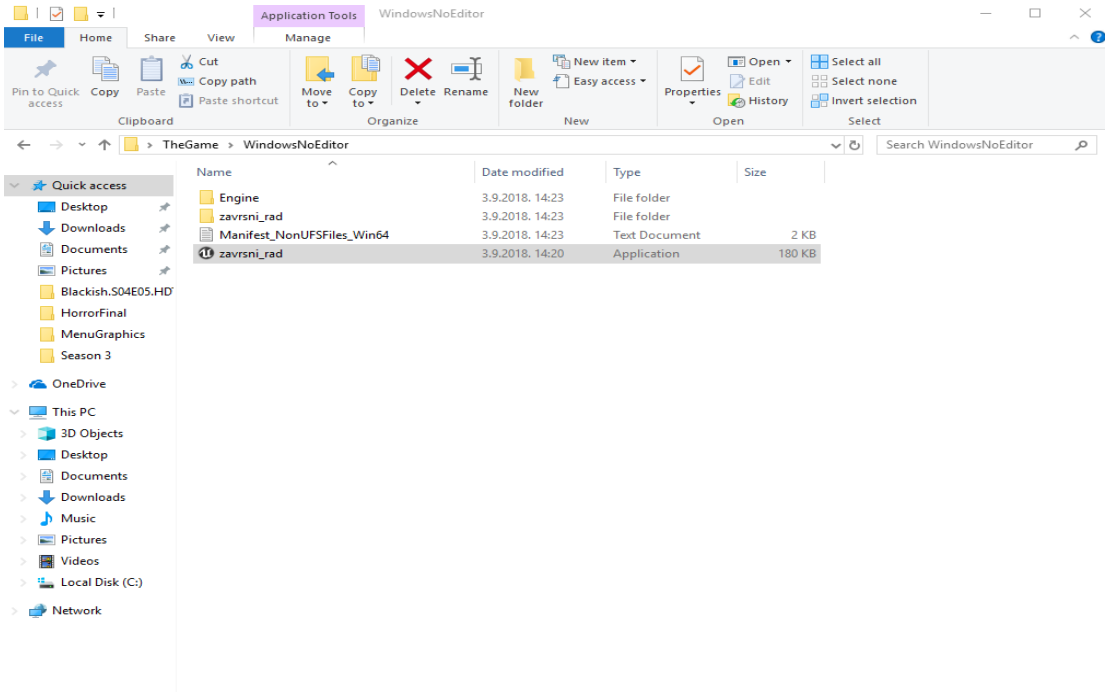
Slika 4.55: Alat za izgradnju rasvjete, refleksija i LOD-a

Nakon što postupak izgradnje završi, može se početi postupak pakiranja igre. Prije samog pakiranja savjetuje se provjera postavka pakiranja. Postavke pakiranja otvaramo klikom na *File-Package Project-Packaging settings.*



Slika 4.56: Postavke pakiranja

Potrebno je postaviti "*Build Configuration*" na "*Shipping*" i odrediti mapu u koju će se igra pakirati. Nakon podešavanja parametara, započinje se postupak pakiranja klikom *na File-Package Project-Windows-Windows(64bit).* Ovo je finalni postupak pri izgradnji igre. Nakon što postupak pakiranja završi, igra je spremljena u mapu i spremna je za otvaranje.



Slika 4.57: Mapa s sadržajem igre

4.11. Pokretanje igre

Slika 4.58: Igra (slika 1)

Slika 4.59: Igra (slika 2)

Slika 4.60: Igra (slika 3)

Slika 4.61: Igra (slika 4)

5. Zaključak

Cilj ovog rada nije bio samo izrada 3D računalne igre, već i upoznavanje s novim programskim alatom što je Unreal Engine 4. Zahvaljujući Epic Games-u koji je besplatno publicirao svoj alat na tržište danas pojedinac samostalno može napraviti računalnu igru. Unreal Engine 4 dozvoljava dizajneru da svoje djelo oživi bez programera uz pomoć blueprint vizualnog skriptiranja. Prije nego što je Epic Games oslobodio svoj UE4 to je bilo nezamislivo. Naravno, iskustvo rada u 3D programima uvelike olakšava ovaj proces jer UE4 daje mogućnost kastomizacije i moguće ga je prilagoditi da bude što sličniji programskom sučelju u kojem imamo iskustva.

Ako usporedimo igre od prije dvadesetak godina i danas, možemo vidjeti ogroman napredak, što se tiče grafike a i same igrivosti i optimizacije. Sve više ljudi provodi slobodno vrijeme upuštajući se u fiktivni prostor da se malo udalje od stvarnosti. Neki pak žive od profesionalnog igranja video igara. Kada bismo nekome prije dvadesetak godina spomenuli što nas čeka u budućnosti, ne bi nam povjerovao. Organizatori Olimpijskih igara najavili su da razmatraju mogućnost o uvođenju kompetitivnih video igara u svoj olimpijski program.

Proces izrade video igara nije težak kao što se smatra. Potrebno je imati ideju i uložiti vrijeme da se svlada baratanje tehnika potrebnih za izradu igre kao što su modeliranje, teksturiranje, animiranje, postavljanje rasvjete i u ovom slučaju blueprint vizualno skriptiranje. Ovaj projekt pokazuje kako jedna osoba samostalno može napraviti jednostavnu 3D igru, a kada zamislimo da u velikim korporacijama stotinjak ljudi radi na samo jednom malom segmentu igre, možemo vidjeti zašto igre danas izgledaju tako dobro. Prije dvadesetak godina malo tko je mogao predvidjeti ovakvu budućnost u ovoj industriji. Sada se nalazimo u istoj poziciji i možemo zaključiti je da ne postoji granica i da će iz godine u godinu biti bolje i naprednije. Sve što možemo reći za industriju video igara jest da ima vrlo svijetlu budućnost.

U Varaždinu, _____ _____

Datum Potpis

6. Literatura

[1] Ryan Shah: Master the Art of Unreal Engine 4 - Blueprints, lipanj 2014.

Dostupno na:

https://www.amazon.com/Master-Art-Unreal-Engine-Adventure/dp/1506119042

[2] Tim Sweeney: The end of the GPU roadmap, 2009.

Dostupno na:

http://mrl.cs.vsb.cz/people/gaura/agu/TimHPG2009.pdf

[3] Unreal Engine 4 vs. Unity: Which Game Engine Is Best for You?

Dostupno na:

https://www.pluralsight.com/blog/film-games/unreal-engine-4-vs-unity-game-engine-best

[4] D. Vusić, Z. Sabati, A. Bernik: 3D modeliranje u primjerima 1., Sveučilište Sjever, Varaždin, 2014.

[5] D. Vusić, A. Bernik, R. Geček: 3D modeliranje u primjerima 2., Sveučilište Sjever, Varaždin/Koprivnica, 2016.

[6] Unreal Engine 4 Službena dokumentacija (Unreal Engine 4 Documentation

Dostupno na: *https://docs.unrealengine.com/en-us* , rujan 2018.

[7] Epic Games, about

Dostupno na: *https://www.epicgames.com/site/en-US/about* , rujan 2018.

[8] Unreal Engine 4, features

Dostupno na*: https://www.unrealengine.com/en-US/features*, rujan 2018.

7. Popis slika

Slika 2.1: Epic Games Logo
Preuzeto s: https://digitalmediawire.com/wp-content/uploads/2017/03/Epic-Games.jpg 2
Slika 2.2.: Unreal Tournament 1999 Game Of The Year Edition
Preuzeto s: Https://Kotaku.Com/Unreal-Tournament-S-Facing-Worlds-Is-Still-The-Best-Mul-1659822174 3
Slika 3.1: Primjer "Sprint" Blueprinta Igre 5
Slika 3.2: Level Editor 7
Slika 3.3: Traka Kartica 8
Slika 3.4: Traka Izbornika 8
Slika 3.5: Traka S Alatima 9
Slika 3.6: Traka S Modovima 10
Slika 3.7: Prozor 11
Slika 3.8: Traka S Detaljima 12
Slika 3.9: Traka S Aktorima 13
Slika 4.1: Ideja - Igra "The Forest"
Preuzeto s: Https://Www.Rockpapershotgun.Com/2017/01/18/The-Forest-Review-Early-Access-2/ 14
Slika 4.2: Konačni Izgled Igre 15
Slika 4.3: Epic Games Launcher 16
Slika 4.4: Odabir Predloška 16
Slika 4.5: Stvaranje Nove Razine 17
Slika 4.6: Skaliranje Nove Razine 17
Slika 4.7: Modeliranje Razine 18
Slika 4.8: Parametri "Sculpt" Alata 18
Slika 4.9: Stvaranje Materijala 19
Slika 4.10: Unreal Engine 4 Teksture 20
Slika 4.11: Materijal (Blato, Trava, Kamen) 20
Slika 4.12: Postavljanje Materijala 21
Slika 4.13: Alati Za Bojanje 21
Slika 4.14: Teksturiranje Razine 22
Slika 4.15: Ue4 Marketplace 22
Slika 4.16: Dodavanje Kolekcije Prirodnih Modela I Elemenata 23
Slika 4.17: Kitedemo 23
Slika 4.18: Postavljanje Trave, Kanjema, Drveća 24
Slika 4.19: Unlit Način Rada 24
Slika 4.20: Rotiranje Rasvjete 25
Slika 4.21: Parametri "Skysphere" 25
Slika 4.22: Ažuriranje Rasvjete 26
Slika 4.23: Elementi Hud-A
Preuzeto s: Https://Www.Freepik.Com/Free-Photos-Vectors/Ui-Elements 26
Slika 4.24: Uvoz Elemenata Hud-A 27
Slika 4.25: Stvaranje Hud Widgeta 27
Slika 4.26: Postavljanje Hud-A 28
Slika 4.27: Parametri Traka 28

Slika 4.28: Elementi Korisničkog Sučelja Glavnog Menija
Preuzeto s:
Https://Www.Freepik.Com/Free-Photos-Vectors/Ui-Elements
Https://Secure.Static.Tumblr.Com/0f4e3a0d12da3d9add23b56ff01c1fb7/J153o0j/3spoj6l16/Tumblr_Static_Tumblr_Static_1ugjb31dmadc0sc80csgso88g_640.Jpg 29
Slika 4.29: Dizajn Korisničkog Sučelja Glavnog Menija 29
Slika 4.30: Funkcionalnost Glavnog Menija 30
Slika 4.31: Blueprint Sustava Zdravlja 30
Slika 4.32: Sustav Zdravlja Povezan Na Hud Traku Zdravlja 31
Slika 4.33: Prikaz Animirane Grafike Hud-A 31
Slika 4.34: Sličice Animirane Grafike 32
Slika 4.35: Blueprint Animirane Grafike 32
Slika 4.36: Prikaz Animirane Grafike Četiri Sekunde Nakon Ulaska U Igru 33
Slika 4.37: Otvaranje Novog Matine-A 33
Slika 4.38: Animiranje Matine-A 34
Slika 4.39: Blueprint Za Aktivaciju Matine-A 34
Slika 4.40: Blueprint Za Provjeru Broja Drva 35
Slika 4.41: Blueprint Za Mijenjanje Objektiva 35
Slika 4.42: Blueprint Za Aktivaciju Vatre I Uklanjanje Drva Iz Ruksaka 36
Slika 4.43: Dolazak Na Mjesto Aktivacije S Resursima 36
Slika 4.44: Aktivacija Vatre I Promjena Zadatka 37
Slika 4.45: Dodavanje Kutije, Teksture, Hitboxa I Rotacije 38
Slika 4.46: Blueprint Funkcionalnosti Kutije Koja Puni Zdravlje 38
Slika 4.47: Izgled Kutije U Igri 39
Slika 4.48: Sustav Brzog Trčanja 39
Slika 4.49: Pritisak Na Tipku "F" Pali I Gasi Svjetiljku 40
Slika 4.50: Funkcionalnost Sutava Zdravlja 40
Slika 4.51: Podešavanje Kolizije 41
Slika 4.52: Waterplanes Model 42
Slika 4.53: Podešavanje Jezera 42
Slika 4.54: Sakrivanje Mesha Glavnog Lika 43
Slika 4.55: Alat Za Izgradnju Rasvjete, Refleksija I Lod-A 43
Slika 4.56: Postavke Pakiranja 44
Slika 4.57: Mapa S Sadržajem Igre 44
Slika 4.58: Igra (Slika 1) 45
Slika 4.59: Igra (Slika 2) 45
Slika 4.60: Igra (Slika 3) 46
Slika 4.61: Igra (Slika 4) 46

8. Prilog

DVD