

# Izrada 3D okruženja u Unity razvojnom okruženju

---

Lapov, Krševan

Master's thesis / Diplomski rad

2018

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Graphic Arts / Sveučilište u Zagrebu, Grafički fakultet**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:216:004104>

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

Download date / Datum preuzimanja: **2025-03-07**



Repository / Repozitorij:

[Faculty of Graphic Arts Repository](#)



SVEUČILIŠTE U ZAGREBU

GRAFIČKI FAKULTET

KRŠEVAN LAPOV

**Izrada 3D okruženja u Unity  
razvojnem okruženju**

DIPLOMSKI RAD

Zagreb, 2018.



Sveučilište u Zagrebu  
Grafički fakultet

KRŠEVAN LAPOV

**Izrada 3D okruženja u Unity  
razvojnem okruženju**

DIPLOMSKI RAD

Mentor:

doc.dr.sc. Tibor Skala

Student:

Krševan Lapov

Rješenje o odobrenju teme diplomskog rada

# ZAHVALA

## **SAŽETAK**

Trend eksplozivnog rasta industrije video igara započeo je oko 2013. godine, a gazi naprijed iz godine u godinu uz sve veće profite. Kako se tom pozitivnom rastu tako skoro ne nazire kraj, posebno je zanimljiv dio grafičke industrije koji se bavi izradom video igara. Za izradu video igara po današnjim standardima potrebno je upotrijebiti svu moć dostupnih alata i usluga koje nude sustavi za izradu igara koji se nazivaju razvojna okruženja.

U ovom radu biti će opisan proces izrade različitih 3D modela u programu Blender, te izvoz istih u razvojno okruženje Unity, u kojem će u konačnici krojiti vizualno okruženje videoigre. Posebnu pozornost pridodati će se izradi mapa za teksture u Adobe Photoshopu, materijalima, svjetlosti te odnosu veličina zbog dodatne realnosti te geometriji samih modela.

Postupak izrade biti će detaljnije opisan u samom radu, no primijeniti će se rad u programima koje studenti koriste od samog početka na kolegijima Grafičkog fakulteta. Na taj način zainteresirani pojedinci moći će uvidjeti postupak izrade modela ili krajolika, koje je moguće iskoristiti u projektima koji su danas u rapidnom porastu, projektima poput videoigara ili filmova.

### **KLJUČNE RIJEČI:**

3D Modeliranje, 3D Okruženje, Blender, Unity, Photoshop, Video igra

## **ABSTRACT**

Rapid growth of the video game industry began at the end of 2013, and is going stronger each year it passes. As the end of this fast growing trend has not nearly get till the end, this part of graphic industry is especially interesting to observe. To create video games that match today's standards of quality it is necessary get advantage of every service and tool available. Systems that are providing this kind of services and tools are called game engines.

In this work, process of creating various 3D models in Blender will be described, as well as export to Unity, where it will create final 3D environment for video game. Special attention will be put on creating texture maps in Adobe Photoshop, creating materials, lighting and sizes in favor of more realistic look.

The process of creation solely will be explained in great detail later in this work, but along whole process, software of choice will be familiar to students of Faculty of Graphic Arts. That way, individuals can get a wider picture about creation of model or environment and usage of it into great projects like video games or movies.

### **KEY WORDS:**

3D Modeling, 3D Environment, Blender, Unity, Photoshop, Video game

# SADRŽAJ

1. UVOD.....	1
2. IZBOR IDEALNOG SOFTVERA.....	3
2.1. 3DS Max i Maya .....	3
2.2. Zbrush .....	3
2.3. Blender .....	4
3. SOFTVER ZA OBRADU FOTOGRAFIJE .....	6
3.1. GIMP .....	6
3.2. Krita .....	6
3.3. Adobe Photoshop .....	7
4. IZBOR IDEALNOG RAZVOJNOG OKRUŽENJA .....	9
4.2. Unity .....	12
4.3. Unreal Engine 4.....	12
4.4. CryEngine.....	13
5. IZABRANI SOFTVER .....	15
6. PRAKTIČNI DIO IZRADE 3D OKRUŽENJA- BLENDER.....	16
6.1. Skiciranje.....	16
6.2. Modeliranje u Blenderu.....	17
6.2.1. Odnos veličina.....	18
6.2.2. Vizualan stil .....	18
6.2.3. Low-poly Tehnika Modeliranja.....	18
6.2.4. Sustavi generiranja slike u Blenderu .....	21
6.2.5. Izrada modela za korištenje u sceni .....	22
6.3. Izrada teksture.....	25
6.3.1. «Odmotavanje» modela .....	25
6.3.2. Izrada teksture za 3D modele u Photoshopu.....	27
6.4. Izrada terena .....	31
6.5. Izrada završne scene .....	36
7. PRAKTIČNI DIO IZRADE 3D OKRUŽENJA- UNITY .....	38
7.1. Izrada novog projekta u Unity-u.....	38
7.2. Uvoz modela u Unity .....	43



7.3. Unity trgovina.....	45
7.4. Primjena materijala u Unity-u .....	46
7.5. Završni detalji .....	48
8. ZAKLJUČAK .....	51
9. POPIS SLIKA I TABLICA.....	52
10. POPIS MANJE POZNATIH RIJEČI.....	54
11. LITERATURA.....	55

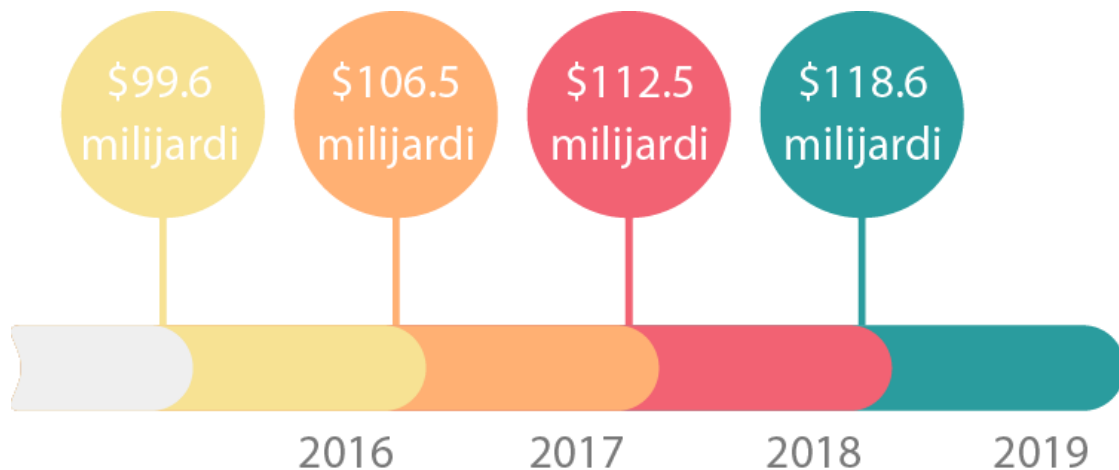
# 1. UVOD

U rujnu 2013. godine, Rockstar Games, američki izdavač videoigara je izdao Grand Theft Auto V i ušao u povijest. Tijekom prva 24 sata otkako je pušten u prodaju, igra je zaradila preko 800 milijuna dolara na tržištima diljem svijeta, što je više od bilo koje igre prije nje, zapravo više od bilo kojeg filma ili ploče u povijesti. Te iste godine industrija videoigara je ostvarila dvostruko veću zaradu od filmske industrije, 70,4 milijardi naspram 35,9 milijardi američkih dolara.

Zanimljiv je porast industrije video igara i njen uspjeh u nadmašivanju filmske industrije koja je držala broj jedan u svijetu zabave modernog društva toliko dugo vremena. Filmovi su objektivno gledajući pasivniji oblik zabave, gdje se od korisnika ne traži puno pozornosti ili akcije. Igre u drugu ruku od korisnika zahtijevaju upravo suprotno, zahtijevaju konstantnu pozornost, interakciju, reflekse, testira njihovu snalažljivost, način razmišljanja i pamćenje. Kako su igrači većim dijelom povezani s protagonistom i pričom u igri u usporedbi s filmom, tako je i iskustvo igre puno dublje. Beskrajni 3D svjetovi, mogućnosti odluke, kontrola glavnog lika, utjecanje na sudbine sebe samih te likova oko sebe, priče ispričane na drugačiji način igrače uvlači u svoj izmišljeni svijet milijama daleko od stvarnosti na način na koji filmovi do sada nisu uspijevali. No industriju videoigara ne čine samo duboke, kompleksne priče ispričane u predivno izgrađenim 3D svjetovima, ona je satkana od mnogo podvrsta igara, bile one rađene u 2D-u ili 3D-u. Danas je sasvim normalno na društvenim mrežama dobiti zahtjev za životom u igri Candy Crush Saga, od nekog bližeg ili daljeg roda u zrelijim godinama života. To je potpuno nova skupina korisnika koja donedavno nije postojala. Igranje na mobilnoj platformi tako je nadmašilo igranje na konzolama i PC-u po prvi puta u 2016. godini, kada je ostvarilo zaradu od 36 milijuna dolara više od prvog konkurenta [1].

Uz to, sve je popularniji i e-sport, mrežne igre toliko popularne i igrane da sa svim svojim značajkama više spadaju u kategoriju sporta nego igara. Održavaju se razna okupljanja, druženja i natjecanja diljem svijeta, svoje vještine iznimno dobri igrači streamaju na popularnim servisima Twitch i YouTube te ih pritom uživo gleda nekoliko desetaka tisuća ljudi. 2016. godine održano je svjetsko prvenstvo u SAD-u, u popularnoj mrežnoj igri League of Legends, a američka vlada je omogućavala vizu profesionalnim igračima te igre kategorizirajući ih kao profesionalne sportaše. To daje sasvim drugu dimenziju videoigrama, te

deminutivni izraz “igrice” koji se ustalio u svakodnevnom govoru nekako ne ide u isti koš s gore navedenim brojkama. Otkako su se prvi put pojavile u ljudskoj podsvijesti, tehnološki gledano davnih 1980-ih i 1990-ih, “igrice” su prešle dalek put da bi bile ovo što su danas [2].



Slika 1. Zarada industrije videoigara

Prošle godine, same igre na mobilnim platformama su pretekle industriju filma po zaradi. Po predviđanjima stručnjaka, ovi omjeri se neće puno mijenjati u bližoj budućnosti. Industrija videoigara gazi naprijed iz godine u godinu uz sve veće profite. Svijetloj budućnosti u svijetu videoigara ne nazire se kraj, stoga je posebno zanimljiv dio grafičke industrije, čiji će se razvoj istražiti u ovom radu.

## 2. IZBOR IDEALNOG SOFTVERA

### 2.1. 3DS Max i Maya

Gore navedene brojke itekako su impresionirajuće pa bi bilo zanimljivo otkriti kako napraviti nešto poput toga, nekakve modele koji će u konačnici tvoriti okruženje u kojem će se igrač moći kretati. Kratkom pretragom na mreži svih mreža dolazi se do brzih i jasnih informacija o tome kakva su uopće ulaganja potrebna za nekakav pokušaj u ovoj industriji. U modeliranju se najviše koriste programi od Autodesk, 3D Studio Max i Maya, a licenca za svakog od njih pojedinačno košta 1.984,40 \$ po godini [3]. 3DS Max je kompatibilan za MacOS i Windows, dok je Maya program samo za Windows operativni sustav. Uz to što je potrebno kupiti softver, potrebno ga je i nekako naučiti. Ovi programi redom su korišteni u profesionalnim studijima, pa je do materijala za učenje teško doći, ali su oni zato profesionalniji i konkretniji. Autodeskova služba jako je ažurna u rješavanju problema, a to je jedan od glavnih razloga zašto su najkorišteniji među velikim korporacijama ali i u manjim studijima.

### 2.2. Zbrush

Sljedeći izbor koji se učestalo spominje je Zbrush, relativno noviji program koji se koristi u svijetu izrade video igara. Cijena od 895\$ jednokratno vrijedi za licencu koja obuhvaća jednog korisnika, a može se instalirati na Windows i MacOS te se na njima i koristiti, pod uvjetom da je u istom trenutku pokrenuta samo jedna verzija. Iako je u utrku došao kasnije, intuitivno sučelje prilikom modeliranja i oblikovanja (*Sculpting*) u kombinaciji sa Substance Painterom (program za izradu materijala koji se kasnije primjenjuju na modelima) pruža jako kreativan radni tok [4].

Kako se već da zaključiti, za izradu ikakvog materijala koji će činiti virtualno okruženje, potrebno je podosta ulaganja. Za okušavanje u nečem ovakvom trebalo bi se skupljati mjesecima. Srećom, postoje programi otvorenog koda, kojima se bavi zajednica razvojnih programera, prikupljaju prihode od velikodušnih donacija te time financiraju svoj rad, često radeći i potpuno besplatno. Sve to da bi omogućili širem krugu ljudi korištenje računala u jednake svrhe kao i ljudi koji imaju plaćeni softver, neovisno o budžetu kojim korisnik raspolaže. Takav softver češće ima nedostataka i problema od softvera koji na primjer izrađuje tvrtka

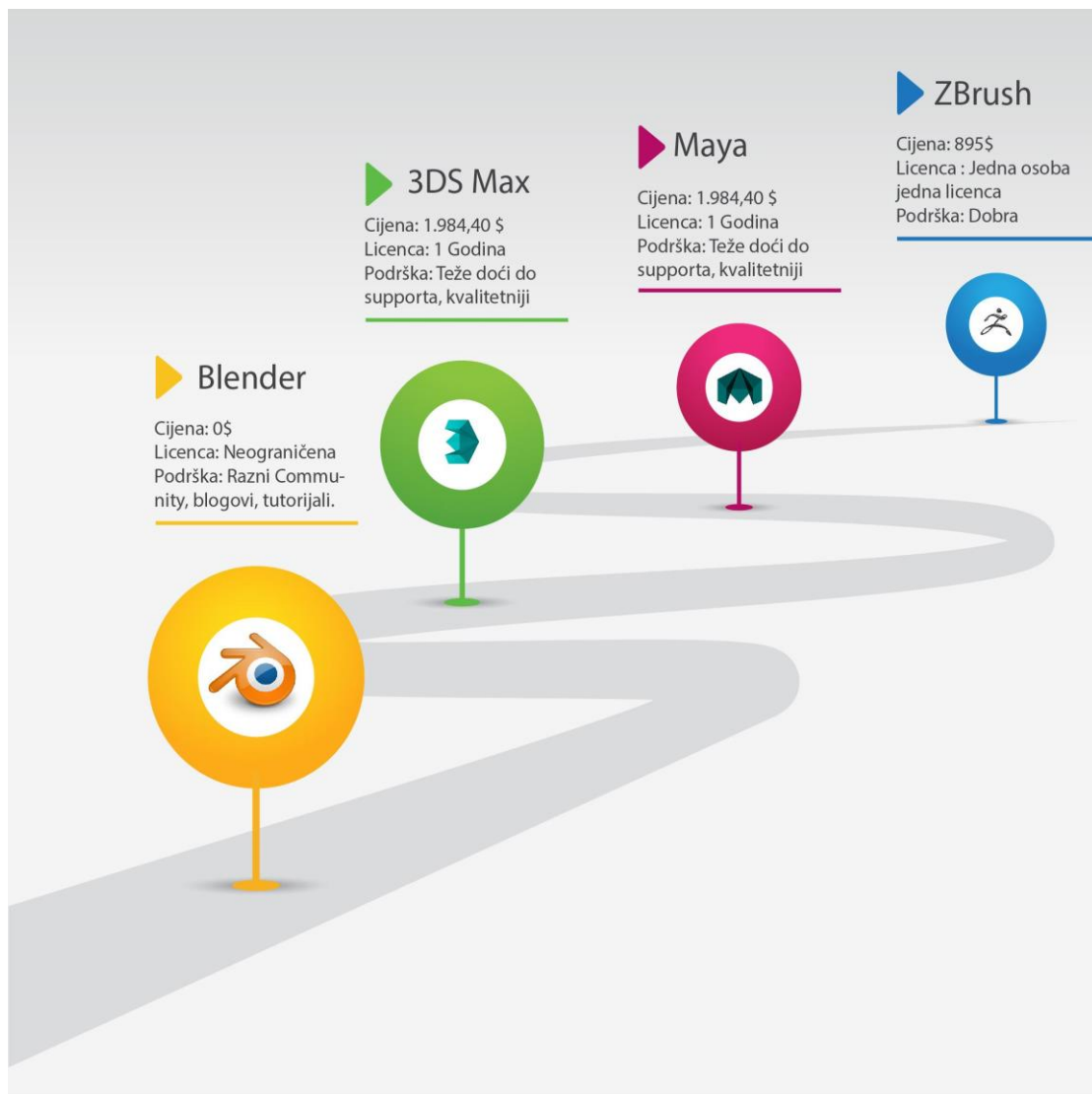
specijalizirana za uski dio kojim se bavi, ali širenjem zajednice programi se sve brže i bolje razvijaju, svakim danom rastu i daju krajnjim korisnicima sve više mogućnosti.

### **2.3. Blender**

Tako se dolazi do alternative gore navedenim programima. Blender, program otvorenog koda, u kojem se može modelirati, teksturirati, montažirati (*eng. rigging*), animirati, oblikovati (*eng. sculpting*), izrađivati (*renderirati*) itd. Uglavnom, u Blenderu se može napraviti gotovo sve što se zamisli u svijetu 3D-a, a uz sve to njegova cijena je 0\$. Program pomoću kojega se može toliko puno napraviti je besplatan za osobnu i komercijalnu uporabu.

Unatoč svemu tome, Blender je izbjegavan u profesionalnom okruženju. Ima svojih mana, korisničko sučelje je nepraktično i nezgrapno za početnike, princip rada je drugačiji nego u sličnim konkurentnim programima, a podrška koju studiji uživaju od strane proizvođača u skupim programima igra veliku ulogu. Princip rada nešto je drugačiji, pa svi modeleri koji su navikli raditi u skupljim opcijama, radije ostaju na nečem provjerenom i pouzdanom unatoč velikom napretku koji je Blender ostvario zadnjih par godina. Njegova trenutna verzija je 2.79, a uskoro će doći inačica 2.80, s puno noviteta, koja će donijeti podosta intuitivnije sučelje, kojim će se približiti konkurenciji vizualima, ali i privući početnike. Sve to Blender svrstava pod valjanu alternativu skupocjenom softveru, koju svi mogu priuštiti i koja je sistemski nije toliko zahtjevna. Blender potražuje za pokretanje minimalno 2GB radne memorije, dok ostatak gore navedenih rješenja zahtjeva duplo, 4 GB radne memorije. No može se pokretati i na sistemima s manje RAM-a, što znači da ga može pokrenuti veći broj uređaja i širi spektar ljudi koje interesira 3D a nemaju jake sistemske specifikacije svoga računala.

Uz sve to, kako je Blender program otvorenog koda, postoji velik broj blogova, YouTube kanala, tutorijala, online tečaja, te pojedinaca koji su bili na istom mjestu kao i Vi na početku, te su spremni pomoći [5]. Često se rješenje problema može pronaći u seriji slika, teksta ili videa. Zajednica korisnika toliko se proširila te je baza podataka, rješenja i savjeta poprilično porasla pa je početak učenja Blendera sada dosta lakši nego prije par godina.



Slika 2. Razlozi zašto Blender

Blender se kao program uči i na nekoliko kolegija na Grafičkom fakultetu, na kojima se studenti uče osnovama modeliranja te nekim naprednijim stvarima poput animiranja, stoga je to još jedan od razloga zašto je korišten u ovom radu. Netko tko je upoznat s osnovama lakše će krenuti u izradu modela i dublje istraživanje samog procesa izrade 3D okruženja. Dok netko tko i nema široko predznanje, ako želi lako će doći do potrebnih informacija u rastućoj zajednici razvojnih programera, 3D modelera i umjetnika.

### **3. SOFTVER ZA OBRADU FOTOGRAFIJE**

Adobeov paket programa prvi je odabir velikom dijelu korisnika u grafičkoj industriji. Od digitalne izrade sadržaja, sadržaja za korištenje u webu, sadržaja za novine, časopise, letke, vizitke, apsolutno bilo kakve pripreme za tisak, obrade fotografije, animiranju, video, filmskoj i televizijskoj produkciji. Polako u svojim programima implementiraju razne opcije za izradu i obradu 3D sadržaja, ne prepuštajući tako niti jedan segment multimedijalne manipulacije konkurenciji.

Što se obrade digitalne fotografije tiče, Adobe ima par konkurenata koji su vrijedni spomena. Softver za obradu fotografije će se koristiti kako bi se izradile teksture koje će koristiti 3D modeli izrađeni u Blenderu. Te teksture mogu biti različite, one mogu davati obojenje pojedinom modelu, mogu mu na pojedinim mjestima davati refleksiju, prozirnost, izbočine, prividne detalje, te mnoge druge karakteristike koje će u konačnici tvoriti materijal toga 3D predmeta.

Iako Blender u sebi ima dio za obradu fotografije u svom paketu alata, koji bi mogao poslužiti svrsi, uporaba zasebnog programa za obradu daje puno više mogućnosti i fleksibilnosti. Takav način rada je i praksa u igraćoj industriji. Programi koji će se razmotriti su: GIMP, Adobe Photoshop i Krita.

#### **3.1. GIMP**

GIMP je program otvorenog koda, ima puno funkcionalnosti, poprilično je moćan, ali isto tako je i malo teži za naučiti od preostalog dvojca. Nezgrapnog je sučelja koji kao da odiše nekim starijim vremenima i kao da žudi ka opasnom faceliftingu. Ima pregršt dodataka koji ga čine moćnim, a u velikom broju slučajeva to su stvari koje ostali kandidati imaju implementirano. Kod GIMPA se ti dodaci moraju naknadno instalirati, što ponekad predstavlja problem jer se ne održavaju svi dodaci jednako za sve operativne sustave i mogućnost je da budu nefunkcionalni [6]. Velika prednost je ta što je GIMP kao program otvorenog koda besplatan.

#### **3.2. Krita**

Krita je isto tako program otvorenog koda koji je dosta noviji od GIMPA, lijepo sučelje te popriličan broj mogućnosti i funkcionalnosti. Velika zajednica korisnika učestalo prijavljuje probleme a programeri ih ažurno ispravljaju. Čini se kako bi u

budućnosti mogao pružiti još veću konkurenciju Adobeovom monopolu u profesionalnom korištenju. Nažalost, do sada je velik broj korisnika prijavio pad performansi u korištenju Krite, ali to je slučaj kod većine programa u ranom stadiju. Kako je otvorenog koda, Krita je besplatna i to je naravno veliki plus. [7]

### **3.3. Adobe Photoshop**

Adobe Photoshop je program tvrtke Adobe koja dominira tržištem softvera u grafičkoj industriji već desetljećima, što je u svijetu tehnologije zaista mnogo. Pregršt funkcionalnosti, lijep i profesionalan izgled, iznimne performanse s obzirom na broj različitih uređaja na kojem se izvodi, konzistentnost alata i usluga koje nudi te optimiziranost i stabilnost u radu su odlike koje ga postavljaju kao izbor broj 1 velikoj većini grafičkih firmi, agencija i mladih profesionalaca u različitim sferama te branše. Učestalost nadogradnji, kvalitetna korisnička podrška i velik broj mogućnosti koje savršeno rade od prvog pokretanja samo neke su od odlika koji ga čine tako dominantnim. Iznimno kvalitetan komad softvera koji ima ogroman broj tutorijala i neovisno o korisničkoj podršci jer ima i ogroman broj korisnika koji su skloni dijeljenju svojih radnih navika, kratica i savjeta širom interneta. Korištenje Adobe Photoshopa se za razliku od druga dva izbora plaća, ali zato često ima različitih pogodnosti. Jedna od pogodnosti je 60% popusta na bilo koju aplikaciju Adobeovog paketa za sve studente i profesore. Ako ste u prosvjeti, Photoshop se tako može nabaviti za 9.99\$ mjesečno, što je puno prihvatljivije od cifri koje se kreću oko programa za 3D modeliranje [8].



Kako se na nekim kolegijima na Grafičkom fakultetu učio Adobe programski paket, u ovom radu će se koristiti upravo on u procesu izrade tekstura za 3D modele. To nikako ne znači da se pomoću preostala dva alata ne mogu dobiti isti rezultati, samo uz malo više truda i na malo drugačiji način.



Slika 3. Razlozi zašto Photoshop

## 4. IZBOR IDEALNOG RAZVOJNOG OKRUŽENJA

Razvojno okruženje je skup alata i usluga u kombinaciji s grafičkim sučeljem koji čine sustav potreban za izradu video igara. Razvojna okruženja za izradu video igara mogu biti specijalizirana za izradu 2D ili 3D igara, dok neki od njih podržavaju izradu igara obje vrste. Razvojna okruženja razmatrana u ovom radu pokrivaju sve tipove igara. U njima se mogu izraditi FPS (*First-Person Shooter*) igre - pucačine, RTS (*Real-Time Strategy*) igre - strategije, trkaće igre, TPS (*Third-Person Shooter*) - igre iz trećeg lica ili RPG (*Roleplaying game*) - igre na poteze.

Služe primarno kako bi svojim alatima olakšali posao programerima i dizajnerima u sastavljanju igre, izradi dijelova i modela za igre te njihovoj integraciji u igru. Ovisno o odabranom okruženju, igra koja se radi može ciljati na neku od popularnih igraćih platformi (Playstation, Xbox, Nintendo, PC, Android, iOS...), a neka razvojna okruženja podržavaju prilagodbu igre na više platformi. Iako razvojno okruženje nije u potpunosti potrebno za izradu igre jer je moguće napisati i iskodirati određene usluge koje se pružaju u razvojnom okruženju, to samo znači da je svejedno potrebno iskodirati dijelove tog razvojnog okruženja pomoću kojega bi se igra napravila, što je u konačnici ništa drugo nego zasebno razvojno okruženje. To se u praksi prakticira, samo u malom broju slučajeva, kada je igra nezahtjevna te je potreban jako malen dio razvojnog okruženja da bi u potpunosti funkcionirala.

Odabir razvojnog okruženja je najbitniji dio na koji razvojni programeri i 3D umjetnici moraju posvetiti pažnju kako bi izvukli najviše od svog proizvoda u konačnici. Prije izrade svega se treba odgonetnuti koje će se razvojno okruženje koristiti kako bi korisnici/igrači dobili najbolje iskustvo. Okruženja u kojem će se igra razvijati danas najčešće nude široku paletu mogućnosti te se u njima mogu izgraditi klasične arkadne igre poput *Tetrisa*, *Zmijice*, *PacMan-a* i druge, ali i puno moćnije igre koje koriste najnovije riječi tehnologije i moćne vizuale, igre poput GTA ili *Assassin's Creeda*. Danas je prisutan velik broj razvojnih okruženja i izbora ima na pretek.

Kako će u ovom radu biti prikazana izrada 3D okruženja, odluku ćemo donijeti po 3D prednostima i manama svakog. Isto tako, kao ulaz u izradu 3D okruženja, ne može se reći da je svejedno koliko će taj projekt koštati, pa želimo biti obzirni i u tom smislu. U ovom radu će se pokušati izabrati jedan od najboljih, najsvestranijih i najjednostavnijih.

#### **4.1. Odlike današnjih razvojnih okruženja**

Današnja razvojna okruženja došla su do te razine da mogu generirati vrlo impresivne grafičke vizuale. Kako bi se to što jednostavnije izvelo, već je navedeno, mogu se uvesti i različite grafike iz drugih platformi. Za većinu igara razvojna okruženja nude arhitekture s visokim performansama računanja vizuala, omogućavajući tako da igra ima najbolje moguće performanse u odnosu na vizuale ovisno o platformi na kojoj se izvode.

Uz vizualni aspekt video igre, za velik dio njezinog doživljaja zaslužna je i glazbena podloga. Uz zvučne efekte interakcije, kretnji, glasova pojedinih likova pa sve do glazbe u pozadini nekih scena ili u izborniku, glazbena podloga može biti ta koja čini razliku između osrednje igre i nezaboravnog iskustva. Danas se zvučni efekti mogu izraditi od nule uz pomoć raznih programa za slaganje zvukovnih isječaka, bez potrebe snimanja ikakvoga zapisa. No ako se odluči u izradu glazbene podloge ući bez ikakvog softvera za njezinu izradu, u trgovini integriranoj u većini razvojnih okruženja moguće je pronaći velik broj audio zapisa koji se nakon toga mogu kombinirati da tvore potpuno novu podlogu. Ovisno o ponudi integrirane trgovine u ponudi se mogu naći i audio zapisi specificirani za razvojnu platformu na kojoj će se igra izvoditi. Razvojna okruženja nude razne pametne opcije kombinacije zvukova koje u sebi imaju već integrirane. S tim pametnim opcijama razvojni se tim može usredotočiti na kompoziciju zvuka u kontekstu radnje u igri ili dodavati zvučne efekte po potrebi u potpunosti kako su to i zamislili.

Razvojna okruženja prate sve trendove igrace industrije, prilagođavaju se potrebama i učestalo se nadograđuju. U zadnje vrijeme sve više je igara s opcijom mrežnog igranja, bilo lokalnog ili preko interneta s ljudima iz svih krajeva svijeta. Velik broj igrača se voli udružiti sa svojim prijateljima u istoj igri ili aplikaciji,

neovisno radilo se o PC-u, igraćoj konzoli ili mobilnom operativnom sustavu na kojem se ista izvodi. Kod nekih tipova igara, mrežno igranje zahtjeva veliku procesorsku moć servera koji se brine da se igra izvodi bez smetnji. Srećom, današnja razvojna okruženja imaju radne tokove koje je moguće prilagoditi za čim bolje izvođenje na određenoj platformi. Uz to, moguće je i postaviti skripte na neke komponente igre, čime bi se ubrzao proces izvođenja i poboljšalo sveukupno iskustvo mrežnog igranja.

Svaka igrana realistična igra treba imati integrirani sustav fizike koji će se brinuti o tome kako koji objekt u igri reagira na podražaje okoline, druge objekte ili na okolinu samu. Da se ne bi moralo kodirati za svaki objekt posebno, gdje će biti, kako će reagirati i što će se dogoditi nakon nekog događaja, sustav međuobjektne izračune izvodi samostalno. Tako će se na primjer, ovisno o namještenim postavkama za određen objekt, automatski izračunavati njegova brzina pomaka, udaljenost nakon sudara s drugim objektom ili vremenski utjecaji na njega. Sve se to u razvojnim okruženjima napraviti sa samo par linija koda, a igra dobiva realističnu notu.

Nezamislivo je pokrenuti igru bez da se prvo ne učita nekakav izbornik iz kojeg će se odabrati jedna od ponuđenih opcija igre. Uz to, potrebno je i nekakvo grafičko sučelje koje će korisnika obavještavati o bitnim informacijama za igru. Izrada gumba, padajućih izbornika, kliznih odabirača itd. Razvojna okruženja nude razne alate za jednostavniju izradu izbornika.

Jedan od ključnih dijelova svakog razvojnog okruženja za igre je izrada skripti. One se mogu koristiti kako bi definirali logiku same igre i mogu određivati način na koji se objekti unutar igre ponašaju. Unutar samog razvojnog okruženja mogu se pronaći unaprijed definirane stavke koje se već pri samom startu mogu primijeniti na različite komponente. Tako se, primjerice, kamera pomoću unaprijed pripremljenih skripti lako može postaviti kao primarna, a sustav animacija se može učitati na nekog lika i koristiti ih kao native.

## 4.2. Unity

Unity je smatran kao jedan od najboljih razvojnih okruženja jer je opremljen s velikim setom alata i mogućnosti koji su lako koristivi čak i nekom tko nije računalni mag. Korisničko sučelje jako prijateljski nudi svoje opcije korisniku koji relativno lako dolazi do željenih stvari. Jedna od najboljih značajki Unitya je integracija među velikim brojem različitih platformi. Tim svoj projekt na kojem radi može lagano premješati između čak 25 različitih platformi što je daleko više od dvojca s kojim ćemo ga uspoređivati [9]. Unity zajednica broji velik broj članova koji su voljni pomoći s projektom od samog početka.

Još jedna od funkcija koja čini Unity jednim od najboljih razvojnih okruženja je mogućnost korištenja velikog broja formata koji koriste vodeći 3D programi na tržištu. Direktno i bez potrebnog pretvaranja iz jednog formata u drugi korisnici će moći uvesti formate izrađene u 3D Maxu, Blenderu, CINEMI, Mayi, Zbrushu, SoftImageu i mnogim drugima. Nadalje, u startu razvojni tim ima pristup na više od 15 000 besplatnih i plaćenih modela, zvukova, animacija, dodataka, materijala, skripti koje može koristiti pri kreiranju svoje igre.

Unity koristi programski jezik *C#* ili *JavaScript*, koji je preferiran više od *C++* programskog jezika utoliko što je prilagodba s Jave na *C#* puno bezbolniji prebačaj. Sintaksa je vrlo slična. Uz to Unity ima brzo sučelje i nije toliko sistemski zahtjevan te se može pokretati čak i na Windows XP Service Pack 2 operativnom sustavu.

## 4.3. Unreal Engine 4

Sljedeći izbor je Unreal Engine 4, najnovije razvojno okruženje jedne od najvećih američkih softverskih tvrtki, Epic Games. UE4 je nasljednik Unreal Development Kita, poznatijeg i pod kraticom UDK u svijetu igara, na kojem su bile bazirane pretežito igre rađene za prošlu generaciju konzola i PC.

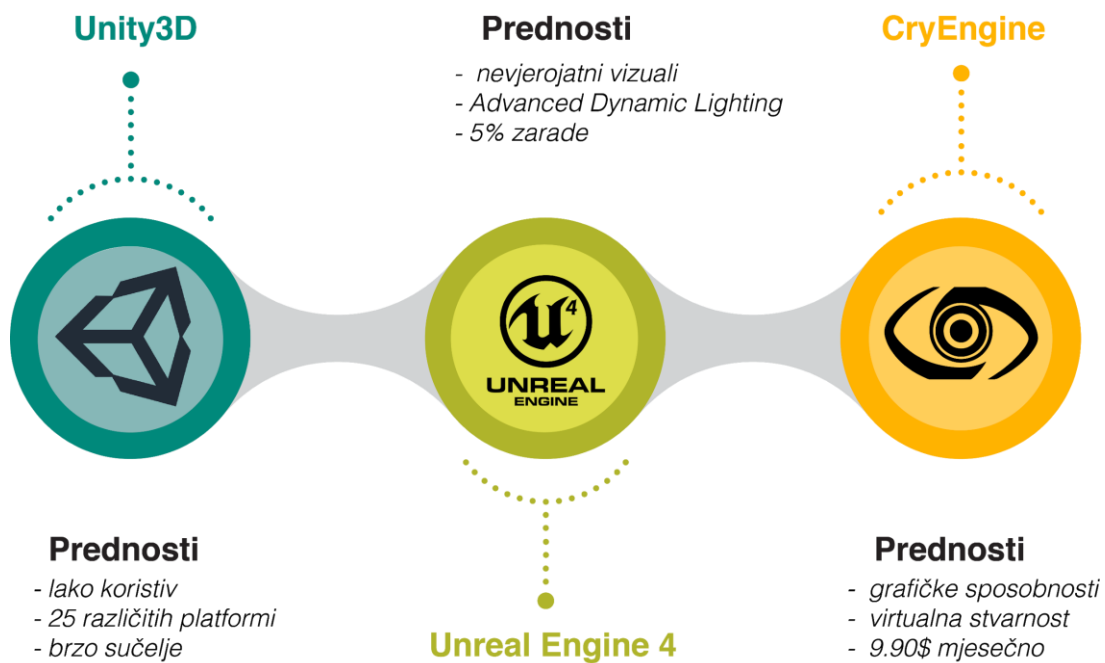
UE4 pruža alate za izradu zapanjujućih vizuala, koji iskustvu igranja daju realističnu notu. Sustav čestica je izrazito napredan te podržava simulaciju i do milijuna čestica u jednoj sceni. Jedna od inovativnih tehnologija koje koristi, "Blueprint Visual Scripting", omogućava izradu video igara korištenjem samo nacрта. Čak i netko tko nema veze s programiranjem, može do određene mjere izraditi igru, naravno s određenim ograničenjima, no to je samo još jedna opcija

koja govori o kako se moćnom razvojnom okruženju radi. Jasno je da je UE4 rađen s moćnijim hardverom na umu, pa je jedan od nedostataka to što ne podržava prošlu generaciju konzola, ali zato podržava razvijanje za 10 različitih platformi. UE4 besplatan je za korištenje, ali ukoliko u njemu izradite igru i zaradite ikakvu svotu novca, proizvođačima morate platiti naknadu u iznosu od 5% zarade [10]. Epic Games dobiva 5% od Vaše ukupne zarade, bile to kupovine unutar igre, mikrotransakcije ili konačan iznos igre. Tvorci Unreal Engine-a 4 dopuštaju korištenje pune verzije svog razvojnog okruženja bez naknade timovima čiji proizvod ne zarađuje više od 3000\$ po godišnjem kvartalu.

#### **4.4. CryEngine**

U studenom 2007. godine razvojni tim Crytek izdao je igru Crysis ekskluzivno za PC. Tada se većina konfiguracija mučila za pokretanje istog na srednjim grafičkim postavkama, a za visoke grafičke postavke bila je potrebna neuobičajeno jaka konfiguracija. Tada je postao svojevrsni benchmark za PC i samo potvrdio reputaciju jednog od najjačih razvojnih okruženja s neprikosnovanim vizualima, koje je nosio još od 2004. kada je izdana igra Far Cry od izvođača Ubisoft. Nekoć je bilo razvojno okruženje namijenjeno isključivo PC-u, no dolaskom konzola treće generacije (PS3, Xbox360) i njihovom velikom tržištu, Crytek je popustio i modificirao svoje razvojno okruženje za korištenje na konzolama. Danas je razvojno okruženje grafičkih sposobnosti većih no Unity te vrlo slično s mogućnostima UE4. S moćnijim hardverom u vidu, CryEngine je moćno razvojno okruženje koje je nešto zahtjevnije za naučiti, s čime je i teže iskorištavanje njegovog punog potencijala. To ga ne čini pogodnim za početnike.

CryEngine razvojno okruženje podržava izradu sofisticiranih sadržaja virtualne stvarnosti, uključujući sistem generiranja fine magle i oblaka koji omogućuju realističnu vizualizaciju vremenskih uvjeta. Jedna od boljih strana CryEnginea u odabiru između raznih razvojnih okruženja je njegova cijena, a to je 9.90\$ mjesečno za korištenje [11]. Uz to CryEngine ima i posvećeni forum s pitanjima i odgovorima pod nazivom CryEngine Answers, kojeg čini velika zajednica programera i svakodnevno se ažurira.

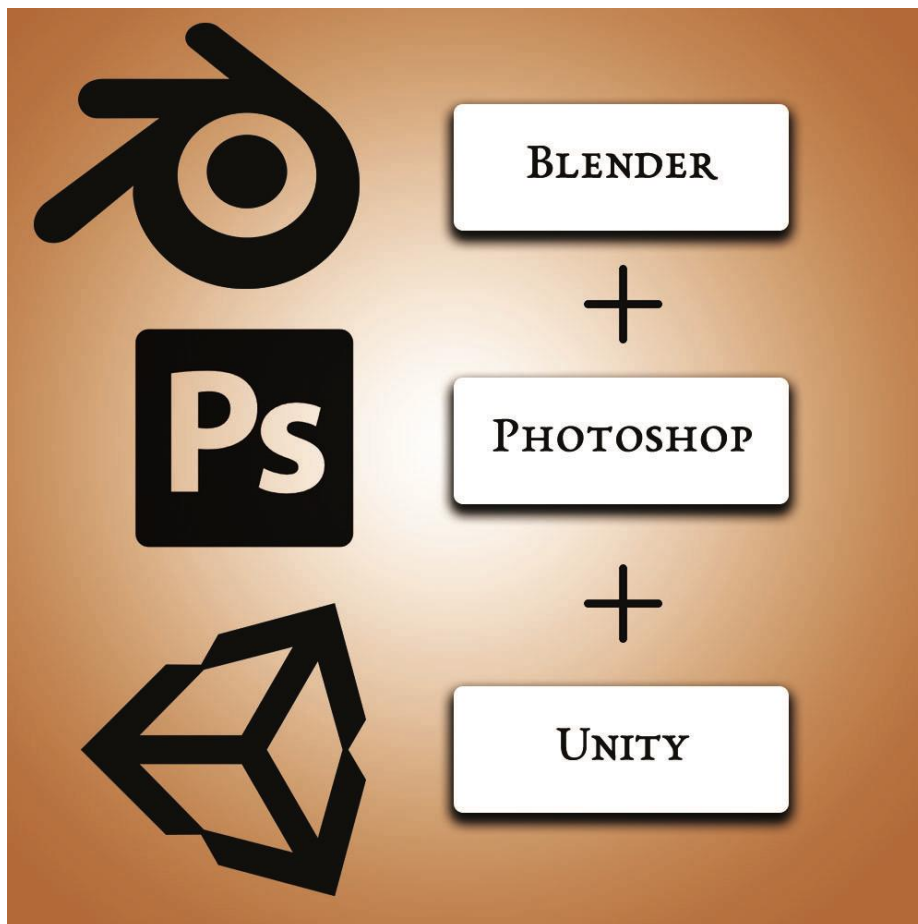


Slika 4. Razlozi zašto Unity

Zbog jednostavnosti, funkcionalnosti i svestranosti, u ovom radu će se koristiti Unity razvojno okruženje, koje je besplatno za korištenje sve dok projekt koji je njime kreiran donosi prihode manje od 100 000\$ godišnje.

## 5. IZABRANI SOFTVER

Temeljitim istraživanjem odlučeno je da će ovaj rad biti izrađen u kombinaciji programa Blender, Adobe Photoshop te Unity. Iz prethodnog istraživanja došlo se do zaključka da se na više načina može odraditi isti posao, te da nema ispravnog niti krivog načina za napraviti 3D okruženje. Pitanje je samo što se želi postići i kako je jednostavnije i pristupačnije.



Slika 5. Odabrani softver



## 6. PRAKTIČNI DIO IZRADE 3D OKRUŽENJA- BLENDER

### 6.1. Skiciranje

Za izradu dobrog okruženja potrebno je unaprijed isplanirati kako će se isti izraditi. Dobra praksa je prije svega napraviti nacrt, nekakvu grubu skicu, da bi se željena ideja što bolje provela u djelo. Tako će se postupiti u ovom radu. Ideja koja se planira provesti u djelo u ovome radu je oblikovati predivan krajolik kojim će se biti moguće kretati, spoj šume, mora, planina i otoka u jednom.



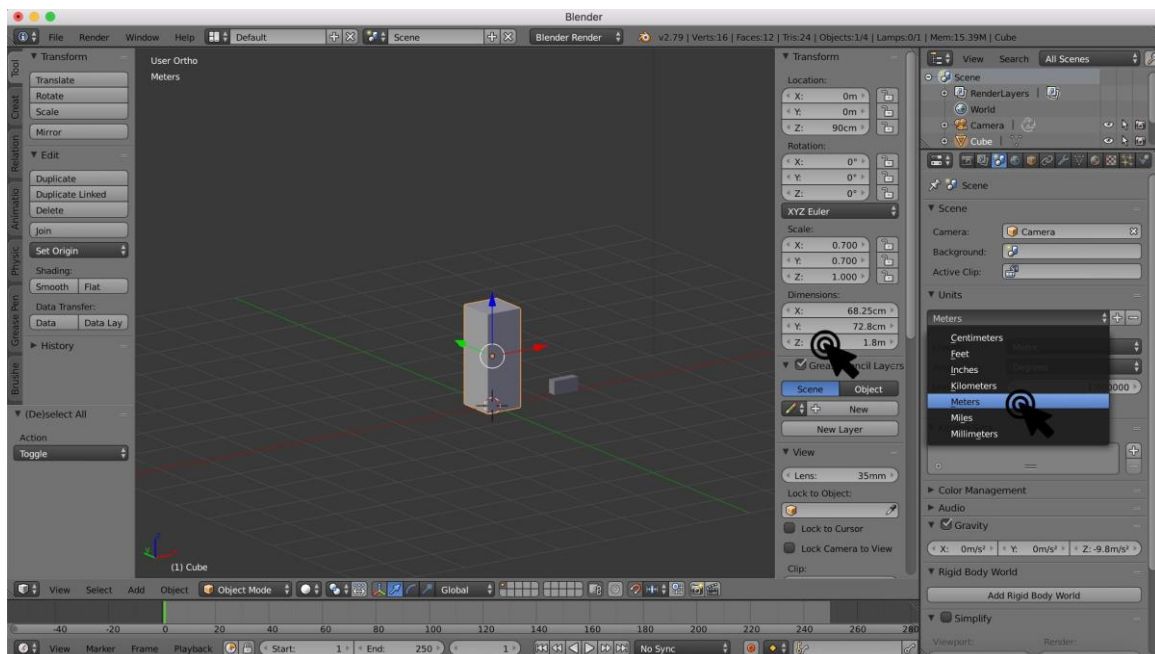
*Slika 6. Skica terena*

Na skici je šturo nacrtan krajolik. Skica ne mora izgledati vizualno spektakularno ali mora predstavljati nešto što će crtaču iste značiti. Dobro je imati nekakvu okvirnu ideju koja se želi postići. Ideja prenesena na papir se u konačnici lakše vizualizira. Po njoj se konstruira, modelira, usavršava i doraduje. Ista ta skica drugima ne mora ništa značiti, drugi ne moraju u njoj vidjeti isto što vidite i Vi, ali bitno je da Vama pomaže. Uz skicu se dalje mogu planirati sljedeći koraci, poboljšavati je usputno, dodavati željene detalje itd. Sljedeći korak u ovom radu je pokretanje Blendera i početak modeliranja.

## 6.2. Modeliranje u Blenderu

U ovom koraku će se površno opisati izrada modela koji će se koristiti u završnoj sceni okruženja. Dobar bi početak bio da se prvo napravi teren, te da se na njega stavi referentni objekt poznatih dimenzija. U ovom radu će se postaviti kvadrat dimenzija prosječnog čovjeka. Objekt dimenzija prosječnog čovjeka je dobar za usporedbu jer je svima vrlo poznat. Većina se, svakodnevno, suočava s puno stvari za koje je poznat omjer prema prosječnom čovjeku. To daje temelje za dobar omjer veličina svih objekata na sceni.

Standardno je u Blenderu postavljen mjerni sustav Blender jedinica, koji je potrebno promijeniti u metričke za lakše snalaženje pri uspoređivanju veličina.



Slika 7. Postavljanje odnosa veličina

### 6.2.1. Odnos veličina

Na *Slici 7.* je prikazano kako postaviti mjerni sustav u Blenderu u metrički. Na slici je isto tako prikazano da je stvoreni objekt promijenio visinu na visinu prosječnog čovjeka (1.8m) te se to vidi i na alatnoj traci s desne strane. Visina je u ovom slučaju os Z u Blenderu, tako je standardno postavljeno prilikom pokretanja i takav je sustav u kojem Blender funkcioniра. Os X označava širinu, a os Y dubinu [12]. Kombinacijom vrijednosti te tri osi dobije se mjesto u trodimenzionalnom prostoru. Za lakše razumijevanje oko odnosa veličina, u scenu je postavljen i objekt koji predstavlja mačku. Kratkom pretragom na internetu dobije se informacija da je prosječna mačka visoka između 23 i 25 cm. Zatim je stvoren objekt koji dimenzijama odgovara mački i postavljen kraj objekta koji predstavlja čovjeka. Tako se dobije uvid koliko je objekt zapravo velik i tim načinom se lakše može odrediti veličina sljedećeg objekta kojeg se izrađuje.

### 6.2.2. Vizualan stil

Prije nego se odluči kako će teren biti izrađen, mora se izabrati vizualan stil koji će se koristiti. Postoji velik broj tipova videoigara, 2D ili 3D te uz to igra može biti izvedena iz prvog lica, trećeg lica, pogleda s visina itd. Sve su to presudni faktori kada se odlučuje kako će igra izgledati te kako će se najbolje prenijeti ideja koja je zamišljena na igrača.

### 6.2.3. Low-poly Tehnika Modeliranja

Cilj ovog rada je približiti postupak izrade modela, elemenata, te u konačnici sveukupnog trodimenzionalnog okruženja svakom tko je za to zainteresiran. U nastavku će biti opisan postupak izrade 3D okruženja s modelima malog broja poligona (*eng. Low poly technique*). Poligoni su geometrijska tijela omeđena trima ili više strana. Prilikom modeliranja poželjno je raditi s četverokutima jer je pomoću njih lakše dobiti željeni oblik. Međutim, cijela geometrija modela se pretvara u trokute prilikom unošenja u bilo koje razvojno okruženje za igre. Razlog tomu je taj što su trokuti najjednostavnije geometrijsko tijelo i najlakši su za proračun pozicije i veličine od ostalih geometrijskih tijela. Tehnika izrađivanja *low-poly* modela je koncipirana tako da se uz što manje generiranih točki oblikuje model koji bi to trebao predstavljati. Ideja je napraviti što zanimljiviji oblik objekta koji se želi koristiti uz što manje geometrije. Ovakav princip se i preporučuje početnicima u

svijetu 3D modeliranja jer se tako na zabavan način upoznaju s osnovama programa, njegovim principima, načinu korištenja itd [13]. Pozitivna strana je i to što se na taj način potiče njihova kreativnost i razvija novi način razmišljanja.

Postavljeni su temelji za izradu 3D okruženja pomoću *low-poly* modela. Za nastavak izrade low poly okoline dobro je prisjetiti se ranije skicirane skice. Ideja je da se napravi što sličnije okruženje onome na skici, no kako se skica bude provodila u djelo tada normalno je da dođe do blagih izmjena, poboljšanja i minorne promjene kompozicije. Sada se može krenuti u izradu, počevši od terena ili osnovnih modela. Na stvaraocu je samom da odluči od kuda kreće. U ovome radu će se prvo krenuti od manjih modela koji će činiti scenu, pa će se kasnije izraditi teren, a eventualne izmjene će se raditi usputno.

Na skici korištenoj u ovom radu može se vidjeti da velik dio okruženja čini šuma, pa će se prvo izrađivati šumski elementi. Izmodelirati će se stablo koje će biti jedno od mnogo objekata koji će izgrađivati šumu. Cilj je napraviti što jednostavnije objekte pa će se tako koristiti i jednostavni oblici pri izradi. U većini slučajeva u modeliranju se od bilo kojeg modela može stvoriti željeni oblik, samo je bolje početi sa što sličnijim početnim modelom jer će se brže doći do rezultata. Ne treba nepotrebno gubiti vrijeme kada se logičkim odabirima može uvelike olakšati stvar od početka. Deblo stabla biti će izrađeno od okomito postavljenog cilindra, a krošnja će biti izrađena od kombinacije više jednostavnih oblika. Uz to, uvijek treba paziti da ukupan broj točaka koji kroje geometriju stabla ne bude velik. Tako nema potrebe pretjerivati s izradom niti težiti nekakvim fotorealizmom.



*Slika 8. Low-Poly Stabalce*

Izmodelirani objekt daleko je od realističnog, ali to nije ni bitno jer je ovo vizualan stil kojem smo težili. Model drvca je ukrašen jednostavnim materijalima, jednostavne smeđe boje za deblo i tamno zelene krošnje. Materijali koji se vide na slici su napravljeni u Blenderu te su usko vezani uz sustav na koji Blender funkcionira.

#### 6.2.4. Sustavi generiranja slike u Blenderu

Ovo je drvce izrađeno u Blenderovom integriranom sustavu generiranja slike. Do 2013. godine Blender je koristio interni sustav (*Blender Internal*) za generiranje slike pomoću kojeg je bilo teško dobiti kvalitetne vizuale i pogoditi točne interakcije svjetlosti s fizičkim objektima. Od 2013. u Blender je implementiran *Cycles*, sustav za generiranje slike koji simulira gibanje pravih zraka svjetlosti. Nakon što zraka svjetlosti dođe do nekog objekta, ona se tu ne zaustavlja. Rasprši se, odbija i mijenja smjer te u konačnici daje rezultate sličnima što bi dobili u stvarnosti. U sljedećoj nadogradnji brojčane oznake 2.80, doći će do velikih napredaka vezanih uz generiranje slike. Uz dosadašnji izbor sustava za izradu slike Blender će imati u ponudi i *Eevee*, sustav koji donosi proračune u stvarnom vremenu. Na novijim konfiguracijama s jačim grafičkim karticama izračun slike će se događati gotovo instantno. *Eevee* će donijeti napredak najviše u animacijama u kojima se ne cilja na fotorealističnost, jer ipak za sada, *Cycles* kao sustav generiranja slike dominira među navedenima po kvaliteti izračuna.

Slika ovog drvca je generirana u *Cycles* sustavu za generiranje slike. Ovo drvce ima krošnju i deblo drugačije boje. Korištena su ukupno 2 materijala za njegov prikaz u *Cycles* sustavu. Materijali iz *Cyclesa*, zbog drugačijeg načina generiranja slike nisu kompatibilni s *Blender Internal* sustavom, ali će biti iskoristivi u *Eevee-u*. *Eevee* je trenutno dostupan samo u beta verziji Blendera koja trenutačno nije stabilna, pa se za nastavak rada odlučilo za korištenje aktualne verzije Blendera brojčane oznake 2.79. Materijali između različitih sustava generiranja slike u Blenderu međusobno nisu kompatibilni, pa ih stoga neće biti moguće ni iskoristiti u ciljanom razvojnom okruženju, Unityu [14]. Izrađeno je drvce od malog broja poligona, na njega su primijenjeni materijali i izgenerirana je slika samog drvca da bi se vidjelo kako će drvce izgledati i kasnije kada ga se unese u Unity.

### 6.2.5. Izrada modela za korištenje u sceni

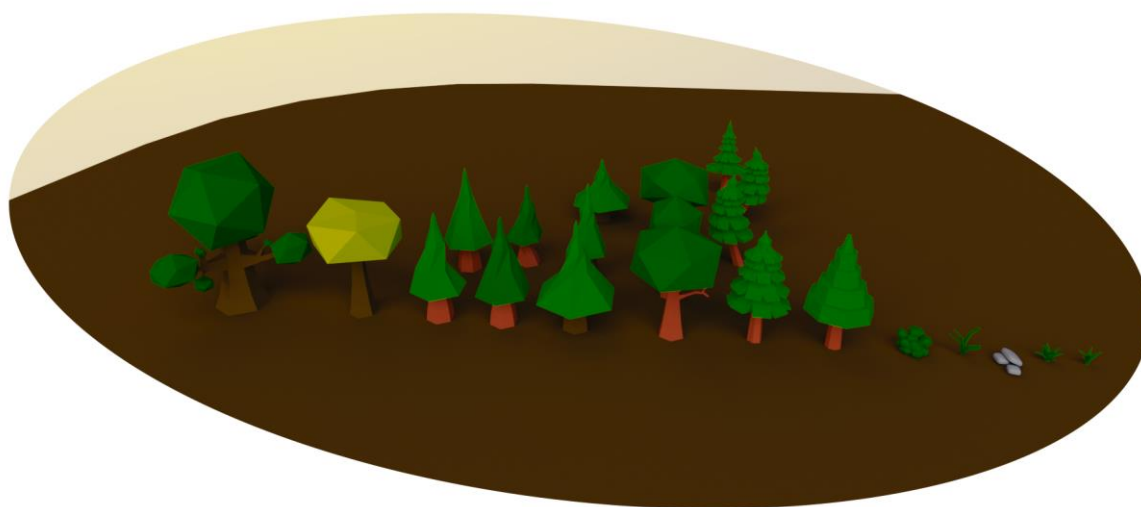
Nakon izrade jednog modela drva koji će se koristiti, po istoj šabloni će biti izrađen i ostatak modela potrebnih za kreiranje scene. Kreće se od sličnih objekata, od drugačijih stabala. Za deblo se postavlja low-poly cilindar okomit na površinu, a za krošnju se postavljaju neki od jednostavnijih geometrijskih oblika. Jedan od jednostavnijih objekata u Blenderu je Ico sfera, kugla sačinjena od trokuta. U Blenderu se može generirati i UV sfera, koja je pogodnija za teksturiranje, ali to u ovom primjeru nije potrebno. Za vizualan stil kojeg se odlučilo pridržavati puno je pogodnija Ico sfera. Osim stabala, za izradu zanimljivog terena još će se koristiti travke, grmići i kamenčići. Kada se svi objekti izrade, poslože se jedni pored drugih te ih se po potrebi može uzimati i koristiti u izradi scene.



*Slika 9. Modeli koji će se koristiti za izradu šume*

Na *Slici 9.* se vidi kako su svi izrađeni modeli po vizualnom stilu slični. Oni će kreirati konačnu šumu sa skice. Velik dio terena će prekrivati šuma, a broj trenutno izrađenih modela koji će tvoriti šumu je premalen. Trenutno će biti očit uzorak ponavljanja pojedinih stabalaca te će sve skupa izgledati repetitivno.

Prilikom izrade jednog modela zato se izrađuju i malo modificirani modeli, čiji proces izrade neće oduzimati previše vremena, a u konačnici će scenu učiniti nasumičnijom i zanimljivijom. Prilikom izrade jednog stabalca, duplicirat će se postojeće, te će se to novo stabalce malo izmijeniti, kako bi se izbjegao dojam repetitivnosti.



*Slika 10. Malo izmijenjeni modeli stabala*

Na *Slici 10.* vidi se da je dovoljno primijeniti malene preinake na jednom modelu da bi se postigao potpuno novi izgled na drugom.

Sljedeći objekt koji će se modelirati je kuća. Pogledom na skicu može se vidjeti da osim šumovitog i planinskog terena te mora, najveći centar zanimljivosti je kuća.

Za početak izrade koristi se obična kocka, koja se postavi na sredinu scene da je lakše modelirati. Zatim se doda još jedna kocka kojoj se na jednoj strani dva brida spoje kako bi tvorili trokut. Nakon toga se pozicioniraju oba objekta tako da trokut tvori krov, a kocka trup kuće. Trokut koji je napravljen od kocke se podijeli na 4 jednaka dijela uz njegovu dužinu, te na dva jednaka dijela u visinu. Nakon toga se od samo jednog od ta 4 segmenta modeliranjem stvori dimnjak.



Još preostaje geometriju oba dijela, krova i trupa kuće podijeliti na manje dijelove te malo izmijeniti kako bi kuća pratila vizualni stil ostatka okoline. Kada je sve izmodelirano, potrebno je oba dijela spojiti naredbom «CTRL+J» pod nazivom Join.



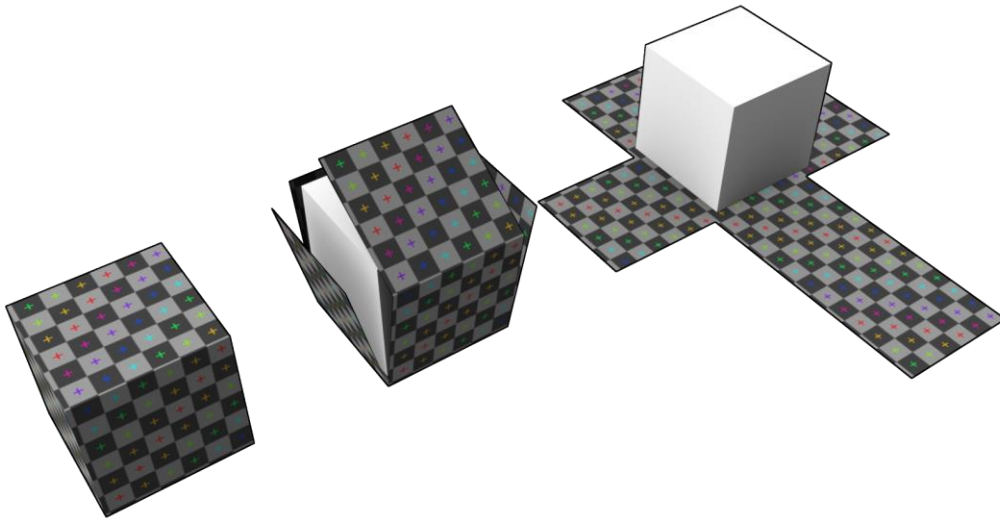
*Slika 11. Gotov model kućice*

Na *Slici 11.* se vidi da izrađeni model prati vizualni stil ostatka modela koji će krojiti konačnu scenu te pritom ispunjava uvjet malenog broja točaka koji čine njegovu geometriju. U ovom slučaju se neće ići istim principom dodavanja materijala modelima kao prije. U ovom trenutku je poželjno napraviti materijal za kuću tako da izgleda identično i kada se model uveze u Unity. To se radi tako da se na model kuće stavi tekstura.

## 6.3. Izrada teksture

### 6.3.1. «Odmotavanje» modela

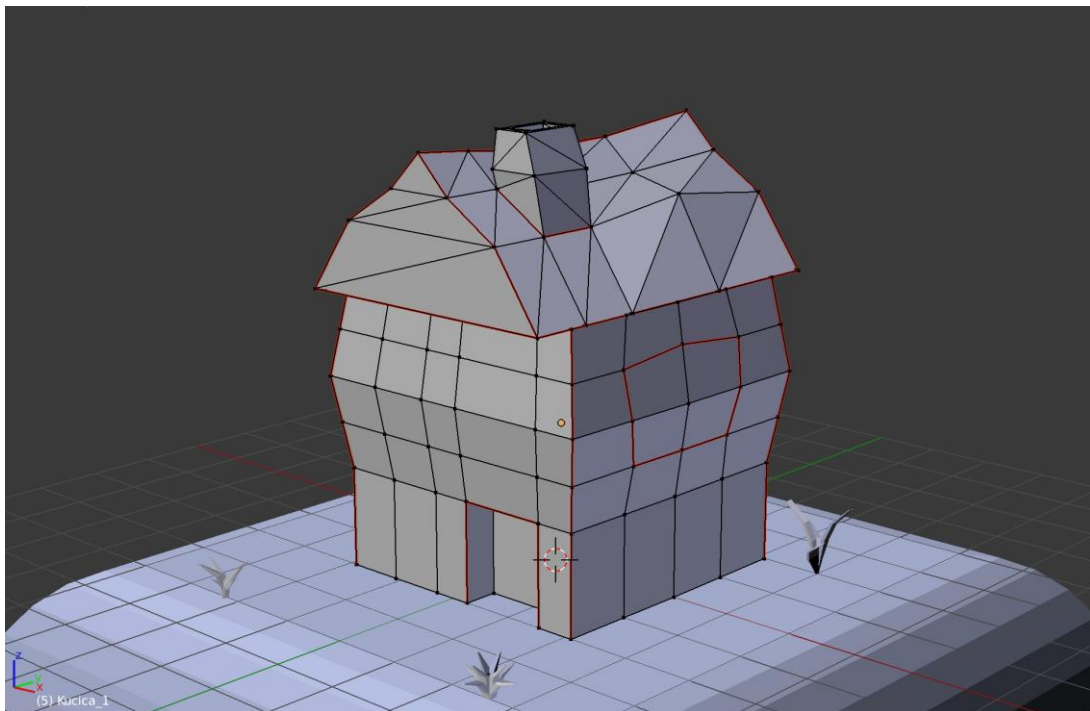
Najsigurniji način da model izgleda identično u programu za 3D modeliranje i u razvojnom okruženju u kojem će se izrađivati igra je taj da se na modelu primjeni tekstura. Tekstura će obojati određena područja modela u jedno, a druga područja u drugo. Da bi se to uspješno izvelo prvo treba «odmotati» 3D model, tj. prikazati kako bi taj trodimenzionalni objekt izgledao da je dvodimenzionalan. Da se za primjer uzme kocka od papira, zadatak bi bio odmotati ju da izgleda kao dvodimenzionalni papir.



Slika 12. "Odmotavanje" 3D modela  
(preuzeto: [commons.wikimedia.org](https://commons.wikimedia.org))

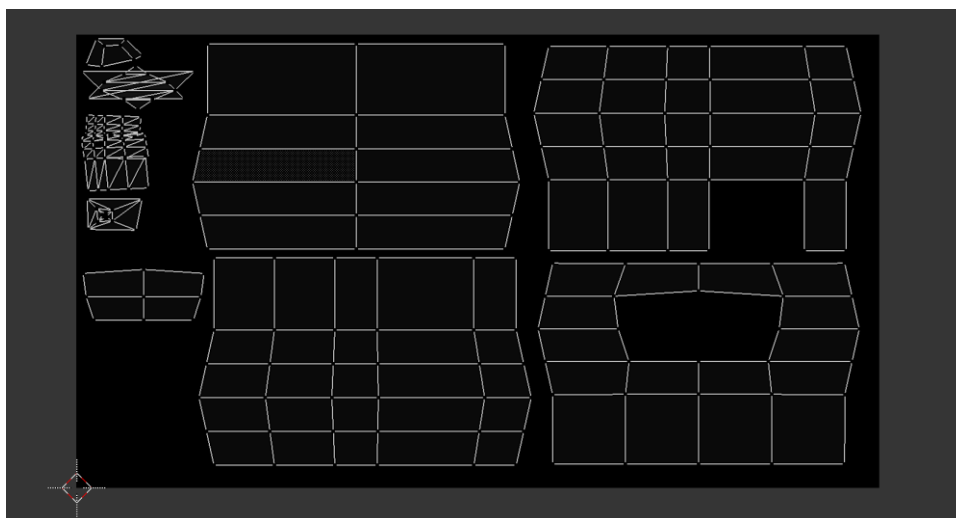
Vrlo brzo dolazi se do zaključka da se od papirnate kocke može dobiti dvodimenzionalni list jedino tako da se na nekim mjestima zareže. Upravo tako se «odmotava» trodimenzionalni model. Na mjestima na kojima bi se u realnosti neki objekt trebao zasjeci da bi se cijeli mogao odmotati a da se pritom ne raspadne, tako i modeler treba «zasjeci» model u virtualnom svijetu da bi se mogao što bolje odmotati. U ovom slučaju se ne koristi nož, već «*Mark Seam*» opcija u Blenderu. Ona označava koji bridovi će biti zarežani. Ima više vrsta odmotavanja modela, ali

za najprofesionalnije rezultate koristi se opcija «*UV Unwrap*». UV u tom nazivu predstavlja dvije osi, kao os X - širinu i os Z - visinu u trodimenzionalnom sustavu.



*Slika 13. Naznačivanje "rezova" na modelu*

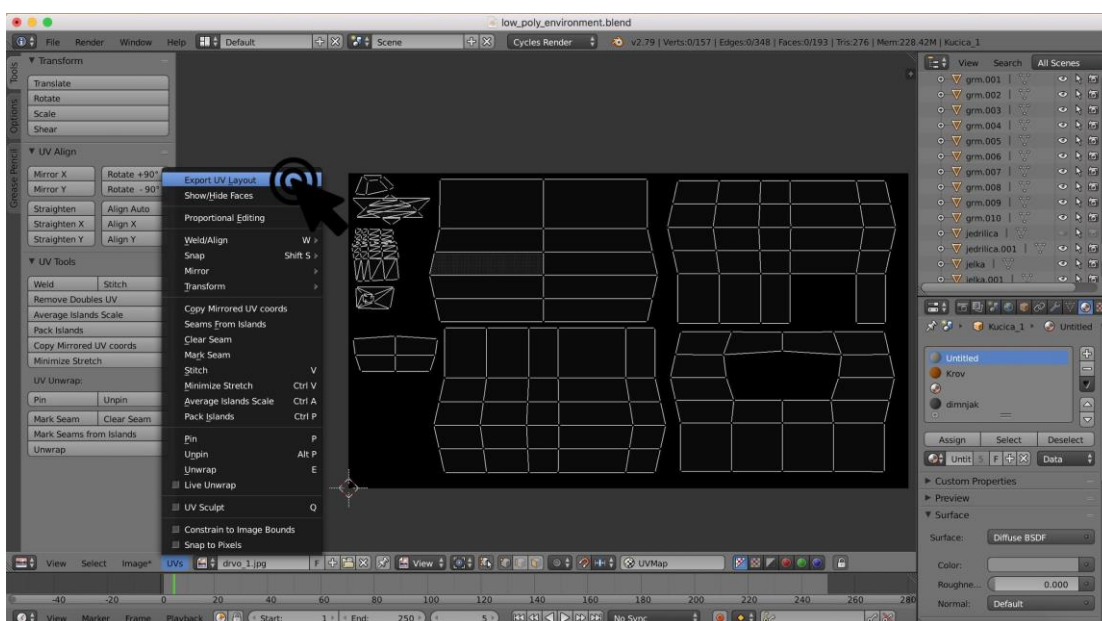
Na *Slici 13.* se može vidjeti da je model zarezan na logičkim mjestima. Na mjestima gdje se model savija pod kutem od  $90^\circ$  i na mjestima gdje dolazi do promjene materijala. Zatim se model odmotava naredbom «*UV Unwrap*» te se dobije cijeli trodimenzionalan model na jednoj, dvodimenzionalnoj plohi (*slika 14.*).



*Slika 14. «Odmotana kućica»*

### 6.3.2. Izrada teksture za 3D modele u Photoshopu

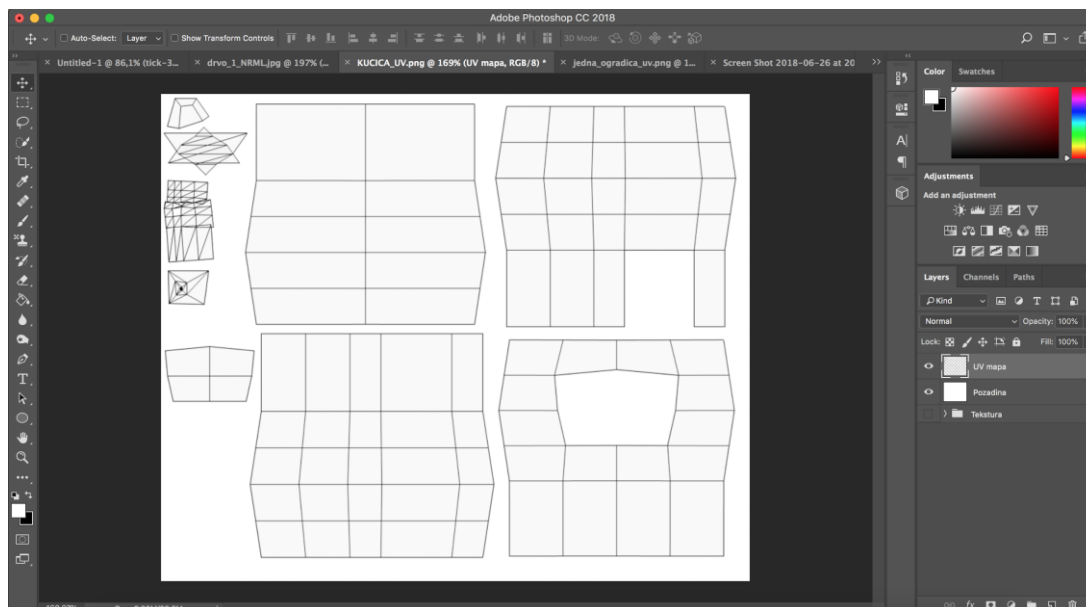
Iako Blender ima ugrađeni dio za obradu fotografije u svom paketu alata, koji bi u nekim slučajevima bio dostatan, uporaba posebnog programa za obradu fotografije daje puno više mogućnosti. Unutar samog Blendera se može pod opcijom «UV/Image Editor» manipulirati UV kartom odmotanog modela, no zbog navedenih razloga nakon što je model odmotan nastavak izrade se seli u drugi program, Adobe Photoshop. Prije odlaska u Adobe Photoshop, u Blenderu se u «UV/Image Editoru» odabire izvoz UV karte željenog objekta (Slika 15.).



Slika 15. Izvoz UV mape iz Blendera

U prozoru koji se prikazuje nakon odabira izvoza UV karte, odabire se tip datoteke (najčešće .jpeg ili .png) pod kojim se želi spremiti UV karta. Također, može se odabrati veličina slike u pikselima te vidljivost UV karte. Optimalna vidljivost UV karte je 25% te će se pod tim postavkama izvesti iz Blendera i uvesti u Adobe Photoshop. Svi do ovog trenutka napravljeni koraci omogućuju digitalnu obradu UV karte 3D modela kućice, te kreiranje teksture specifično krojene za nju.

Nakon otvaranja UV karte u programu Adobe Photoshop dobiva se potpuna kontrola nad izradom teksture koja će krasiti model *kućice*.

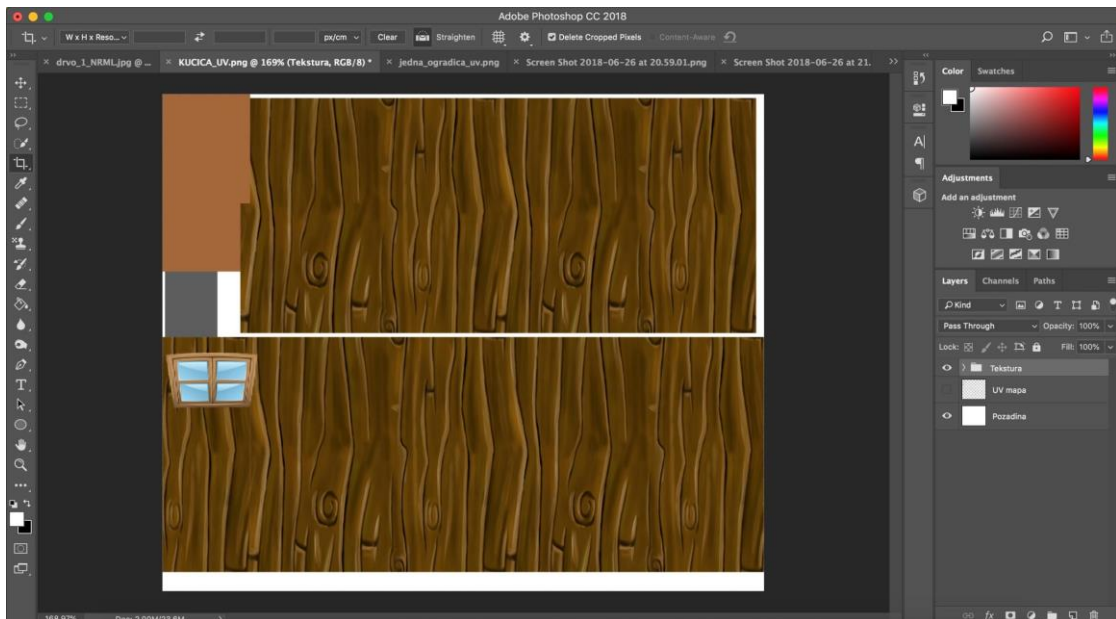


*Slika 16. UV mapa modela u Photoshopu*

Na *Slici 16.* vidljive su sve strane kućice te se nameću beskrajne mogućnosti kako tu istu kućicu teksturirati. Treba imati na umu dosadašnji izbor modela i materijala koji izgrađuju scenu. Kako do ovog trenutka u radu pratio stilistički lijep i nježan stil, suprotan od realističnog, u tom stilu će se teksturirati i ova kućica. Tekstura koju će ova kućica koristiti može se nacrtati ili se može koristiti kakva fotografija. Izbor je vrlo širok, a može se kombinirati postojeća tekstura s neakvim vlastitim elementima. Dobro je iskoristiti već postojeće slike, ukoliko su dostupne i odgovaraju zamisli i željenom vizualnom stilu. Time se štedi vrijeme i radni tok je brži, jednostavniji i produktivniji. Na internetu su dostupni razni kvalitetni izvori tekstura za uporabu u svojim kreacijama, od kojih se neki plaćaju a neki od njih su i besplatni.

Za zidove kućice u ovom radu odabrana je tekstura rukom nacrtanog drva, za prozor je odabrana tekstura rukom nacrtanog prozora dok su za krov i dimnjak odabrane obične boje. Za bolji vizualni dojam teksturu drva potrebno je napraviti neponavljajućom, tj. potrebno je postići da se nakon njenog dupliciranja ne vidi očiti uzorak ponavljanja. Od obične ponavljajuće teksture to se vrlo lako može

napraviti pomoću ugrađenog filtera u Photoshopu pod nazivom «Offset». Slika se u kombinaciji tog filtera sa postojećim alatima može jednostavno modificirati i napraviti neponavljajućom. Ovdje dolazi do izražaja prethodno namještena vidljivost UV karte od 25% te se po njoj mogu zalijepiti teksture i može se obložiti točno kako smo zamislili.



Slika 17. Konačna tekstura kućice

Na Slici 17. je prikazana konačna tekstura koju će model kućice imati. Nakon što su sve stvari posložene na svoje mjesto, *UV karta* koja je služila za precizno postavljanje bez posljedica se može ukloniti. Tekstura se zatim sprema pod tipom datoteke (najčešće *.jpeg* ili *.png*) i spremna je za korištenje na modelu. Dobra praksa je da se teksture izrađuju u kvadratnom formatu te u veličinama potencije broja 2. Visina i širina teksture tako iznosi 512px x 512px, 1024px x 1024px itd. Takav način izrade tekstura se primjenjuje u igračkoj industriji još od njezinih začetaka. Grafička jedinica za obradu podataka (*GPU*) najbrže izračunava i učitava teksture upravo takvog formata. Ako se tekstura izradi u formatu koji nije kvadratnog oblika potencije broja 2, to ne znači da se tekstura neće učitati. To samo znači da će *GPU* trošiti više resursa pri njezinom učitavanju i da će učitavanje duže trajati. Tako se nepotrebno opterećuje platforma na kojoj se igra izvodi što za posljedicu ima nepoželjnu neoptimiziranost. [15]



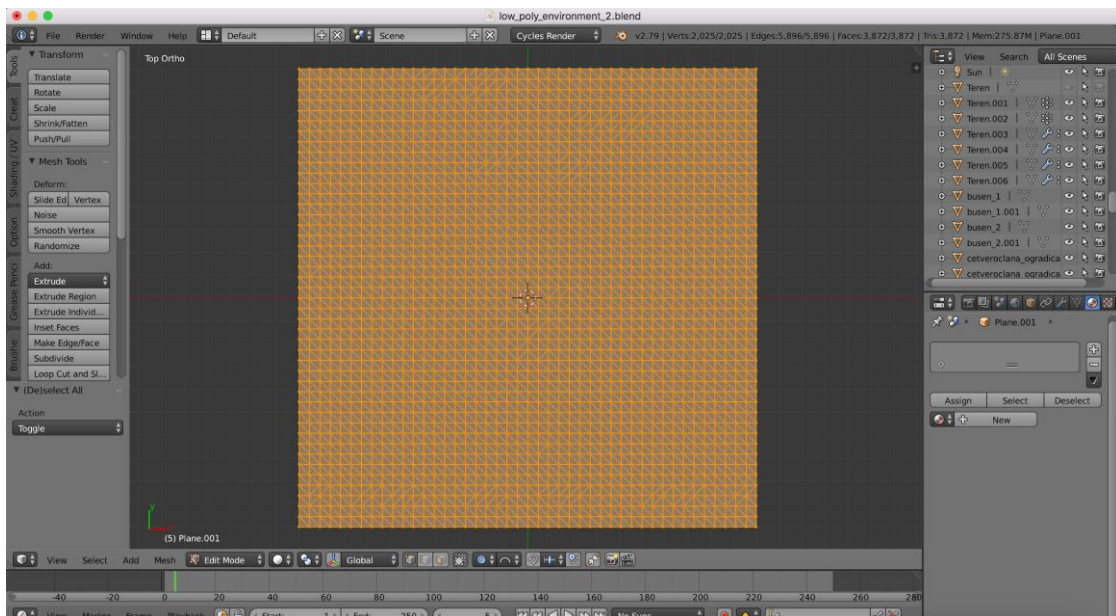
*Slika 18. Konačni model kućice s teksturom*

Na *Slici 18.* je prikazan 3D model kućice s nalijepljenom teksturom na sebi. Može se primijetiti kako nema očitih ponavljanja teksture drva, te kako je na svakom području na kojem je prethodno bilo zamišljeno sada odgovarajuća tekstura. U konačnici se dobije 3D model vrlo lijepog vizualnog stila i lake geometrije koji će biti zanimljiv detalj u završnoj sceni.

## 6.4. Izrada terena

Nakon izrade objekata koji će činiti scenu, potrebno je izraditi teren. Pod terenom se smatra onaj objekt po kojem će se igrač kretati, na kojem će se svi drugi objekti nalaziti, krajolik čiji će se istraživati i razgledavati. Teren će također biti izrađen u Blenderu. Unity razvojno okruženje ima integriran alat za izradu terena, ali u ovom radu će za njegovu izradu biti odabran Blender zbog zanimljivog vizualnog stila kojeg se postiglo dosadašnjim načinom izrade.

Unutar Blendera, izradi se obična ploha i pozicionira se u sredinu scene radi lakšeg manipuliranja. Ploha se zatim poveća nekoliko puta ne bi li se postiglo veliko polje u odnosu na veličinu igrača koji će se kretati njime. Nakon toga se ploha podijeli (*Subdivide*) desetak puta na jednake malene dijelove u svrhu lakšeg manipuliranja njima i lakše izrade krajolika. Tako se podijeljena ploha na manje pravokutnike u konačnici pretvara u manje trokute, najjednostavnije poligone.



Slika 19. Podijeljena ploha na malene poligone

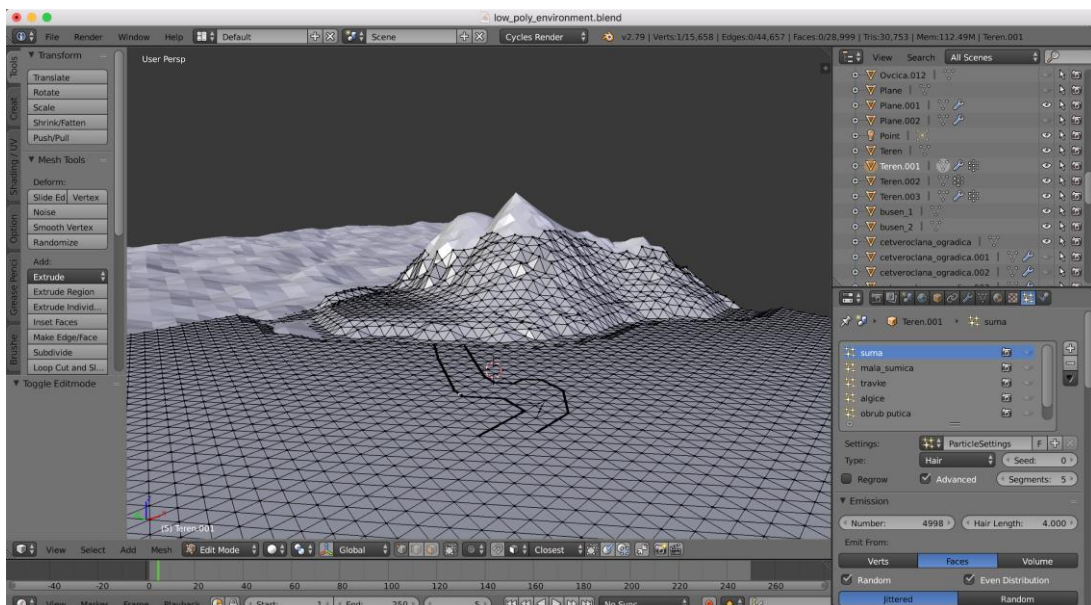
Trokuti su najjednostavniji poligoni koji su poželjni u ovakvoj izradi terena zbog jednostavnijeg prikaza gradacije geometrije u odnosu na kvadrate.



Nakon toga se modelira krajolik na kojem će se objekti nalaziti. Na početnoj skici se vidi da krajolikom dominiraju planine, stoga se prvo modelira osnovni dio terena na kojem će se lako nadograditi veće uzvisine. Blenderov integrirani alat za proporcionalno izmjenjivanje geometrije pod nazivom «*Proportional Edit*», koji se aktivira pritiskom tipke *O* na tipkovnici po standardnom sučelju, dodaje uzvisine, te kako se postupno modelira, konačna scena se sve više definira.

Modeliranje je postupak koji nema jasno definiran kraj te se na zasebnom objektu može provesti iznimna količina vremena kako bi se dobilo ono što se prvotno zamislilo. Nekad je pametnije završiti izradu modela nauštrb osobnog zadovoljstva te prijeći na drugi objekt jer se tako štedi velika količina vremena, a to se vrijeme može kvalitetnije iskoristiti na izradu drugih detalja koji će na kraju učiniti scenu zanimljivijom. Za izradu 80 – 85% nekog modela potrebno je pola od ukupnog vremena izrade, dok je druga polovina rezervirana za ostatak. Zato je potrebno dobro definirati što je u sceni bitno, koji će to dio plijeniti pažnju te koliko a što je manje bitno. Na ništa što neće biti vidljivo ne treba trošiti vrijeme. Kako nema nepotrebne geometrije, tako je broj poligona manji i konačna scena je lakša ciljanoj platformi za izračunavanje i izvođenje.

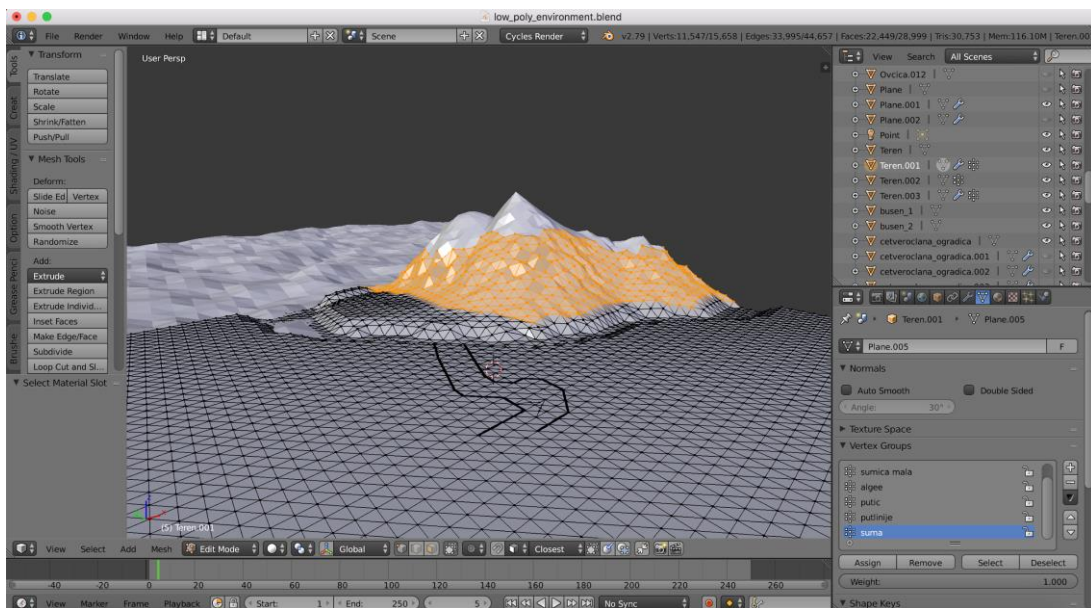
Česta praksa kod izrade lika koji se u igri kontrolira je ta da se na mjestima gdje mu je odjeća, geometriju koja je tvorila tijelo jednostavno izbriše. Tako se modeli bolje optimiziraju i igra je lakša. Isti postupak se primjenjuje na nekim modelima koji imaju unutrašnjost. Kuće, zgrade, nastambe, stanovi ili bilo kakve građevine se rješavaju modeliranja interijera ako isti taj interijer nije potreban u cijeloj priči.



Slika 20. Izmodelirani krajolik

Na *Slici 20.* se može vidjeti ploha izrađena od trokutastih poligona, izmodelirana da tvori krajolik. Planine se jednostavno uzdignu koristeći alat za proporcionalno izmjenjivanje geometrije, a na isti način se teren i udubi.

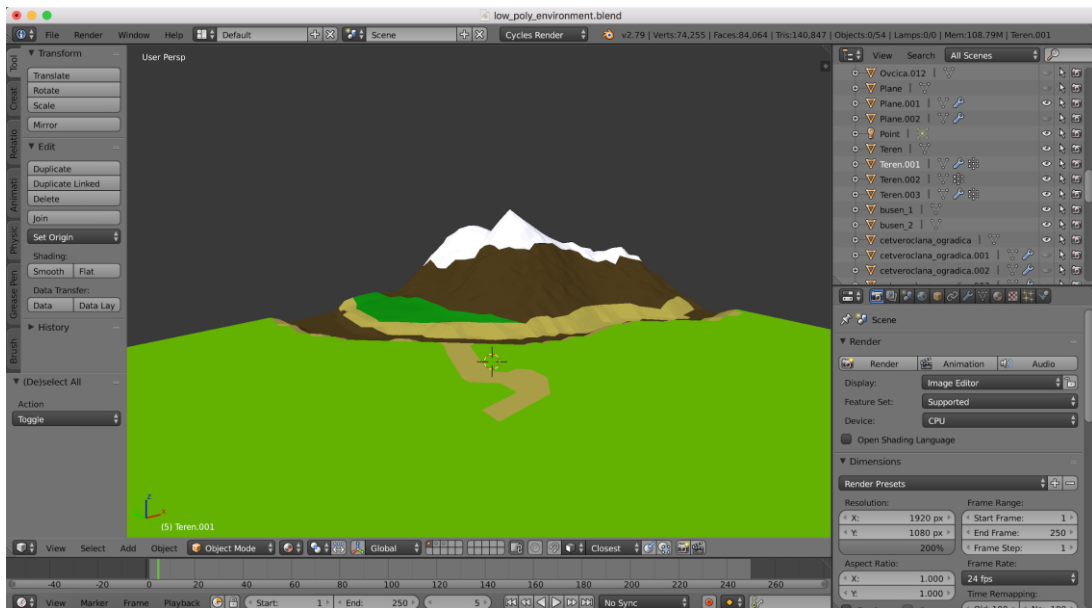
Nakon modeliranja krajolika, na njega se nanosi materijal. Koristit će se paleta boja koje prate dosadašnji stil modela. Boje će biti živopisne, jednostavne i oku ugodne. Najbolji odabir za primjenu više materijala na jedan objekt je *“odmotavanje 3D modela”*. Zbog prikaza što većeg opsega tehnika u ovom radu se neće koristiti tehnika odmotavanja. Umjesto odmotavanja, za svaki zaseban materijal nanosi se jednostavna bojana prethodno označenim područjima. Za izmjenu terena potrebno je ući u opciju izmjenjivanja (*Edit mode*), te označiti željena mjesta i spremi ih kao skupinu točaka pod zajedničkim nazivom. Zatim se označe mjesta koja čine planine i pridodaju im se materijali karakteristični za planinske i brdovite krajolike.



Slika 21. Naznačena mjesta za stjenoviti materijal

Na Slici 21. se vidi cijelo naznačeno područje koje će koristiti planinski, brdovit materijal. Na spremljeno označeno područje se primjenjuje materijal, a na ostale dijelove terena se primjenjuje drugačiji materijal specifičan za to područje. Za odabir materijala ide se u izbornik svojstava (eng. «*Properties*») gdje se materijal može i izraditi. Materijal se može izraditi od više sjenčara (eng. «*Shaders*») koji međusobnom kombinacijom mogu stvoriti zanimljive vizualne efekte. Dobar sustav kombiniranja sjenčara je u alatu koji je ugrađen u Blender, pod nazivom «*Node Editor*». U *Node Editoru* se komplicirane kombinacije sjenčara mogu lakše grafički prikazati tako da se svi elementi koji se međusobno spajaju povežu oku ugodnom krivuljom. *Node Editor* u konačnici nudi jednostavnije manipuliranje materijalom. Isti sustav miješanja sjenčara je prisutan i u drugim konkurentnim 3D programima te je za sada najnapredniji sustav za tu svrhu.

U sceni koja je zamišljena za izradu u ovom radu neće biti korištene napredne postavke miješanja sjenčara jer su elementi koji čine scenu jednostavnog vizualnog stila. U te potrebe je dovoljno pod postavkama nekog modela odabrati opciju «*Materials*» te dodati standardan «*Diffuse*» sjenčar, odabrati željenu boju, primijeniti je na odgovarajuću geometriju i dobije se gotov model.



Slika 22. Materijali primijenjeni na različitim dijelovima

Na Slici 22. se vide različiti dijelovi terena ukrašeni različitim materijalima. Ovo je teren bez ikakvog utjecaja svjetla, među objektnih interakcija, svjetlosnih refleksija ili sjena. Korišteni materijali su jednostavni, odabrane boje oku ugodne i primjetan je cjelokupni dojam konzistentnosti.

## 6.5. Izrada završne scene

Nakon što je cijeli teren izmodeliran, materijali postavljeni a konačna slika terena se polako kristalizira dodaju se i završni detalji. Nastavlja se sastavljati scena, te kao jedna od najzanimljivijih stvari koje su u središtu zanimanja je kućica. Na skici se vidi da je kućica u podnožju planine, te kako bi ona što više plijenila pažnju ovoga puta će se napraviti iznimka u odnosu veličina. *Kućica* će biti veća nego je inače. To nije problem, jer je vizualan stil u ovom radu takav da realističnost nije od velike važnosti.

Uz sve to, odlučeno je da se radu dodaju neki elementi kako da scena bude zanimljivija. Oko kućice se postavila ograda, iz dimnjaka se dimi, morem plovi jedrilica, alge krasi pješčanu obalu itd. Male stvari dodane na prava mjesta u konačnici čine veliku razliku. Izrada ovakvog vizualnog stila nije vremenski prezahtjevna, ali veća količina objekata koje je potrebno napraviti za scenu oduzima dobar dio vremena.

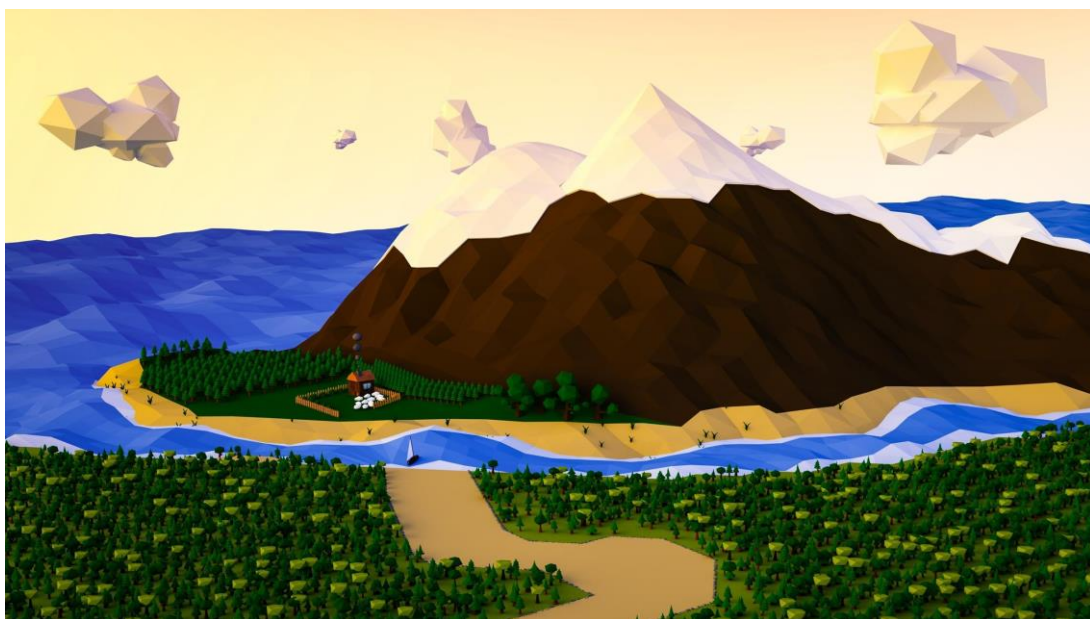


Slika 23. Konačni modeli za scenu

Izmodelirana je cijela scena, od početka do kraja, uključujući sve modele koji se u njoj koriste. Ovisno kako je scena zamišljena, lako je moguće u trenutku improvizirati radi unaprjeđenja konačnog izgleda. Tako se i prilikom izrade ovog

rada scena malo po malo se unaprijedila, a kompozicija i elementi su se preslagali u cilju izrade što zanimljivijeg okruženja.

Kako nešto u konačnici izgleda uvelike ovisi o osvjetljenju. Boja osvjetljenja ključan je segment u tome hoće li okruženje biti u redu, vrlo dobro ili očaravajuće. U ovoj sceni primaran izvor osvjetljenja bila je okolna svjetlost. Simulirano je osvjetljenje slično onom koje se može vidjeti pri zalasku sunca.



*Slika 24. Konačna scena*

Od zamišljene ideje, ideje prenesene na običan list papira, laganim koracima slijedeći dobru praksu postupnog modeliranja došlo se do konačnog modela terena. Uspješno je izmodeliran prvotno zamišljen krajolik u potpunosti koristeći program Blender. Preostaje uvoz izrađenih modela koji će se koristiti u razvojnom okruženju Unity.

## **7. PRAKTIČNI DIO IZRADE 3D OKRUŽENJA- UNITY**

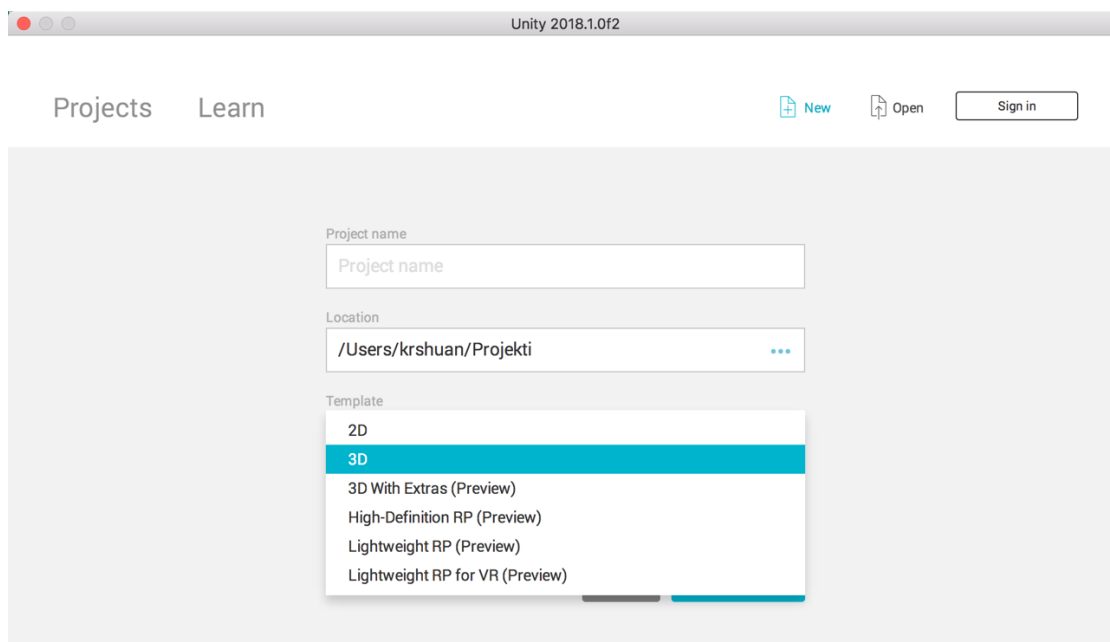
Razvojna okruženja svojim alatima i uslugama omogućuju razvojnim timovima izradu igre brzo i jednostavno. Mogu se nazvati platformama za razvijanje igara. Podržavaju manipulaciju multimedijalnim datotekama i sjedinjuju nekoliko glavnih komponenti koje u konačnici čine igru. Danas ona podržavaju unos različitih grafika i multimedijalnih dijelova iz raznih programa u kojima se mogu izrađivati. Mogu se na primjer uvesti objekti izmodelirani u Blenderu ili teksture izrađene u Photoshopu, a od tih dijelova sastaviti novi model koji će krojiti završnu scenu.

Na tom konceptu se zasnovao i ovaj rad, a uz sve dijelove iz vanjskih programa uvezenih u razvojno okruženje, konačna scena će koristiti i neke od već integriranih funkcija razvojnih okruženja. Funkcije poput osvjetljenja, zvukova, specijalnih efekata, fizike i animacije samo su neke od stvari koje se mogu dodati završnoj sceni ili modelima kako bi se igra napravila zanimljivom. Izrada igre za određenu igraću platformu u većini se razvojnih okruženja može posebno prilagoditi, a dijelovi koji se za nju izrađuju posebno optimizirati radi što boljeg izvođenja.

### **7.1. Izrada novog projekta u Unity-u**

Zbog odabranog low-poly stila vizuala, jednostavnih boja i nezahtjevne geometrije, izrada terena unutar Unity-a nije bila potrebna. Izrada terena u Unity razvojnom okruženju omogućena je uz raznorazne alate koji stoje na raspolaganju modeleru, te je s njima u potpunosti moguće modificirati teren. U ovom radu odlučeno je kako je jednostavnije izraditi teren u Blenderu i potom ga unijeti u Unity-u.

Prilikom pokretanja Unity-a odabire se vrsta igre koja će se izrađivati. Odabire se između 2D ili 3D igre.

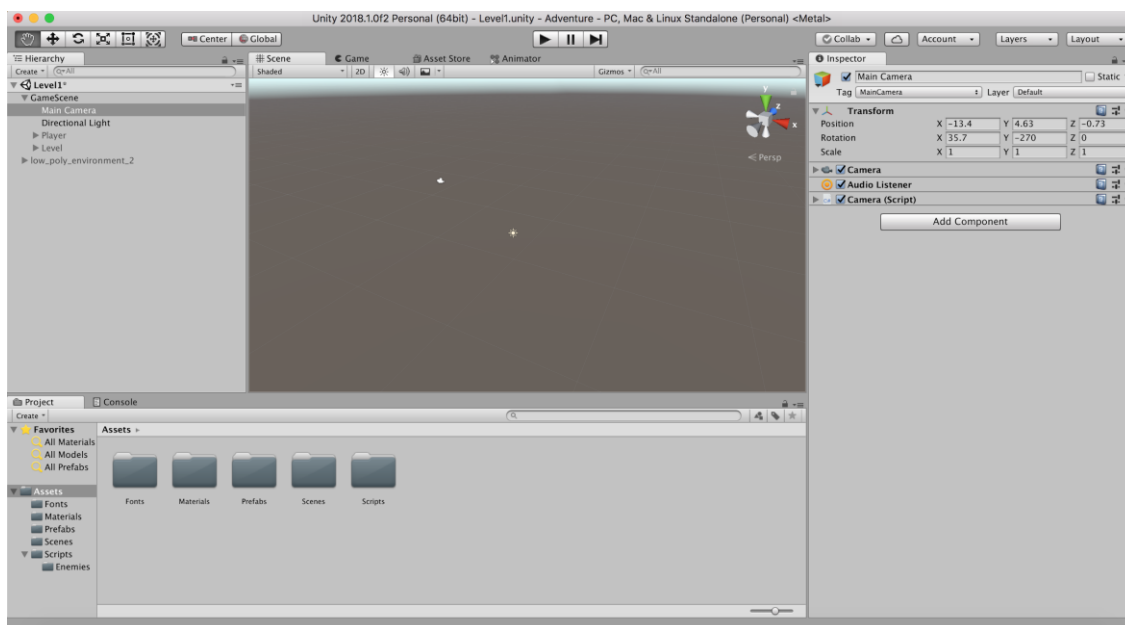


Slika 25. Odabir vrste projekta

Unity je podjednako prikladan za stvaranje 2D i 3D igara. Kako je već odlučeno da će se ovaj rad izrađivati pomoću 3D modela, odabrat će se opcija izrade 3D projekta. Neovisno o tome koja je opcija na početku odabrana, tijekom izrade igre se uvijek može promijeniti tip projekta. Postoje neki slučajevi kada se vizualan stil može ispreplitati, tako da je u jednoj sceni kompleksan 3D svijet s pregršt opcija, smjerova kretanja, radnji za napraviti, a u drugoj sceni jednostavna 2D scena poput kartaške igre. To su malobrojni slučajevi ali i u takvim se projektima u Unity-u može iskombinirati najbolje od oba svijeta.

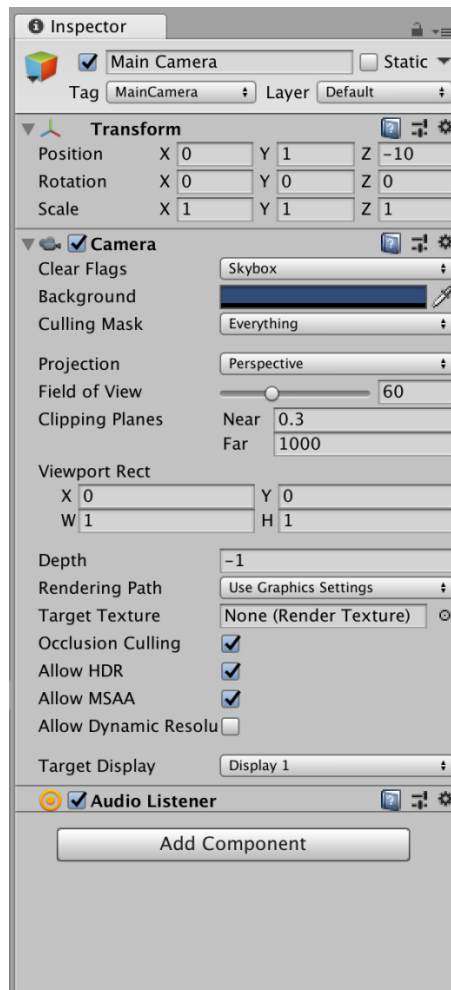


Nakon što se učitaju svi potrebni dijelovi, prikaže se sučelje s rasporedom prozora kao što je vidljivo na *Slici 25*. U prozoru lijevo, pod nazivom hijerarhija (eng. Hierarchy) nalaze se svi objekti na sceni. U tom prozoru objekti se mogu dodavati, stvarati, pomicati po važnosti itd.



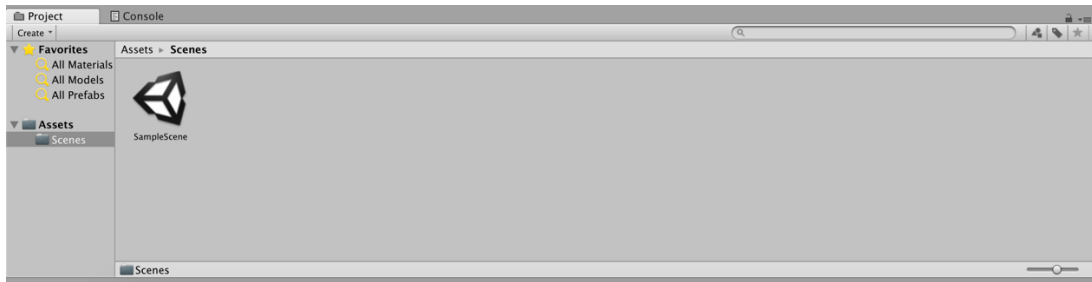
*Slika 25. Razvojno okruženje za 3D igru u Unity-u*

Prozor u sredini predstavlja trodimenzionalan prostor gdje se svi ti modeli nalaze. U prozoru u sredini vide se svi objekti u sceni u punoj veličini. Unutar srednjeg prozora također se svi objekti mogu pomicati, označavati, preslagati te se njima može manipulirati. Uz to, srednji prozor može služiti kao svojevrsna kamera kroz koju se može vidjeti kako će igra s trenutnim parametrima izgledati iz igračeve perspektive. 3D sustav u Unity-u je vrlo sličan onom u Blenderu, samo što su Y i Z osi zamijenjene. Z os u Unity-u predstavlja širinu a Y os predstavlja visinu, dok je u Blenderu obrnuto. U prošlosti je to predstavljalo problem zato što su se 3D modeli izvozili iz Blendera te su se nakon otvaranja u Unity-u rotirali da bi bili normalne rotacije. Isto se to moglo napraviti prvo u Blenderu, te nakon toga izvesti model, ali se s novijim verzijama taj problem riješio.



Slika 26. Inspektor (Inspector)

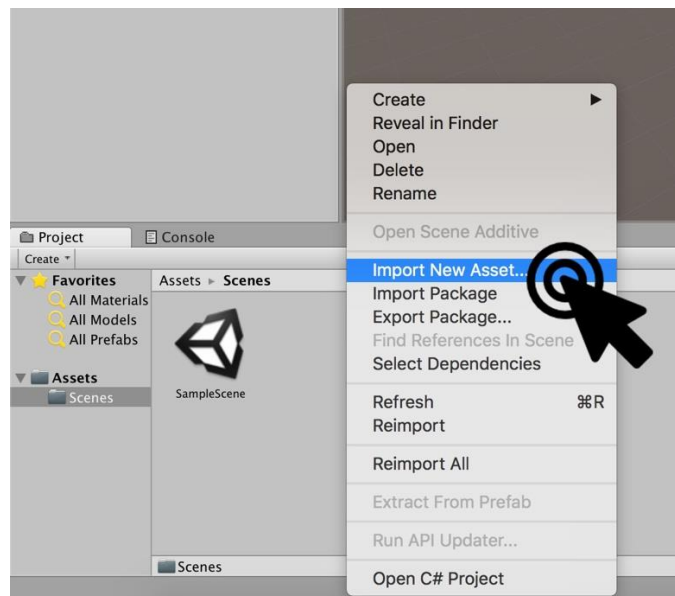
Prozor s desne strane, pod nazivom *Inspektor* (Slika 26.), služi uređivanju postojećih stavki ili dodavanju novih. Nakon što se objekt odabere, u Inspektoru mu se može mijenjati pozicija, rotacija, veličina, materijali te raznorazni parametri specifični za tu vrstu objekta. U Unity-u se mogu kreirati i skripte, koje će pridodanim objektima mijenjati vrijednosti. Tako se uz malenu programersku pomoć može kreirati skripta koja objektu mijenja poziciju, ili ga rotira, te tako izraditi malenu kretajuću prepreku glavnom igraču. Isti princip se može primijeniti i za neprijatelja.



Slika 27. Projekt prozor (Project window)

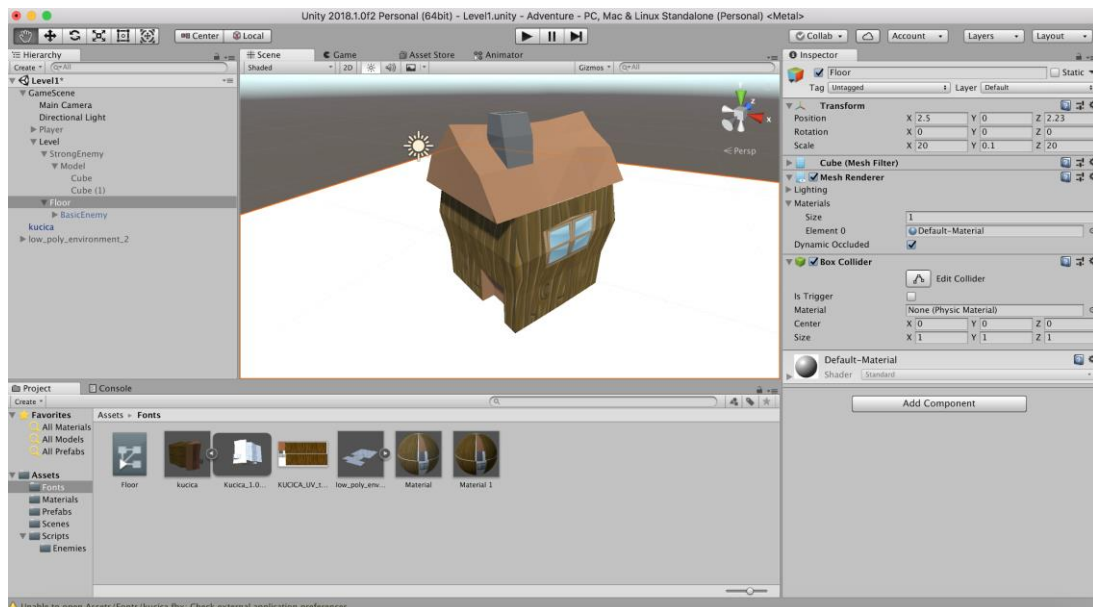
Prozor u podnožju ekrana je prozor pod nazivom *Project* i u njemu se nalaze svi potrebni dijelovi za izradu scene. Od modela, slika, tekstura, materijala, fontova, skripti pa sve do zvukova i zvučnih efekata, sve to sadrži prozor Project. Unutar njega se naravno mogu i izrađivati novi dijelovi, novi materijali, nove skripte, novi specijalni efekti, animacije, te sve potrebno za izradu igre. Prilikom instalacije Unity-a ponudi se i instaliranje Visual Studio-a, programerskog razvojnog okruženja koje uveliko olakšava pisanje skripti. Sve radi vrlo kvalitetno i u konstantnoj međusobnoj sinkronizaciji. U *Project* prozoru se uvodi model izrađen u Blenderu koji će se upotrebljavati u ovoj sceni. Kada se jednom uvedeni model odluči iskoristiti u sceni, prilikom njegove uporabe stvorit će se objektna instanca koja će taj objekt predstavljati u sceni. Ako se taj objekt iz scene izbriše, lako se ponovno kreira njegova instanca iz *Project* prozora. Proces izrade scene se nastavlja u Unity-u i objekt Kućica izrađen u Blenderu se ubacuje u razvojno okruženje.

## 7.2. Uvoz modela u Unity



Slika 28. Uvoz izrađenog modela u Unity

Iz Blendera se model Kućice izvozi kao *.fbx* format, te se ubacuje u prozor Project. Prilikom stavljanja kućice u scenu automatski se događa konverzija i kućica se rotira onako kako bi trebala, automatski zamjenjujući onaj «nesporazum» između Y i Z osi.

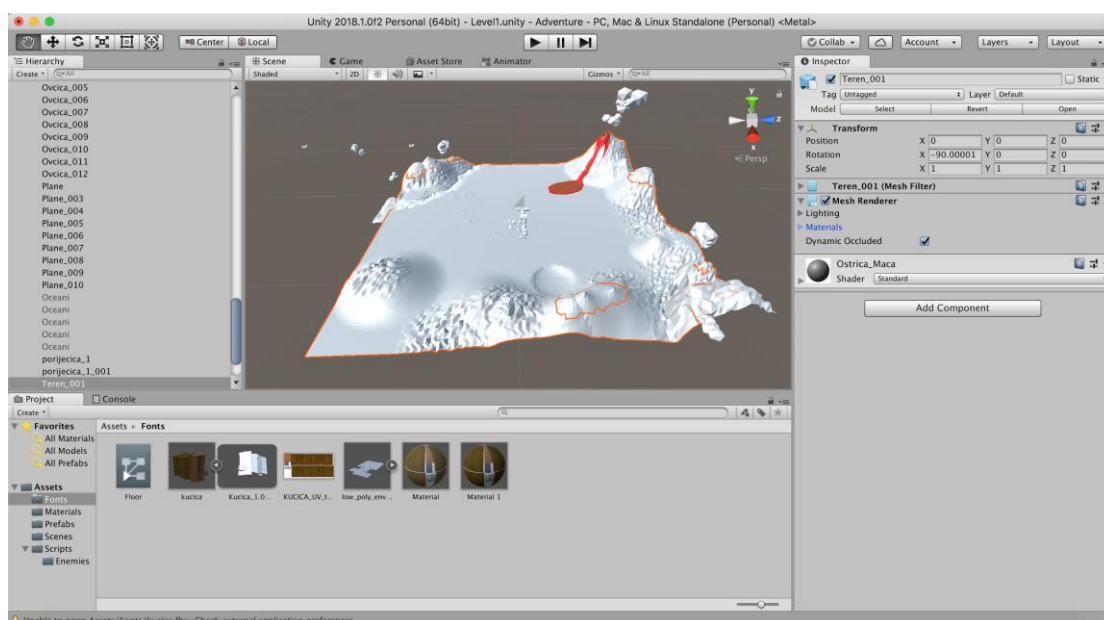


Slika 29. Model Kućice u *.fbx* formatu unutar Unity-a

Na Slici 29. se vidi primjena izrađene teksture u Photoshopu kao materijala na modelu kućice. Tekstura se savršeno uklopila na model.

Umetanje modela u Unity razvojno okruženje kao *.fbx* datoteke funkcionira dobro, ali Blender i Unity imaju jaku sinergiju pa je postupak moguće ubrzati. Unity može učitati izvornu Blender datoteku ekstenzije *.blend*, tako da modele koji se kreiraju u Blenderu ne treba posebno prilagođavati da bi se koristili u Unity razvojnom okruženju. Paralelno se može modelirati objekt u Blenderu, a u Unity-u manipulirati tim istim objektom s tim da se promjene napravljene u Blenderu instantno prikažu u Unity-u. Takav radni tok dodatno ubrzava stvari i poželjan je radi efektivnosti.

Nakon toga preostaje uvoz *.blend* datoteke terena i uređivanje istog. U *Project* prozor se odvuče *.blend* datoteka i zatim se učita u sceni. To može potrajati nekoliko sekundi, ovisi o veličini objekta.

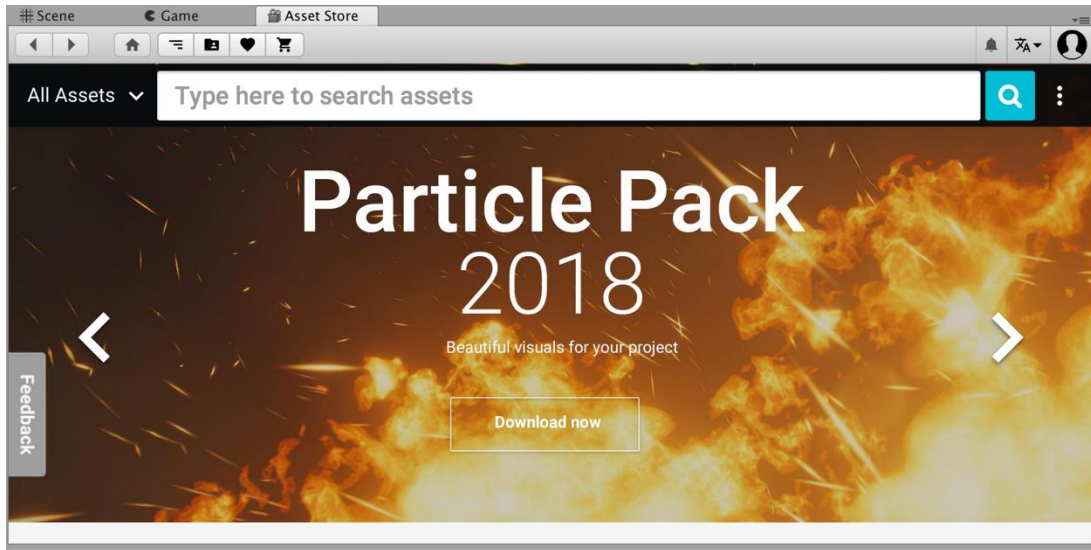


Slika 30. Umetanje *.blend* datoteke u Unity

Uvoz datoteke isto se tako može obaviti time da se željeni model ili neka druga komponenta jednostavno odvuče u Projekt prozor, vrlo intuitivno i efikasno. Ta opcija jako je poznata svim korisnicima računala kao «*drag & drop*» funkcija. Nakon odvlačenja ili običnog uvoza komponente u Projekt prozor, Unity automatski analizira sadržaje te komponente. Ako Unity prepozna da je komponenta sastavljena od više stavki, svaku stavku pojedinačno prikaže u Projekt prozoru te je s njima moguće manipulirati.

### 7.3. Unity trgovina

Unity raspolaže s ugrađenom trgovinom dijelova za razvijanje igre. U Unity Asset Store trgovini može se preuzeti na tisuće besplatnih i plaćenih modela, zvukova, animacija ili tekstura koje se kasnije mogu koristiti u izradi igre [16].



Slika 31. Unity trgovina s dijelovima

Izrada igre timski je posao koji iziskuje ogromne količine vremena, različitih znanja, pristupa i razmišljanja, stoga je uobičajeno da se razvojem igre bave studiji s nekoliko različitih timova i veći. Samostalna izrada je dugotrajan i otežan proces u usporedbi s izradom u timu, stoga je dobro iskoristiti svaku pomoć koja se nudi. Upravo tu pomoć može se naći u Unity-ovoj ugrađenoj trgovini dijelova u obliku već izrađenih modela, tekstura, animacija itd. Ako je neki objekt potreban u sceni, a dostupan je za besplatno preuzimanje u Unity trgovini, bilo bi dobro to onda iskoristiti. U ovom radu se naglasak stavlja na izradu 3D okruženja, pa su svi modeli koji čine scenu, posebno izrađeni, ali treba imati na umu korištenje Unity trgovine.

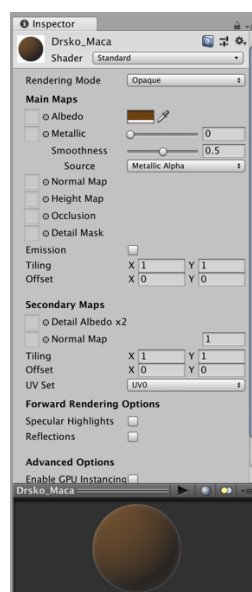
## 7.4. Primjena materijala u Unity-u

Materijali i njihova inkompatibilnosti između različitih softvera čest je slučaj, pa zbog toga cijeli uvezen teren izgleda kao da je obojen u bijelu boju.



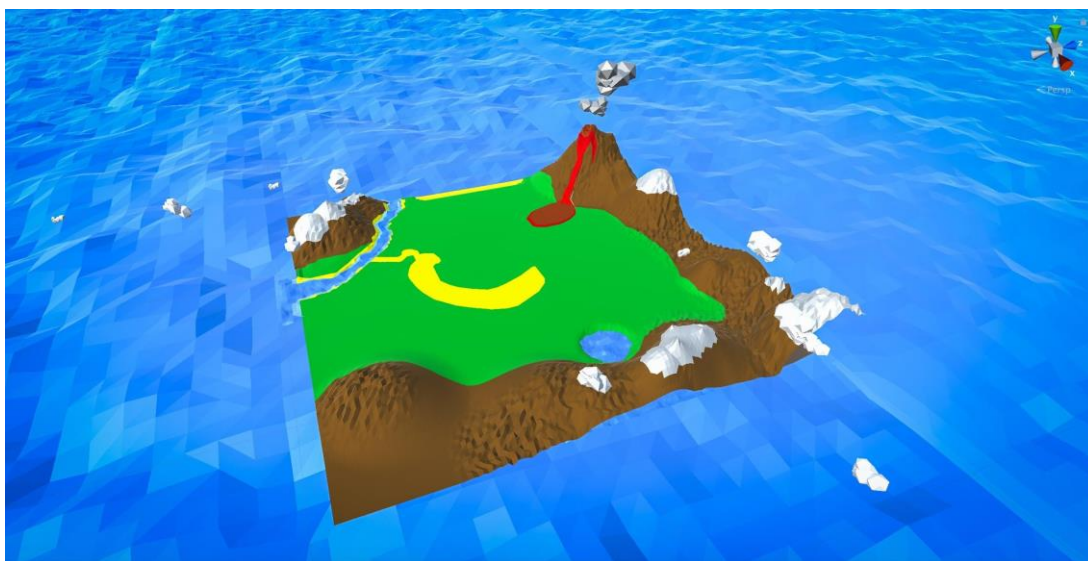
Slika 32. Postupak izrade materijala u Unity-u

Zbog jednostavnog vizualnog stila korištenog kroz čitav rad, izradit će se materijali u vrlo sličnom stilu kao i prilikom njihove izrade u Blenderu. U Project prozoru kreiraju se novi materijali jednostavnih ali živopisnih boja. Broj izrađenih materijala će biti jednak broju izrađenih materijala u Blenderu.



Slika 33. Panel za modificiranje materijala

Nakon što su se svi materijali izradili, primijenili su se na točno željene dijelove. Datoteka koju Blender izradi prilikom spremanja u sebi ima pohranjena mjesta na kojima je već bio neki materijal, samo se taj materijal prilikom uvoza u Unity ne može prikazati, zato je svaki uvezen materijal jednake, bijele boje. Na mjestima gdje je bio planinarsko brdovit materijal, sada se postavio novo izrađen materijal iste boje. To se ponovilo za sve materijale na raspolaganju te su se nakon toga materijali jednostavno primijenili.



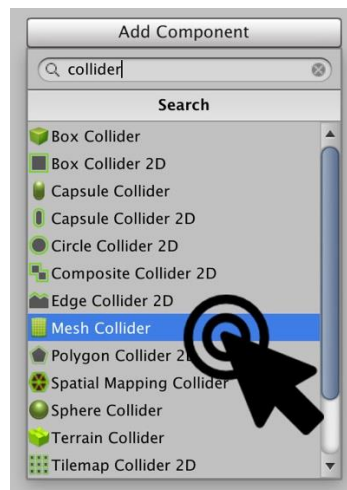
*Slika 34. Teren sa primijenjenim materijalima*

Na *Slici 34.* prikazan je teren sa svim primijenjenim materijalima prilagođen razvojnom okruženju Unity. Može se vidjeti kako boje, uz sami vizualni dojam nisu ni blizu željenoj, konačnoj formi kojoj se teži. To je zato što se u glavnom prozoru ne prikazuju svi segmenti koji će biti prisutni u konačnoj sceni. U sceni se ne prikazuje svjetlost kakva će se prikazivati u konačnici, a postoje i opcije isključivanja refleksija svjetlosti u objektima u sceni. Postoje i druge opcije isključivanja dodatnih vizualnih efekata radi boljih performansa prilikom slaganja same scene.



## 7.5. Završni detalji

Da bi se igrač mogao kretati po stvorenoj geometriji, u Unity razvojnom okruženju je potrebno u desnom prozoru (*Inspector*) dodati komponentu «*Mesh collider*». Ona daje objektu karakteristiku da bilo što u doticaju s njom reagira, tj. da se ostvaruje interakcija. Bez te komponente, stvoreni teren ne bi imao nikakve značajke terena, tj. kad bi se na njega postavili svi objekti što trebaju činiti scenu, jednostavno bi propali kroz teren kao da ne postoji.

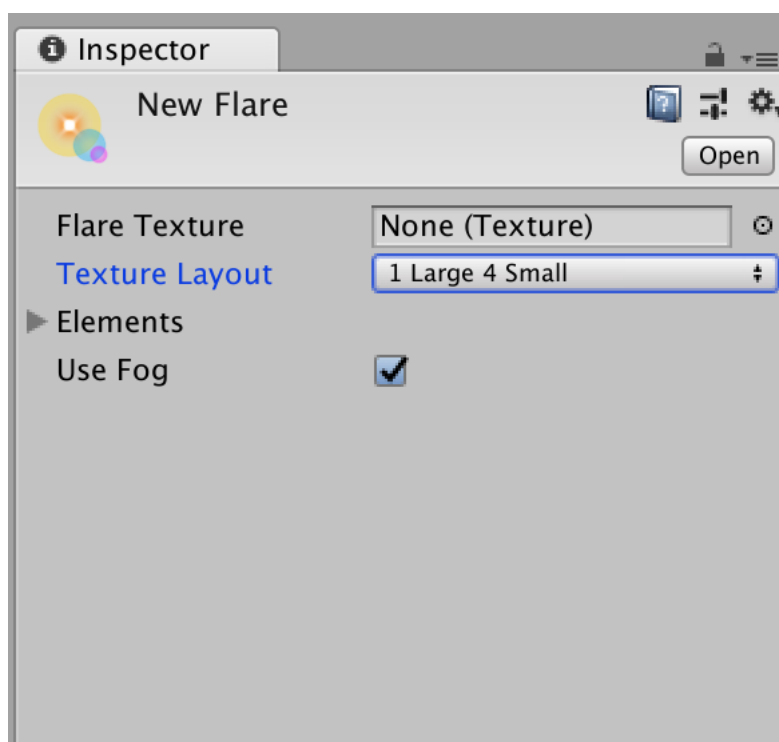


Slika 35. Dodavanje opcije "Mesh Collider" modelu

*Mesh Collider* je opcija koja je pogodna za dodavanje terenu, jer nije jednostavnog oblika. Ona generira prepreke ili prohodan put za igrača ovisno o svojoj geometriji. Primjerice, planina neće biti prohodna zbog toga što je visoka i ne može se njom jednostavno hodati, dok će ravni putevi biti prohodni i njima će se moći doći do cilja. *Mesh Collider* opcija je precizna i daje najbolje rezultate za ovaj zadatak. No preciznost ima svoju cijenu. Kako se prepreke računaju kroz mnogo točaka od kojih je geometrija terena rađena, to negativno utječe na performanse izvođenja, jer više geometrije za računanje znači i korištenje više procesorske snage. Stoga je poželjno ne pretjerivati s uporabom *Mesh Collidera* u objektima koji se nalaze na sceni, već je preporučivo koristiti *Box Collidere* ili slične *Collidere* koji su izrađeni od male geometrije. Na primjer, ako je u pozadini scene neka nebitna kuća koja služi samo za ograničavanje igračeve kretanje te za upotpunjavanje krajolika, nije potrebno koristiti *Mesh Collider* na apsolutno svim njezinim dijelovima, od dimnjaka do prozora pa sve do vrata. Na kuću se može primijeniti *Box Collider* koji će izvrsno poslužiti svrsi, ograničit će igračevo kretanje,

a procesorska snaga potrebna za njeno izračunavanje će biti malena, jer se radi o maloj geometriji *Box Collidera*.

Unity razvojno okruženje je bogato funkcijama koje mogu obogatiti svaku scenu, neovisno o kakvom se vizualnom stilu u njoj radilo. Jedna od opcija koja se može dodati je ta da prilikom pokreta igrača, tj. da se prilikom njegovog pogleda prema nebu u sunce (u slučaju da se igra izvodi iz prvog lica) na nebu ocrtaju zrake koje idu iz pravca sunca prema igračevom oku (kameri). To simulira efekt koji se u stvarnosti može vidjeti prilikom pomicanja objektiva kamere prema suncu. Nije efekt koji se događa kada ljudsko oko pogleda prema suncu ali kako se igra izvodi na zaslonu nekog monitora ili televizije, ljudski mozak tu pojavu percipira kao normalnu.



Slika 36. Dodavanje bljeskanja (Flare)

U opcijama Inspektora mogu se postaviti različiti parametri koji će odrediti blještavost novo dodane komponente u sceni. Eksperimentiranjem se mogu dobiti razni zanimljivi rezultati, ovisno o tome što se želi postići, može se ići na realizam ili na nekakav znanstveno fantastični efekt.

Uz to su dodani jednostavni modeli ranije izrađeni u Blenderu, dodana im je *Mesh Collider* ili *Box Collider* komponenta i teren kojim će igrač koračati je spreman.



*Slika 37. Teren u Unity-u*

## 8. ZAKLJUČAK

Izrada videoigre je kompleksan proces koji iziskuje ogromne količine vremena, različitih znanja, pristupa i razmišljanja te se kod opcije samostalne izrade igre valja dobro pripremiti. U ovom radu pojašnjene su osnove i principi izrade, kompleksno modeliranje i sastavljanja modela, te opcija korištenja besplatnih modela koji se nude po raznim zajednicama, trgovinama unutar razvojnih okruženja itd.

Prikazani su konkretni razlozi zašto se i kada pojedini softver koristi, te su u skladu s odabirom pojedinačno detaljnije obrazloženi. Poseban naglasak stavljen je na najčešće greške koje početnici rade ulaskom u ovaj svijet, te su obrazložene dobre prakse kako ih izbjeći uz što manje troškove.

Danas se već mogu naći video kartice nekih proizvođača koje podržavaju izračun putanje svjetlosti u stvarnom vremenu (*ray-tracing*), što se već duže vrijeme smatra svetim gralom u ovoj industriji. Jačanjem hardvera iz godine u godinu, procesi izrade video igara bit će sve kompleksniji. Ipak, načela izrade koja su objašnjena u ovom radu će ostati ista, a to je dobra početna točka od koje zainteresiran pojedinac može krenuti.

## 9. POPIS SLIKA I TABLICA

Slika 1. Zarada industrije videoigara .....	2
Slika 2. Razlozi zašto Blender .....	5
Slika 3. Razlozi zašto Photoshop .....	8
Slika 4. Razlozi zašto Unity .....	14
Slika 5. Odabrani softver .....	15
Slika 6. Skica terena.....	16
Slika 7. Postavljanje odnosa veličina.....	17
Slika 8. Low-Poly Stabalce.....	20
Slika 9. Modeli koji će se koristiti za izradu šume.....	22
Slika 10. Malo izmijenjeni modeli stabala .....	23
Slika 11. Gotov model kućice .....	24
Slika 12 Slika 12. "Odmotavanje" 3D modela.....	25
Slika 13. Naznačavanje "rezova" na modelu .....	26
Slika 14. «Odmotana kućica».....	26
Slika 15. Izvoz UV mape iz Blendera .....	27
Slika 16. UV mapa modela u Photoshopu .....	28
Slika 17. Konačna tekstura kućice.....	29
Slika 18. Konačni model kućice sa teksturom .....	30
Slika 19. Podijeljena ploha na malene poligone .....	31
Slika 20. Izmodelirani krajolik .....	33
Slika 21. Naznačena mjesta za stjenoviti materijal.....	34
Slika 22. Materijali primijenjeni na različitim dijelovima .....	35
Slika 23. Konačni modeli za scenu.....	36
Slika 24. Konača scena .....	37
Slika 25. Odabir vrste projekta .....	39
	52

Slika 26. Inspektor (Inspector).....	41
Slika 27. Projekt prozor (Project window).....	42
Slika 28. Uvoz izrađenog modela u Unity.....	43
Slika 29. Model Kućice u .fbx formatu unutar Unity-a.....	43
Slika 30. Umetanje .blend datoteke u Unity.....	44
Slika 31. Unity trgovina s dijelovima.....	45
Slika 32. Postupak izrade materijala u Unity-u.....	46
Slika 33. Panel za modificiranje materijala.....	46
Slika 34. Teren sa primijenjenim materijalima.....	47
Slika 35. Dodavanje opcije "Mesh Collider" modelu.....	48
Slika 36. Dodavanje bljeskanja (Flare).....	49
Slika 37. Teren u Unity-u.....	50

## 10. POPIS MANJE POZNATIH RIJEČI

**Programi otvorenog koda** - programi za čiji je nastanak odgovorna jedna osoba, zajednica ili skupina ljudi, a koda potpuno dostupnog svima zainteresiranima te je besplatan za korištenje (str. 3.)

**Montažiranje** - u 3D-u, primjena "kostura" na neki objekt. Taj kostur objektu daje mogućnost manipuliranja, te se često koristi pri animiranju likova, ili izradi pokretnih likova u igri (str. 4.)

**Oblikovanje** - u 3D-u, tehnika oblikovanja modela slična oblikovanju gline u stvarnosti (str. 4.)

**Teksturiranje** - primjena neke teksture (boje, uzorka, slike) na određenu podlogu (str. 8.)

**Razvojno okruženje** - sustav alata i usluga potrebnih za izradu nekog softvera, u ovom slučaju video igre (Str. 9.)

**Igraća platforma** - sustav na kojem se određena igra izvodi (Playstation, Xbox, PC...) (str.9.)

**PC konfiguracija** - popis svih komponenti računala (str. 11.)

**Benchmark** - skup testova različite prirode radi utvrđivanja maksimalnih performansi računala (str. 11.)

**Poligon** - geometrijski oblik s promjenjivim brojem stranica, najjednostavniji poligon je trokut (str. 16.)

**Low-poly modeliranje** - tehnika modeliranja tako da se željeni objekti izmodeliraju uz što manje geometrije, a da zadovoljavaju traženi vizualni stil (str. 16.)

**Sustav za generiranje slike** - popis pravila i algoritama po kojima računalo izračunava putanju svjetlosti i kako će slika u konačnici izgledati (str. 19.)

**UV mapa** - 2D karta pojedinih točaka 3D modela, koristi se za pravilno teksturiranje modela (str. 26.)

**Sjenčar** - komponenta koja samostalno ili u kombinaciji više sjenčara izrađuje konačan materijal spreman za primjenu (str. 33.)

**Skripta** - popis naredbi koje automatiziraju određene procese u razvojnim okruženjima. (str. 38.)

## 11. LITERATURA

1. **Peter Zackariasson**, Timothy Wilson (lipanj 2014.) The Video Game Industry: Formation, Present State, and Future (Routledge Studies in Innovation, Organization and Technology), knjiga o financijama u igraćoj industriji; Nakladnik: Routledge
2. Forbes, američki poslovni časopis (lipanj 2013.)  
Dostupno na: [www.forbes.com](http://www.forbes.com), lipanj 2013.
3. **Kelly Murdock** Autodesk (studeni 2017.) Maya 2018 Basics Guide, stručni priručnik za Autodeskove programe; Nakladnik: SDC Publications
4. Pixologicova službena dokumentacija  
Dostupno na: [pixologic.com](http://pixologic.com), srpanj 2018.
5. **Gordon Fisher** (kolovoz 2014.) Blender 3D Basics: Second Edition (knjiga o osnovama programa Blender); Nakladnik: Packt Publishing
6. GIMP-ova službena dokumentacija  
Dostupno na: [www.gimp.org](http://www.gimp.org), srpanj 2018.
7. Kritina službena dokumentacija  
Dostupno na: [www.krita.org](http://www.krita.org), srpanj 2018.
8. Andrew Faulkner (siječanj 2018.) Adobe Photoshop CC Classroom in a Book (knjiga o naprednijem korištenju Adobe Photoshopa); Nakladnik: Adobe Press
9. **Joe Hocking** (travanj 2018.) Unity in Action: Multiplatform game development in C# (knjiga o osnovama razvojnog okruženja Unity); Nakladnik: Manning Publications
10. Unreal-ova službena dokumentacija  
Dostupno na: [docs.unrealengine.com](http://docs.unrealengine.com), srpanj 2018.
11. Crytek-ova službena dokumentacija  
Dostupno na: [docs.cryengine.com](http://docs.cryengine.com), srpanj 2018.
12. Allan Brito (svibanj 2008), Blender 3D Architecture, Buildings, and Scenery, knjiga o modeliranju u Blenderu; Nakladnik: Packt Publishing
13. BlenderArt profesionalni časopis (siječanj 2008.) o low-poly modeliranju  
Dostupno na: [blenderart\\_Jan08\\_Issue14.PDF](#), siječanj 2008.



14. Enrico Valenza (2015.) Blender Cycles: Materials and Textures Cookbook, Third Edition, (knjiga o manipulaciji materijalima u programu Blender);  
Nakladnik: Packt Publishing
15. Istraživanje o teksturama objavljeno na stručnom portalu Katsbits,  
Dostupno na: [www.katsbits.com](http://www.katsbits.com), 2015.
16. Alan Thoran, Dr. Edward Lavieri (veljača 2017.) Rapid Game Development with Unity 5, knjiga o izradi igre pomoću modela u programu Unity;  
Nakladnik: Packt Publishing