Izrada nivoa za video igru

Žgela, Martina

Undergraduate thesis / Završni rad

2020

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

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:122:102117

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-03-29**

Repository / Repozitorij:

University North Digital Repository

Sveučilište Sjever

Završni rad br. 667/MM/2020

Izrada nivoa za video igru

Martina Žgela, 1670/336

Varaždin, rujan 2020. godine

Sveučilište Sjever

Odjel za Multimediju, oblikovanje i primjenu

Završni rad br. 667/MM/2020

Izrada nivoa za video igru

Student

Martina Žgela, 1670/336

Mentor

doc.dr.sc. Andrija Bernik

Varaždin, rujan 2020. godine

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

Prijava završnog rada

Definiranje teme završnog rada i povjerenstva

ODJEL Odjel za multimediju

STUDIJ preddiplomski stručni studij Multimedija, oblikovanje i primjena

PRISTUPNIK Martina Žgela | MATIČNI BROJ 1670/336

DATUM 20.02.2020. | KOLEGIJ 3D Modeliranje

NASLOV RADA Izrada nivoa za video igru

NASLOV RADA NA ENGL. JEZIKU Video game level creation

MENTOR doc.dr.sc. Andrija Bernik | ZVANJE Docent

ČLANOVI POVJERENSTVA

1. pred. Nikolina Bolčević Horvatić, dipl.ing. - predsjednik
2. pred. Snježana Ivančić Valenko, dipl.ing. - član
3. doc.dr.sc. Andrija Bernik - mentor
4. doc.art. Robert Geček - zamjenski član
5.

Zadatak završnog rada

BROJ 667/MM/2020

OPIS

Industrija video igara jedna je od najbrže rastućih industrija na svijetu, a danas se video igre više ne smatraju isključivo zabavom. Važnost ove industrije će prema brojnim statistikama te dosadašnjim uspjesima nastaviti rasti, zbog čega je posebno zanimljiv sektor grafičke industrije koji se bavi njihovom izradom. U ovom radu bit će opisan proces izrade nivoa za video igru putem alata Blender te izrade simulacije u Unity razvojnom okruženju. Blender je besplatan softver otvorenog koda namijenjen za izradu 3D računalne grafike široke namjene, a popularan je odabir za izradu video igara. Vrlo je kompatibilan s Unity razvojnim okruženjem (eng. game engine), zbog čega će se završni dio rada sastojati od jednostavne, kraće simulacije kako bi se u cijelosti mogao dočarati izrađeni nivo.

U radu će, uz kraći teoretski osvrt na korištene alate, biti opisan postupak izrade modela i pripadajućeg krajolika za nivo video igre srednjovjekovne tematike, inspirirane kultnom igrom Skyrim iz 2011. godine, kao i postupak izrade simulacije.

REPUBLIKA HRVATSKA
SVEUČILIŠTE SJEVER

ZADATAK URUČEN 19.06.2020. | POTPIS MENTORA

Predgovor

Tema završnog rada odabrana je iz osobnih interesa prema intrigantnom i kontinuirano rastućem svijetu video igara. Cilj je rada steći nova znanja i pobliže se upoznati s odabranim programima za izradu video igara.

Ovim bih se putem htjela zahvaliti mentoru doc.dr.sc. Andriji Berniku za savjete i pomoć koje je pružao kroz studij i prilikom izrade rada, te na otvorenosti i razumijevanju koje je preslikao u vođenje svojih kolegija na Sveučilištu Sjever.

Također se zahvaljujem svojim roditeljima koji su bili velika podrška tijekom studija te svom partneru na podršci i strpljenju.

Sažetak

U ovom je radu prikazan proces izrade nivoa za jednostavnu video igru, pomoću popularnih alata Blender i Unity. Blender je 3D alat koji se koristi za modeliranje potrebnih modela koji će sačinjavati glavnu scenu, a pomoću Unity-a sve će se spojiti u smislenu cjelinu – nivo za video igru srednjovjekovne tematike. U radu su objašnjeni osnovni pojmovi koji su usko vezani uz korištene alate, dotaknuta je i povijest, kao i prednosti i nedostaci u usporedbi s konkurentnim alatima. Kroz praktični dio rada nastoji se prikazati i pojasniti tijek izrade pojedinih dijelova procesa.

Ključne riječi: video igre, 3D model, game engine, blender, unity

Summary

This paper presents the process of creating a level for a simple video game using the popular tools Blender and Unity. Blender is a 3D tool used to model the necessary models that will make up the main scene, and with Unity everything will be merged into a meaningful whole - a level for a medieval-themed video game. The paper explains the basic concepts that are closely related to the used tools, their history, as well as the advantages and disadvantages compared to other competing tools. The practical part of the paper shows and explains the course of making individual parts of the process.

Keywords: video games, 3D model, game engine, Blender, Unity

Popis korištenih kratica

3D	Three-dimensional Kratica koja označava treću dimenziju.
IDE	Integrated Development Environment Integrirano razvojno okruženje
2D	Two-dimensional Kratica koja označava drugu dimenziju.
PC	Personal Computer Kratica koja označava osobno računalo.
UI	User Interface Korisničko sučelje
GUI	Graphical User Interface Kratica koja označava grafičko korisničko sučelje.
CPU	Central Processing Unit Kratica koja označava glavni procesor.
GPU	Graphical Processing Unit Kartica koja označava grafički procesor.
VR	Virtual Reality Virtualna stvarnost

Sadržaj

1. Uvod 1
2. Industrija video igara 3
3. 3D modeliranje 5
   3.1. Proces 3D modeliranja 5
   3.2. Terminologija 3D modeliranja za video igre 5
4. Blender 7
   4.1. Povijest 7
   4.2. Blender zajednica 8
   4.3. Sučelje 8
   4.4. Najčešće korišteni alati 9
   4.5. Prednosti i nedostaci 10
5. Unity 12
   5.1. Povijest 13
   5.2. Sučelje 13
   5.3. Prednosti i nedostaci 14
   5.4. Instalacija i pokretanje 15
6. Praktični dio – izrada nivoa za video igru 16
   6.1. Modeliranje u Blenderu 16
   6.2. UV mapiranje 23
   6.3. Teksturiranje 23
   6.4. Priprema i izvoz modela iz Blendera za Unity 26
   6.5. Izrada scene u Unity-u 27
7. Zaključak 37
8. Literatura 39

Popis slika 40

Popis tablica 42

1. Uvod

Kroz proteklih nekoliko desetljeća računalna je grafika doživjela nagli napredak te je ujedno postala i pristupačnija široj količini ljudi. Time je i sami 3D svijet postao pristupačniji i postepeno dolazi do razvoja brojnih moćnih alata koji danas čine osnovu i same gaming industrije. Industrija video igara jedna je od najbrže rastućih industrija na svijetu, a danas se video igre više ne smatraju isključivo zabavom. Važnost ove industrije će prema brojnim statistikama te dosadašnjim uspjesima nastaviti rasti, zbog čega je posebno zanimljiv sektor grafičke industrije koji se bavi njihovom izradom. U ovom radu bit će opisan proces izrade nivoa za video igru putem alata Blender te izrade simulacije u Unity razvojnom okruženju. Blender je besplatan softver otvorenog koda namijenjen za izradu 3D računalne grafike široke namjene, a popularan je odabir za izradu video igara. Vrlo je kompatibilan s Unity razvojnim okruženjem (eng. game engine), zbog čega će se završni dio rada sastojati od izrade scene i jednostavne simulacije kako bi se u cijelosti mogao dočarati izrađeni nivo.

Alati su pobliže pojašnjeni kroz teoretski te praktični dio. U teoretskom dijelu rada pokriti će se ponajprije sama industrija video igara te 3D modeliranje općenito, a zatim i osnove, glavne značajke, povijest navedenih korištenih alata te prednosti u odnosu na ostale alate. Prolaziti će se kroz sučelja te pojasniti najčešće korišteni alati i opcije. U praktičnom dijelu rada biti će pojašnjen proces izrade rada, prolazeći kroz Blender s naglaskom na modeliranje, UV mapiranje, teksturiranje i migraciju modela iz Blendera u Unity, a zatim i izradu okoline (eng. environment) u Unity-u, koja je kreirana uz pomoć besplatnih dodataka (eng. assets) te vlastitih izmodeliranih dodataka. Na kraju rada dodati će se i zvukovi kako bi se dobio što realističniji dojam.

Ovim se radom nastojalo upoznati s oba alata i samostalno prolaziti kroz ključne dijelove procesa izrade neke video igre. Također se i čitateljima nastoji približiti mogućnosti ovih izrazito sposobnih alata, kao i proces izrade jedne video igre. Blender je sposoban 3D alat koji se danas može usporediti i sa konkurentima iz industrije kao što su 3DSMax ili Autodesk Maya. Unity je sam po sebi dovoljan za izradu neke jednostavne igre, no ako je cilj prepoznatljivost i unikatnost, onda su alati kao što je Blender idealni da bi se to moglo postići. Općenito, izrada neke simulacije izazovan je posao koji se uobičajeno rješava u timu ljudi iz raznih područja kao što su 3D modeliranje, animiranje, teksturiranje i programiranje. Međutim, rad pokazuje da se određene ideje mogu realizirati i samostalno te da je uz volju i trud relativno jednostavno kreirati neku jednostavnu video igru.

Projekt se može pregledati na sljedećoj poveznici:
https://drive.google.com/drive/folders/1b7bFgWZjUJyFu-JUrhpA9jAObOJ0ygTl?usp=sharing

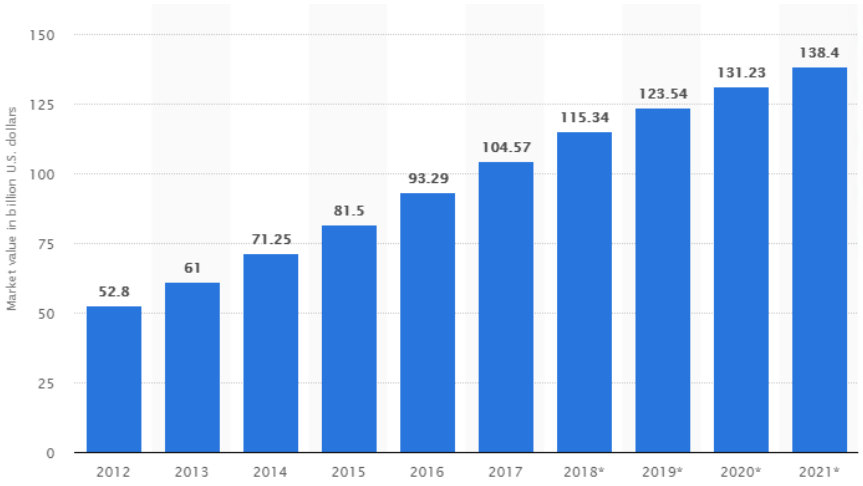
Potrebno je pokrenuti *Camp.exe* datoteku te se zatim nivo može istraživati pomoću standardnih naredbi *W, A, S, D* za kretanje, *C* za čučanj, *Space* za skokove i *Esc* za izlaz iz aplikacije.

2. Industrija video igara

Video igra (eng. video game) je elektronička igra koja se može igrati na računalnim uređajima, poput osobnog računala (eng. PC), igraće konzole (eng. console) ili mobitela. Prema tome, ovisno o platformi na kojoj se igraju, video igre se mogu podkategorizirati na računalne i konzolne. Posljednjih godina s pojavom društvenih mreža, pametnih telefona i tableta, dolazi do novih kategorija video igara, poput mobilnih i društvenih. Već desetljećima video igre pružaju zabavu djeci i odraslima. Od prve pojave 1970-ih i prvih verzija Nintenda i Atari, značajno su napredovale. Dani pikseliziranih zaslona i ograničenih zvukova daleka su uspomena. Današnje video igre postale su realnije no ikad, nudeći fotorealističnu grafiku i simulirajući stvarnost do stupnja koji je u mnogim slučajevima zaista zapanjujući.

Stvaranje videoigara postalo je sve složenije, a troškovi izrade igre koja će se izvoditi na jednoj od glavnih konzola porasli su s tom većom složenošću. Nekad je bilo nezamislivo utopiti milijune u razvojne troškove, ali danas njihov razvoj može koštati desetke, pa čak i stotine milijuna. Sektor video igara je veći od filmske i glazbene industrije zajedno i nastavlja rasti. Iako ne privlači istu pažnju kao te industrije, ipak danas postoji preko dva bilijuna igrača diljem svijeta.

Analitičari predviđaju da će do 2021.godine industrija video igara generirati prihod od 138 bilijuna dolara. Naravno, tehnološke tvrtke se žele uključiti u ovaj tok prihoda pa su tako i giganti kao što su Google, Facebook i Apple ušli u industriju video igara.



Slika 2.1 Vrijednost globalnog tržišta video igara 2012. do 2021., Izvor: Statista.com

Danas ne samo da imamo kontinuirano pojavljivanje novih konzola, računala i uređaja koji nas uvode u igre, već imamo i priliku gledati neke od najboljih svjetskih igrača koji se natječu u e-sportskoj areni. Streaming je također postao ogroman uspjeh za gaming industriju i njezine

ljubitelje. Platforme, poput Twitcha, Mixera i Youtube Gaminga, pružaju mogućnost pojedinim igračima da uživo prenose igranje pretplatnicima diljem svijeta. Bitku oko odabira konzola vode Microsoft (Xbox), Sony (Playstation) i Nintendo, a Google djeluje kao novi konkurent platformama sa svojom platformom za igranje u oblaku (eng. cloud-based gaming platform), Stadia. PC igre otvorile su svijet mogućnosti igračima koji traže dodatnu jednostavnost upotrebe i spretnost kada ležerno igraju (ili se natječu) u svojim omiljenim igrama. Sve ove konzole apsolutno pomiču granice tehnologije nudeći igračima vrhunsku grafiku, brzinu i procesorske moći.

Meteorski uspon gaming industrije tek započinje. Današnji je sektor video igara spreman izazvati bilo koju drugu tehnologiju za titulu najinovativnije industrije. Ono što gaming čini tako posebnim jest da nije samo impresivan i zabavan, već je i uključiv i ekspanzivan. Za gotovo svakog čovjeka postoji igra i način igranja.

3. 3D modeliranje

Proces stvaranja trodimenzionalnih (3D) stvarnih ili imaginarnih objekata naziva se 3D modeliranje. [1] Kroz proces 3D renderiranja, može se dobiti 2D slika 3D modela iz jedne perspektive ili kao alternativa, 3D model se može koristiti kao resurs u real-time grafičkoj simulaciji. [2]

Kao matematička reprezentacija, 3D modeli su u svojoj srži, kolekcija podataka o točkama u 3D prostoru (eng. vertices) te drugih informacija koje zatim računalo interpretira u virtualni objekt koji se iscrtava na zaslonu. Osim stvaranja modela kroz korištenje specijalnih programa, moguće je kreirati modele kroz razne algoritme (proceduralno modeliranje) ili solucijom koja postaje sve popularnija, skeniranjem stvarnog objekta i interpretiranjem istog u računalu razumljiv format. Danas 3D grafika ima široku primjenu, osim gaming industrije, filmske i automobilske, 3D grafika se koristi i u medicini, znanosti i inženjeringu. [2]

3.1. Proces 3D modeliranja

Kod procesa 3D modeliranja razlikujemo:

- **Poligonalno modeliranje** – više točaka je izloženo u 3D prostoru te se one međusobno povežu ravnim linijama tvoreći tako poligon. Najmanji poligon je trokut, a izbor većine 3D umjetnika je poligon koji se sastoji od 4 točke.
- **Modeliranje krivuljama ili NURBS (Non-Uniformal Rational Bezier Splines) modeliranje** – koristi krivulje, koje za razliku od poligonalnog modeliranja, koriste kontrolne točke i nudi parametar „težine“. Pogodno je kod organskog modeliranja ili automobilske industrije.
- **Subdivizijsko modeliranje** – hibrid dvaju spomenutih tehnika

Tehnika ne određuje kvalitetu finalnog proizvoda, ali može znatno utjecati na vrijeme potrebno za stvaranje 3D modela. [1]

3.2. Terminologija 3D modeliranja za video igre

U nastavku slijedi kraći uvid u terminologiju 3D modeliranja za video igre.

Dodaci

Dodaci su sve što se mora kreirati u vanjskom softveru kao što je Blender, a mogu biti modeli, teksture, zvučni efekti, animacije i još mnogo toga.

Broj poligona modela

Prilikom modeliranja za igre najvažnije je uzeti u obzir polibroj ili broj poligona modela (eng. polycount), kojeg treba držati na četverokutima ili „Quads“ i trokutima ili „Triangles“. Polibroj modela je ukupan broj trokutastih poligona potrebnih za crtanje modela u 3D prostoru. Što je veći broj poligona, sustavu je potrebno više vremena da ga renderira.

Renderiranje u pravom vremenu

Očekuje se da će se igre prikazivati određenom brzinom zvanom Frames-Per-Second. Kad igrač pritisne tipku, očekuje da će se vezana akcija odmah prikazati.

LOD Modeli

Modeli razine detalja ili LOD (eng. Level of Detail) modeli su različite verzije modela razlučivosti igre koji se koristi. Ovi se modeli razlikuju u broju poligona i zamjenjuju na temelju udaljenosti modela od kamere. Predstavlja izuzetno važan aspekt optimizacije igara.

Teksturiranje – UV mapiranje

Uv mapiranje nužan je dio stvaranja prilagođenih tekstura za 3D modele. UV mape premošćuju jaz između 3D objekata i 2D tekstura, što omogućuje da se 3D objekt postavi u ravne UV koordinate kako bismo ih obojili detaljima.

Prilikom teksturiranja igre važno je u obzir uzeti razlučivost teksture i koje se vrste mapa moraju koristiti kako bi se dobio određeni izgled ili efekt.

4. Blender

Blender, besplatan 3D program otvorenog koda (eng. open-source), danas je izrazito popularan, a podržava cjelokupni tok (eng. pipeline) 3D-a – modeliranje, izradu kostura (eng. rigging), animaciju, simulaciju, renderiranje, stvaranje kompozicije (eng. compositing), praćenje pokreta (eng. motion tracking), uređivanje videa (eng. video editing) te 2D-a. Naprednim korisnicima Blenderovo okruženje omogućuje prilagodbu programa prema vlastitim potrebama pomoću Python programskog jezika i API-ja ili aplikacijskog programskog sučelja (eng. application programming interface). Osim što je besplatan, Blender je i „cross-platform", što znači da se može koristiti na različitim operacijskim sustavima kao što su to Windows, Macintosh te Linux. Prilagođen je individualcima i malim studijima koji pronalaze korist u njegovom jedinstvenom toku rada i responzivnom razvojnom procesu. Kreiran je od strane malih studija ili samostalnih umjetnika te programera, a prvo izdanje programa izašlo je 1998. godine. Zaštićen je GNU (eng. General Public License), tj. općom javnom licencom.

Cijeli je program otvorenog koda budući da je kreiran kao projekt koji vodi zajednica (eng. community-driven project), kako bi svaka osoba mogla napraviti male ili velike preinake u osnovnom kodu programa, s ciljem izrade programa koji nudi više mogućnosti te je dostupan svima. Ovaj pristup omogućuje kontinuiranu objavu novih značajki, ispravljanje pogrešaka (eng. bugs) i bolju upotrebu samog alata.

4.1. Povijest

3D alat kojeg danas poznajemo nastao je u Nizozemskoj. Ton Roosendaal, osnivatelj NeoGeo -a, najveće kuće 3D animacije u Nizozemskoj, odlučio je da se tadašnji 3D alat studija mora iznova napisati. Taj je proces započet 1995. godine. 1998. godine Ton je osnovao novu tvrtku Not a Number (NaN) koja se posvetila daljnjem razvoju Blendera. Cilj njegove tvrtke bio je razvoj besplatnog online alata za kreiranje interaktivnog 3D sadržaja te komercijalne verzije softvera za distribuciju i objavljivanje. Zbog razočaravajućih rezultata prodaje i tadašnje teže gospodarske situacije, investitori NaN-a početkom 2000. godine isključuju sve operacije tvrtke, a što je uključivalo i obustavu razvoja Blendera. Ipak, podrška korisničke zajednice i kupaca nije mogla dozvoliti da ovaj alat padne u zaborav. U svibnju iste te godine, Roosendaal pokreće neprofitnu zakladu za Blender ili The Blender Foundation. Prvi je cilj zaklade bio pronaći način za daljnji razvoj i promociju Blendera kao projekta otvorenog koda temeljenog na zajednici. U srpnju Roosendaal uspijeva navesti investitore NaN-a na realizaciju jedinstvenog plana fondacije Blendera. Kampanja „Besplatni Blender" u kratkom roku od samo 7 tjedana postiže cilj i uspijeva prikupiti 100.000€. U nedjelju, 13. listopada 2002. godine Blender je pušten u svijet pod GNU

licencom. Od tog dana, razvoj Blendera kontinuirano se nastavlja, a postaje alatom vođenim od strane predanih volontera iz cijelog svijeta, na čelu s izvornim tvorcem Roosendaalom.

4.2. Blender zajednica

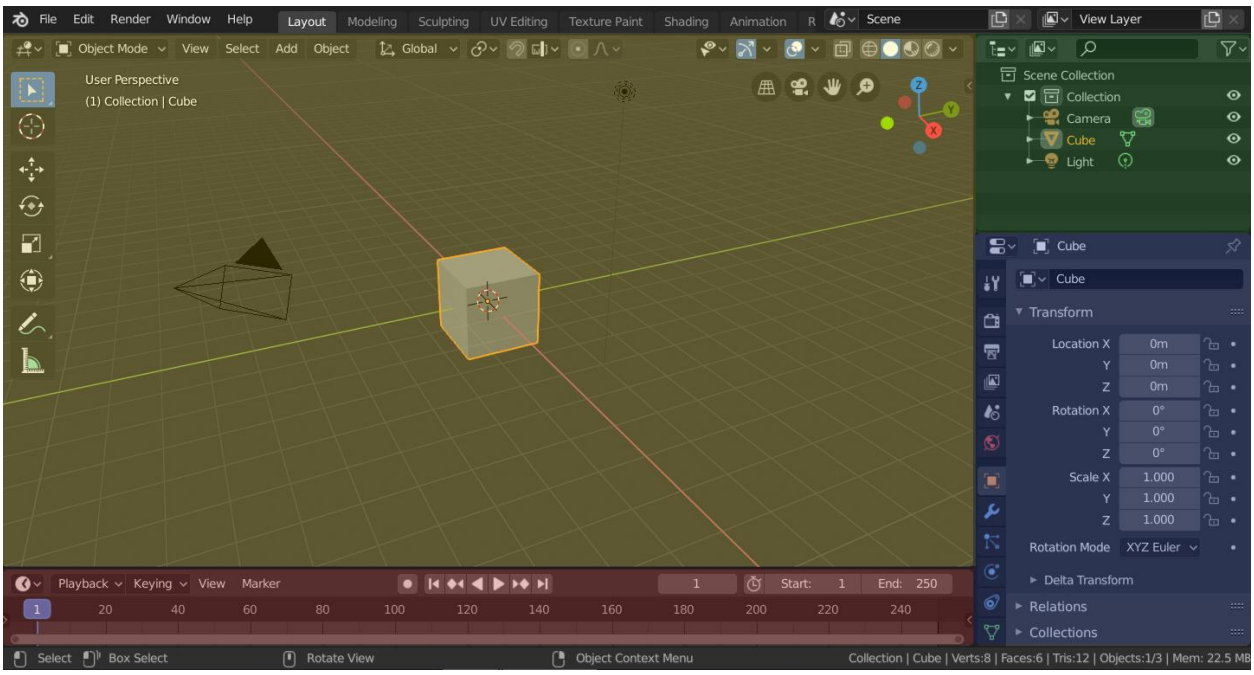
Preko 15 godina Blender predstavlja odličan primjer javnog i društvenog projekta koji milijunima korisnika omogućuje stvaranje 3D sadržaja pomoću besplatnog alata. Na usvajanje Blendera od strane zajednice uvelike je utjecalo što je besplatno dostupan od samih početaka. Otvoreni kod i pristupačnost pridonijeli su tome da ga je sve više ljudi počelo koristiti. Stoga ne čudi da se oko Blendera okupila velika, stabilna i aktivna zajednica korisnika koja je i pokretačka sila razvoja ovog moćnog alata. Nekoliko je neovisnih web stranica kao što su forumi, blogovi, vijesti i web stranice s raznim vodičima, a sve to posvećeno je upravo Blenderu. Jedan od najvećih foruma zajednice jest Blender Artists, gdje se ljubitelji Blendera okupljaju kako bi pokazali svoje kreacije, dobili povratne informacije, zatražili i ponudili pomoć, itd. Osim web stranica, postoje i posebni kanali u zasebnim programima kao što je Discord, a na kojima se može razgovarati uživo s drugim korisnicima Blendera. Blenderova zajednica jedna je od njegovih najvećih karakteristika.

4.3. Sučelje

Grafičko korisničko sučelje ili GUI (eng. graphical user interface) jest način interakcije čovjeka s računalom, kroz manipulaciju grafičkim elementima i dodacima uz pomoć tekstualnih poruka i obavijesti. [4] Kod prvog pokretanja sučelja Blendera, prikazuje se tzv. „splash screen“ odnosno uvodni prozor sa prikazom trenutne verzije programa i mogućnostima kao što su kreiranje novog ili otvaranje nekog prethodnog projekta. Interakcija korisnika s Blenderom svodi se na unos naredbi pomoću tipkovnice i interakciji po sučelju klikom miša.

Slika 4.1 Uvodni prozor Blendera

Budući da je Blenderov GUI konzistentno na svim platformama, nakon pokretanja programa i zatvaranja početnog zaslona, prozor Blendera izgleda kao na slici koja slijedi:



Slika 4.2 Početni zaslon Blendera

Sučelje je podijeljeno na šest glavnih dijelova; zadani radni prostor (eng. default workspace) na samom vrhu, središnji 3D pogled (eng. 3D viewport), vremenska crta (eng. timeline), statusna traka na samom dnu, popis svih objekata u sceni (eng. Outliner) i vrsta urednika (eng. editor type) s raznim opcijama kao što su svojstva za render, scenu, svojstva objekata itd.

Svaki je dio sučelja u potpunosti prilagodljiv ovisno o potrebama korisnika. Moguće je promijeniti njihove veličine, dodati nove, ugasiti postojeće itd. Naravno, korisničke postavke ovog softvera također omogućuju apsolutno prilagođavanje njegovog UI-a i ostalih postavki.

4.4. Najčešće korišteni alati

Blender sadržava velik broj raznih alata za efektno 3D modeliranje. Neki od najčešće korištenih alata su sljedeći:

- **Scale** – omogućuje promjenu veličine objekta po svim osima te promjenu veličina stranica ili bridova objekta.
- **Move** – služi za micanje objekata, točaka, bridova ili stranica, ovisno o osi ili trenutnom pogledu na 3D prostor.
- **Rotate** – koristi se za rotaciju objekta, brida ili stranice, ovisno o osi ili trenutnom pogledu na 3D prostor.
- **Bevel** – koristi se za zaobljivanje oštrih rubova. Pomicanjem kotačića miša za vrijeme korištenja alata možemo povećati ili smanjiti broj segmenata.

- **Apply** – služi za primjenjivanje vrijednosti rotacije, veličine i položaja objekta unutar 3D prostora.
- **Extrude** – najčešće korišten alat koji duplicira vrhove zadržavajući pritom novu geometriju povezanu s izvornim vrhovima. Ekstruzija prema van „nadopunjava“ postojeću geometriju, dok ju ona prema unutra „umeće“.
- **Loop cut** – koristi se za kreiranje novih bridova preko površine odabranog objekta.
- **Delete** – alat pomoću kojeg se mogu brisati objekti, točke, bridovi ili stranice.
- **Make Edge/Face** – služi za kreiranje novih stranica ili bridova među točkama.
- **Fill** – služi za popunjavanje praznog prostora.
- **Duplicate** – omogućuje dupliciranje objekata, točaka, bridova ili stranica. [3]

4.5. Prednosti i nedostaci

Zbog svog širokog spektra mogućnosti Blender se danas može usporediti s mnogim komercijalnim 3D softverima. No u vrijeme Blendera prije verzije 2.80, njegovi su alati bili složeniji i teži za upotrebu, a mogućnost stvaranja kvalitetnih 3D modela i animacija je bila ograničena. S dolaskom verzije 2.80, Blender je postao sličniji ostalim profesionalnim alatima kao što je 3DSMax. Ipak, ovaj alat ne može u potpunosti zamijeniti najpoznatije plaćene opcije poput Maye, ZBrush-a ili 3DSMax-a, koji su već dulje u kontinuiranom razvoju i iza sebe imaju bazu korisnika koji se uglavnom sastoje od profesionalnih kreatora i umjetnika. Međutim, s 2.80 Blender novim i entuzijastičnim korisnicima, tvrtkama za razvoj igara i CG umjetnicima, donosi pravu priliku za ulazak u svijet 3D-a. Prednosti i nedostaci alata ukratko će biti prikazani u sljedećoj tablici:

Prednosti	Nedostaci
Besplatan, open-source alat	Zbog 100% prilagodljivog UI-a, mnogi „povremeni“ korisnici se teže u njemu snalaze.
Nudi brojne značajke; modeliranje, animacija, simulacija fluida, UV mapiranje, izrada kostura, praćenje pokreta, VFX…	Strmija krivulja učenja (učenje iznova i navikavanje na potpuno novo sučelje)
Blender datoteke mogu se izvesti u različite formate	Nedostaje vanjska podrška za druge programe
Moguće je renderirati preko GPU umjesto CPU (uvelike ubrzava renderiranje)	Nije industrijski standard

Brzi alati za modeliranje	Nije specijaliziran za određeni segment
Jednostavan i intuitivan UI	

Tablica 4.1 Prednosti i nedostaci alata Blender

Softver koji je najbliži Blenderu bio bi softver sposoban za izvođenje cijelog CG cjevovoda (eng. pipeline), kao što su spomenuti Maya, 3DS Max ili Cinema 4D. Svi ovi programi imaju svoje prednosti i nedostatke. Na kraju se odluka oko odabira softvera ipak temelji na krajnjem cilju korisnika.

5. Unity

Unity game engine je program koji se koristi za razvoj video igara za PC, mobilne uređaje i računalnu grafiku, razvijen od strane Unity Technologies. Game engine je softver koji kreatorima video igara daje potrebnu skupinu značajki za jednostavan, učinkovit i brz razvoj. Unity okuplja nekoliko osnovnih područja; ubacivanje 2D i 3D umjetničkih radova i dodataka iz drugih softvera kao što su Maya, Photoshop i Blender, sastavljanje tih dodataka u scenu i okruženja, dodavanje rasvjete, zvuka, posebnih efekata, fizike i animacije, interaktivnosti i logike igranja, uređivanje, uklanjanje pogrešaka i optimiziranje sadržaja za ciljane platforme.

Moćan je i jednostavan alat koji veliki broj programera igara koristi kako bi kreirali i ojačali svoje kreacije. Radi se o profesionalnom alatu kojeg koriste i neka od najvećih imena u industriji video igara. Neki od naslova razvijenih u Unity-u uključuju:

- Temple Run 1,2 (2011, 2013)
- Angry Birds (2014)
- Ori and the Blind Forest / Will of Wisps (2015)
- Super Mario Run (2016)
- Inside (2016)
- Rick and Morty: Virtual Rick-ality (2017)
- …

Unity podržava razvoj za svaku značajnu platformu; za mobilne uređaje, računala, konzole, Web, VR i AR. Osim što se koristi za razvijanje igara, koristi se i za razvijanje drugih aplikacija, reklama, interaktivnih priča i umjetničkih djela, vizualizacija, filmova, itd.

Ono što ovaj alat čini još moćnijim jest to što uključuje trgovinu dodataka (eng. asset store) – mjesto na kojem programeri i kreativci mogu prenijeti svoje kreacije i postaviti ih dostupnima za zajednicu.

Osim što je game engine, Unity je i IDE (eng. integrated development environment), integrirano razvojno okruženje koje opisuje korisničko sučelje koje na jednom mjestu daje pristup svim alatima potrebnim za razvoj. Softver ima vizualni uređivač koji omogućuje korisnicima da jednostavno povlače i ispuštaju (eng. drag and drop) elemente u scenu te zatim manipuliraju njihovim svojstvima.

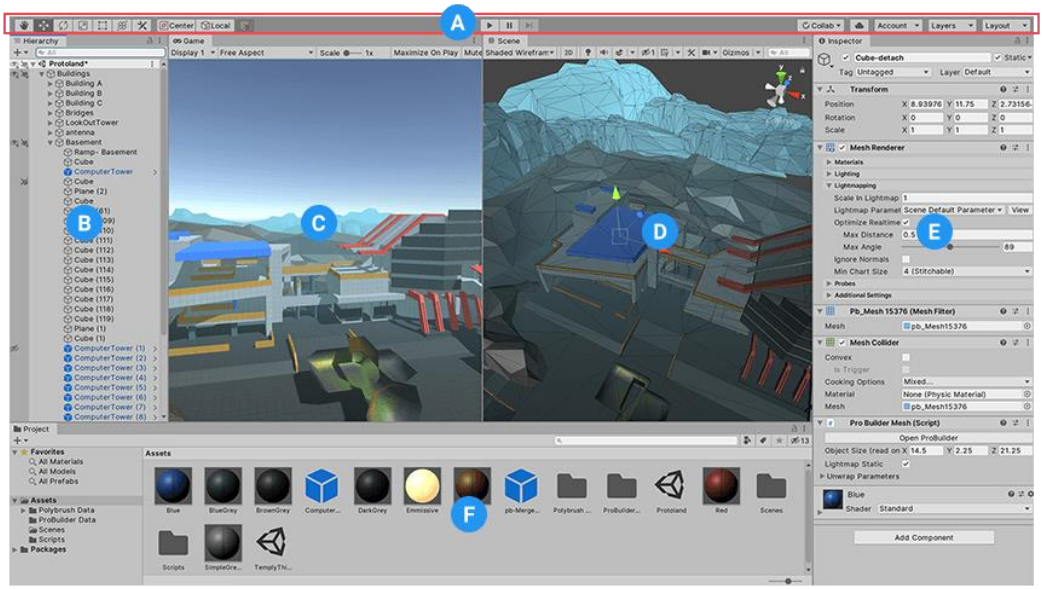
Također, softver nudi i mnoštvo drugih korisnih značajki i alata poput mogućnosti navigacije kroz mape u projektu ili stvaranje animacija putem alata za vremensku traku (eng. timeline). Kad je u pitanju kodiranje, Unity će se prebaciti na alternativni uređivač po izboru korisnika.

5.1. Povijest

David Helgason, Nicholas Francis i Joachim Ante osnovali su Unity Technologies u Copenhagenu 2004. godine, s ciljem kreiranja game engine-a za ostale developere. Prvotno je bio najavljen samo za platformu OS X, no 2005. godine razvijen je i za ostalo računalno i mobilno tržište (iOS, Android, Windows, Playstation, Xbox One, Mac itd.). Predstavljanjem Iphone-a 2007. godine, Unity Technologies naglo dobiva na značenju obzirom da je razvio jedan od prvih game engine-a koji u potpunosti podržava OS X platformu. Unity Asset Store pokrenuta je u studenom 2010. godine kao online tržište za korisnike Unity-a kako bi međusobno dijelili dodatke, bilo to besplatno ili pod nekom određenom cijenom. 2013. godine Facebook odlučuje integrirati komplet za razvoj softvera za igre koje koriste Unity. Komplet je sadržavao alate koji su omogućavali praćenje reklamnih kampanja i dubinsko povezivanje, gdje su korisnici bili izvorno povezani s objavama s društvenih mreža na određene dijelove u igrama, a bili su i u mogućnosti dijeliti slike unutar samih igara. 2016. godine Facebook razvija novu platformu za PC igre. Korištenje ovog softvera proširilo se izvan područja video igara, uključujući tako i film, televiziju i automobilsku industriju. Za automobilsku industriju proizvođači automobila koriste Unity-ovu platformu za virtualnu stvarnost za dizajn i simulacije testiranja automobila u virtualnom svijetu. [6] Danas se Unity trudi postati glavni pokretač za izradu bilo čega u interaktivnom 3D-u.

5.2. Sučelje

U nastavku slijedi opis glavnih dijelova Unity korisničkog sučelja.



Slika 5.1 Korisničko sučelje alata Unity

- **Traka alata (eng. tool bar)** – omogućuje pristup najvažnijim radnim značajkama; osnovni alati za manipulaciju prikaza scene i objekata igre u njoj, kontrole reprodukcije, pauze i zaustavljanja (eng. play, pause, stop), gumbi za pristup Unity Collaborate, Unity Cloud Services i vlastitom Unity računu te izbornik vidljivosti sloja (eng. layer visibility menu) i izbornik za prozore uređivača.
- **Prozor hijerarhija (eng. hierarchy window)** – hijerarhijski tekstualni prikaz svakog game objekta u sceni. Svaka stavka u sceni ima unos u hijerarhiju, tako da su ta dva prozora inherentno povezana.
- **Prikaz igre (eng. game view)** – simulira kako će izgledati konačna prikazana igra kroz postavljene scenske kamere. Simulacija započinje likom na gumb „Reproduciraj".
- **Prikaz scene (eng. scene view)** – omogućuje vizualnu navigaciju i uređivanje scene. Može prikazati 3D ili 2D perspektivu ovisno o vrsti projekta.
- **Prozor inspektora (eng. inspector window)** – omogućuje pregled i uređivanje svih svojstava trenutno odabranog game objekta. Izgled i sadržaj inspektora mijenja se ovisno o game objektu koji mogu imati različite skupove svojstava.
- **Prozor projekta (eng. project window)** – prikazuje biblioteku dodataka koja su dostupna za korištenje u projektu.
- **Konzola** – nalazi se na samom dnu prozora te predstavlja mjesto na kojem se mogu vidjeti upozorenja o mogućim pogreškama [7]

5.3. Prednosti i nedostaci

Unity je suočen s jakom konkurencijom poput Unreal Engine-a i Cryengine-a. No postaje idealnim razvojnim sredstvom kad je u pitanju razvoj igara za mobilne uređaje. Iako je softver nekada nosio ime Unity 3D, postao je jednako sposobnim alatom i za 2D. I ne samo to, način na koji se rukuje grafikom olakšava prijenos iskustva na niži hardver. Obzirom da je Unity cross-platforma, jednako je lako kreirati igre za Ios, PC i igraće konzole, a također nudi i izvrsnu VR podršku. U usporedbi s Unreal-om ili Cryengine-om, Unity nije toliko sposoban za top-end grafiku. Unreal i Cryengine imaju znatno strmiju krivulju učenja te su značajno manje dobrodošli za nove korisnike. Neke prednosti i nedostaci alata biti će prikazani u sljedećoj tablici:

Prednosti	Nedostaci
Izvrstan za cross-platform development	Ne sadrži predložak (eng. template), već je kod svakog projekta potrebno krenuti „od nule".

Učinkovita tehnička podrška	Visoka cijena pretplate, posebno za potrebe razvoja mobilnih igra
User-friendly	Zaostaje po pitanju uređivanja svijeta (eng. world editing), posebice u usporedbi s alatom kao što je Unreal Engine
Jednostavna istraga bugova i dotjerivanje (eng. tweaking)	Za neke korisnike bi UI mogao biti kompleksan
Unity Asset Store (brojni korisni dodaci i sredstva za razvoj)	Nije idealan za kompleksne projekte, poput nekih AAA igara.
Izvrsna grafika za optimalno gaming iskustvo	Česta ažuriranja koja mogu zakomplicirati korištenje alata.
Kreiranje igara bez potrebe za kodiranjem	Velika veličina datoteke za razvoj mobilnih igara.
Izvrstan game engine za VR	Nepristupačnost izvornom kodu što otežava otkrivanje i rješavanje mogućih problema.

Tablica 5.1 Prednosti i nedostaci alata Unity

Na kraju, Unity je i dalje vrlo moćan game engine, a uz dobre vještine kodiranja i kreativan um, s ovim se alatom zaista mogu napraviti čuda.

5.4. Instalacija i pokretanje

Preuzimanje i instalacija ovog alata vrlo je jednostavna. Da bi preuzimanje započelo, potrebno je posjetiti stranicu za preuzimanje Unity-a, gdje se nalazi Unity Hub poveznica za preuzimanje – upravitelj preuzimanja koji omogućuje upravljanje različitim verzijama Unity softvera, kao i svim dodatnim značajkama koje će možda zatrebati u procesu. U ovom je koraku potrebno izraditi profil, te nakon preuzimanja Unity Hub-a slijedi odabir željene verzije. Instalacijski program zatim provodi kroz nekoliko jednostavnih koraka. Za potrebe ovog rada korištena je najnovija verzija, 2020.1.4f1.

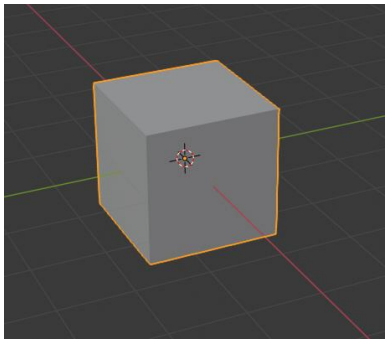
6. Praktični dio – izrada nivoa za video igru

U ovom će se poglavlju pojasniti proces izrade modela u Blenderu te zatim i postavljanje i kreiranje scene u Unity-u kako bi se kompletirao zadatak izrade nivoa za video igru. Za potrebe rada pomoću Blendera izmodelirana su četiri dodatka; mač, škrinja i vaze. Obzirom da bi pojašnjenje izrade i koraka kod svakog od ovih modela bilo podulje te ima zajedničkih koraka, u nastavku će se pojasniti modeliranje mača te UV mapiranje i teksturiranje vaze. Teksturiranje je iz znatiželje odrađeno i u Blenderu i u Unityu.

Za kreiranje okruženja u Unity-u izmodelirani dodaci sastavljeni su u cjelinu, te zajedno sa besplatnim dodacima koji su preuzeti s Unity Store-a, sastavljeno je okruženje srednjovjekovne tematike.

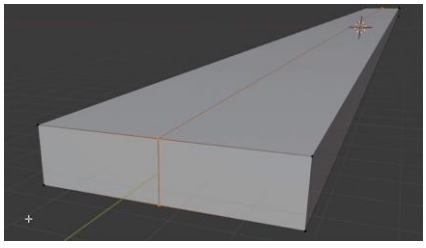
6.1. Modeliranje u Blenderu

Prvi je korak kreiranje jednog od osnovnih geometrijskih oblika; kocka, kugla, cilindar i drugi. U slučaju izrade mača, za početak počinjemo s kockom.



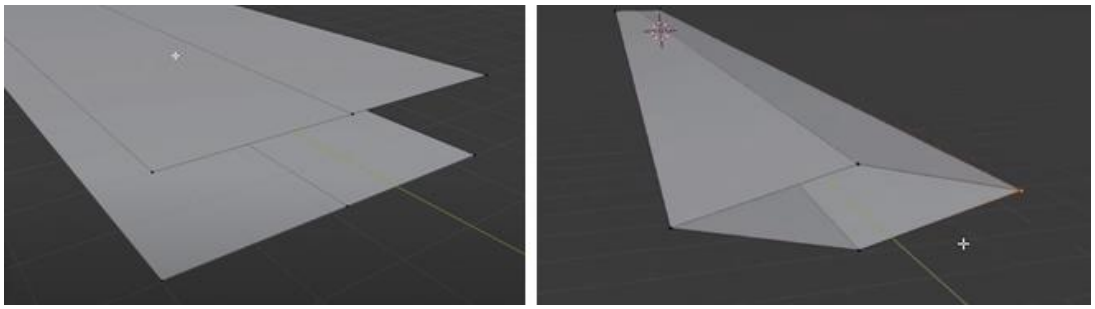
Slika 6.1 Kocka – jedan od osnovnih geometrijskih oblika

Slijedi manipuliranje veličinom, pozicijom i rotacijom tijela kocke. Taj dio odrađujemo u *Object mode*. Nakon dobivanja željenog oblika, uz pomoć alata *Loop cut* namješten je rez na sredini objekta, što će biti središte mača.



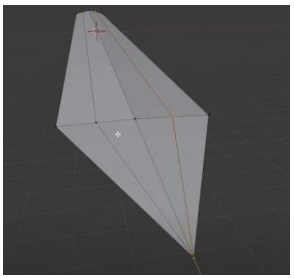
Slika 6.2 Demonstracija alata Loop cut

Sljedeće što je potrebno jest brisanje svih stranica na stranama objekta čime ćemo „zavariti“ njegove vrhove i dno. Potrebno je prebaciti se u *Edit Mode*, a stranice označujemo pomoću alata *Face select*, a brišemo pomoću tipke *Delete* koja otvara padajući izbornik na kojem odabiremo *Faces*. Da bismo spojili vrhove, potrebno je aktivirati alat *Vertex select*, označiti željene točke te kliknuti na izbornik *Vertex > Merge Vertices > At center*. Za špicu mača spajamo samo vanjske točke, dok na kraju mača spajamo sve.



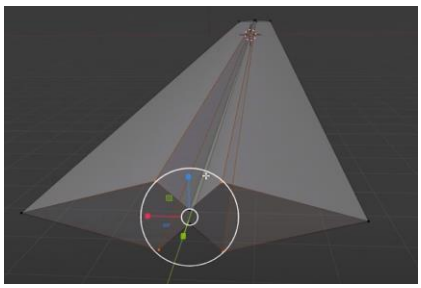
Slika 6.3 Demonstracija manipuliranja točkama

Zatim pomoću *Loop cut* namještamo novi rez blizu špice oštrice, putem *Selection* označujemo sve tri točke te pomoću *Scale* sve točke preslikamo na 0 na Z osi, što vrlo lako možemo pritiskom tipke 0. Dobili smo špicu oštrice mača.



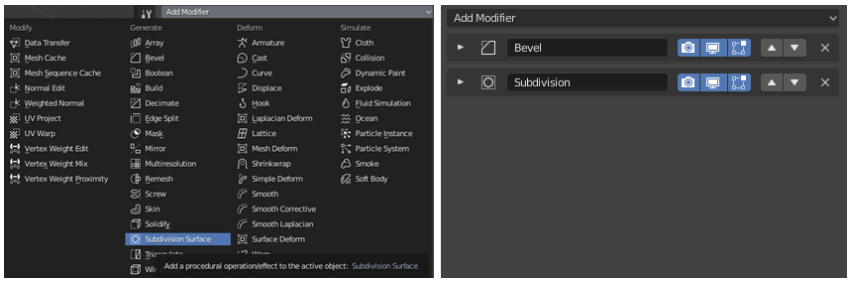
Slika 6.4 Prikaz dobivene špice oštrice mača

S *Loop cut* namještamo još dva reza, a točke dodatno prilagođavamo. U ovom koraku zgodnije je iskoristiti ili *Wireframe View* ili *Viewport Shading* kako bismo uspješno označili i točke sa donje strane objekta. Ti nam prikazi omogućuju da vidimo „kroz“ objekt. Koristeći iste načine, prilagođavamo i početak oštrice koji nakon prilagodbe izgleda ovako:



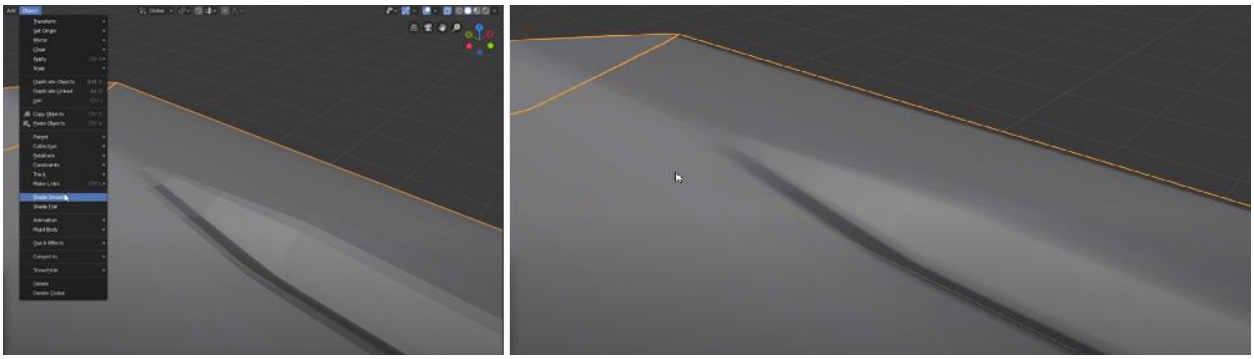
Slika 6.5 Prilagodba ostatka oštrice mača

Slijedi daljnja prilagodba objekta koristeći iznad navedene alate i načine. Time smo dobili oštricu koja je za sada previše poligonalna. Kako bismo izgladili objekt, vraćamo se u *Object mode* u kojem ćemo mu klikom na *Modifier Properties* dodati *Bevel* i *Subdivision Surface*, s time da *Bevel* postavljamo iznad *Subdivision Surface* kao što je na slici ispod, kako ne bi bilo previše izglađeno.

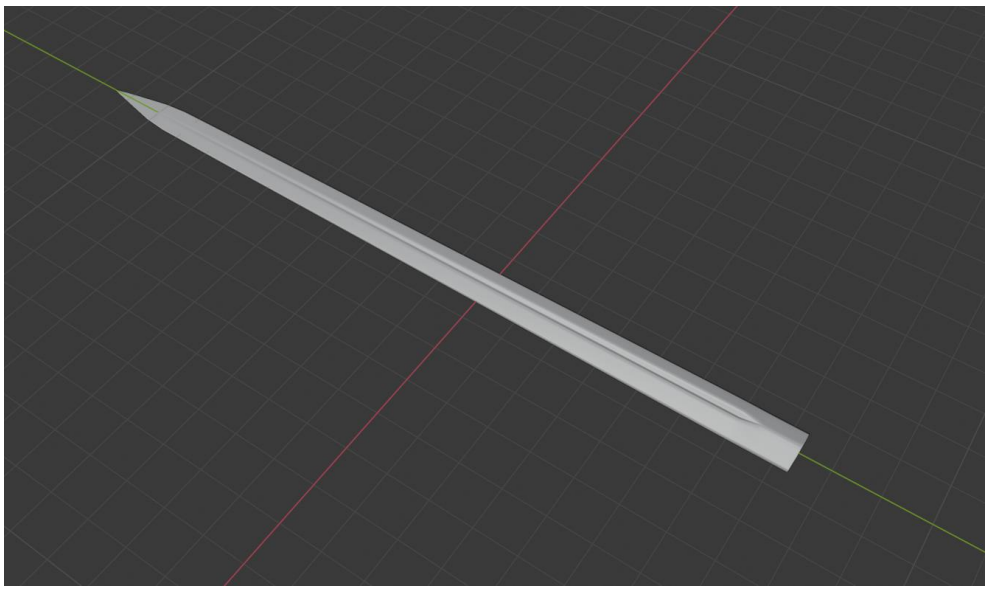


Slika 6.6 Opcije Bevel i Subdivision Surface

Za dodatno zaglađivanje pojedinih dijelova koristi se opcija *Shade Smooth*. Njoj pristupamo kao što je to prikazano na slici ispod. Time je izmodelirana oštrica mača.

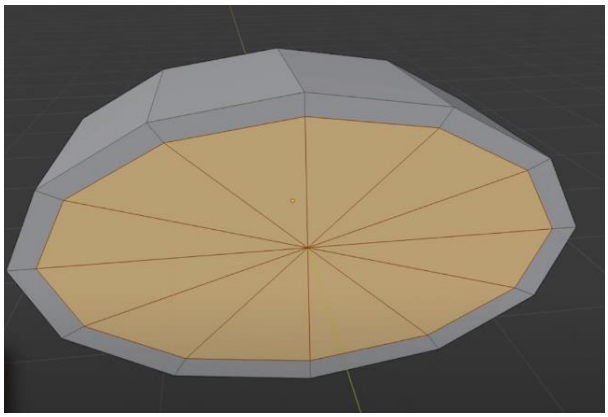


Slika 6.7 Opcija Shade Smooth



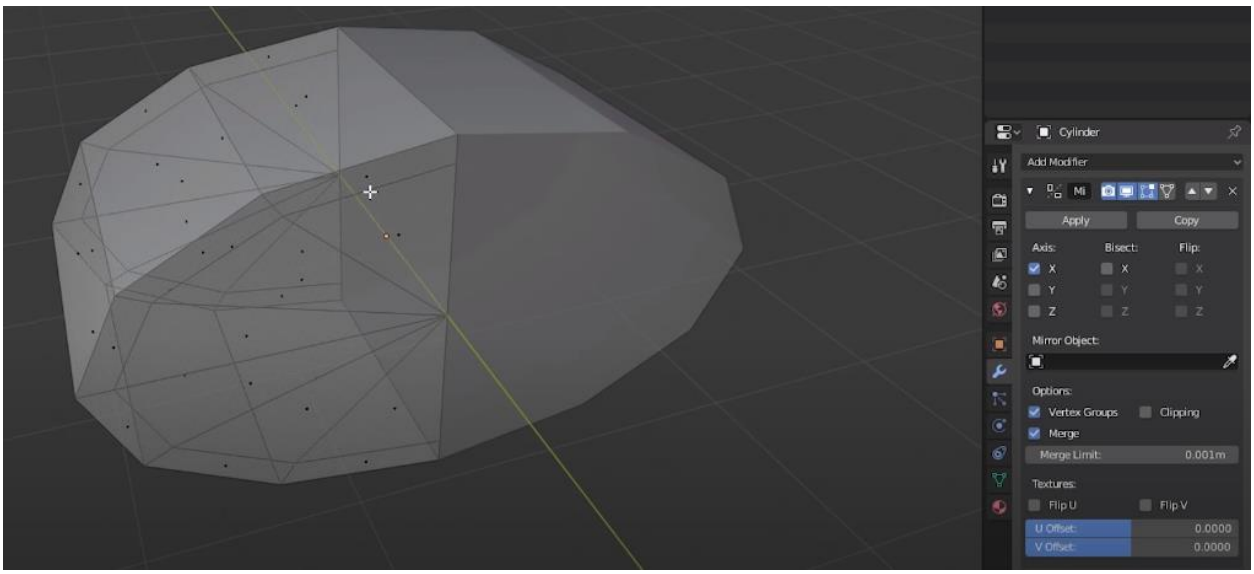
Slika 6.8 Finalan izgled izmodelirane oštrice mača

Ostatak mača izrađen je istim alatima i načinima, ali i uz korištenje drugih alata za modeliranje. Tako je za branik (eng. guard) mača, uz korištenje cilindra, korišten i alat *Extrude* kako bi se popravila topologija objekta. Selektirajući gornje i donje stranice, pomoću *Extrude Region* i klikom na plus koji se pojavi aktiviranjem alata, kreirali smo još poligona koje zatim prilagođavamo sa *Scale*. Time smo dobili još jedan *Edge Loop*.



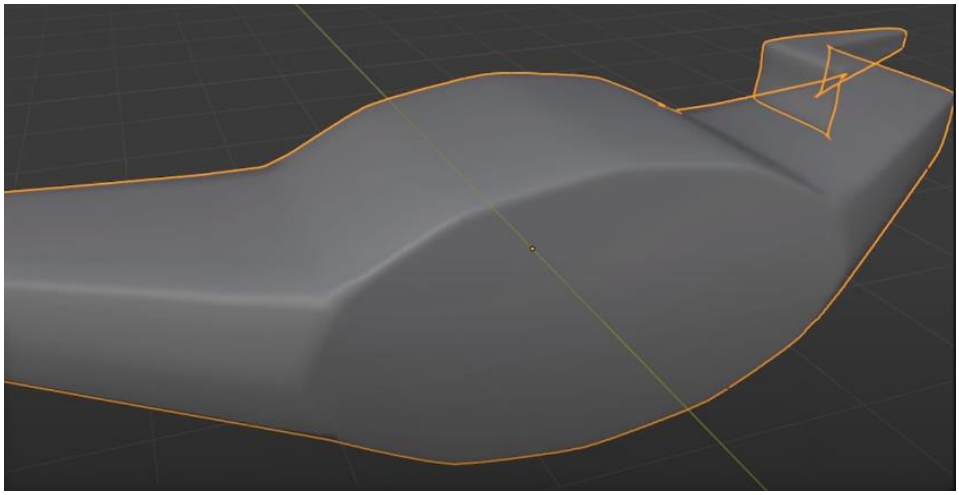
Slika 6.9 Rezultat Extrude Region alata

Kod modeliranja je uobičajena praksa da se modelira polovica objekta te se to preslika (eng. mirror) na drugu polovicu. Oštrica mača jednostavniji je model pa taj korak nije bio potreban, no kad modeliranja branika će se ipak primjeniti. Nepotrebnu polovicu brišemo te klikom na *Modifier Properties* > *Add Modifier*, dodajemo opciju *Mirror*.



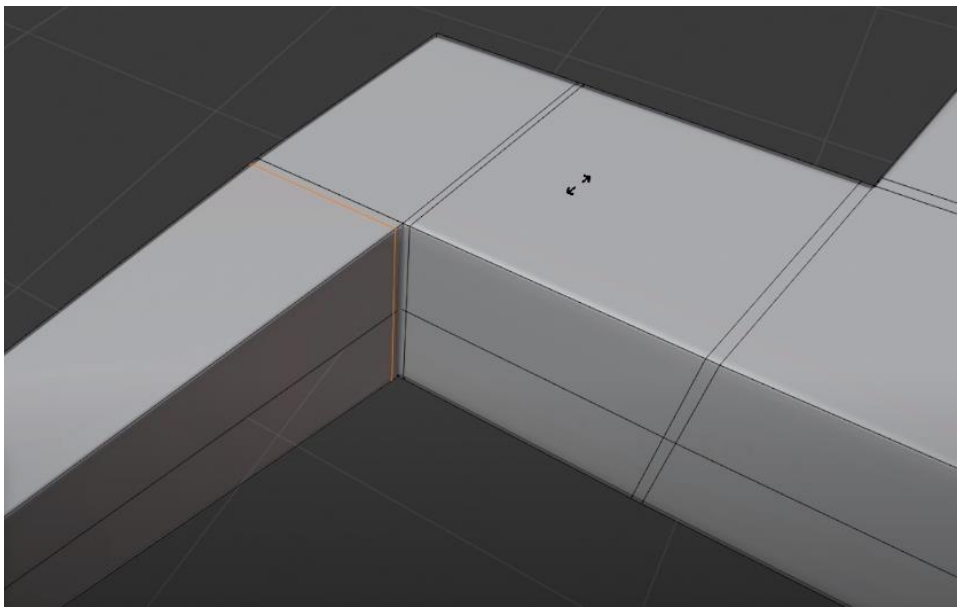
Slika 6.10 Dodavanje opcije preslikavanja

Slijedi prilagodba glavnog dijela branika pomoću već korištenih alata *Extrude Region*, *Scale*, *Move*, *Loop Cut*, te postavljanje modifiera *Bevel* i *Subdivision Surface* i apliciranje opcije *Shade Smooth*.

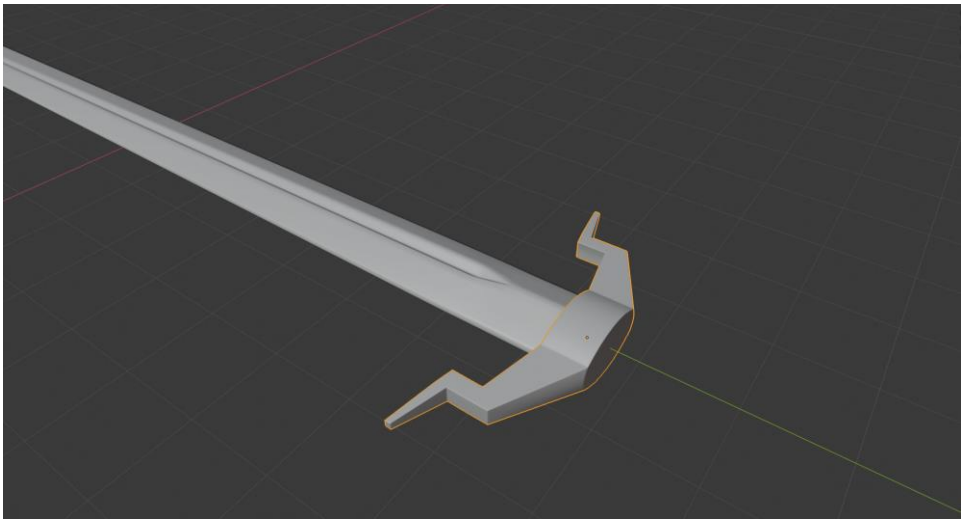


Slika 6.11 Nezadovoljavajući rezultati opcije Shade Smooth

Ponekad se dogodi da se naredbom *Shade Smooth* ne dobivaju zadovoljavajući rezultati te dolazi do potrebe dodatne prilagodbe modela. U ovom je slučaju pod opcijama *Modifier* > *Bevel* dovoljno dodati još jedan segment te limitirati metodu na neki kut (eng. angle), u ovom slučaju od 30°, što znači da se, u slučaju prelaska tog kuta, zaobljenje neće dodatno zaobliti. Eventualne dodatne poteškoće u topologiji modela mogu se počistiti putem *Loop Cut* alata.

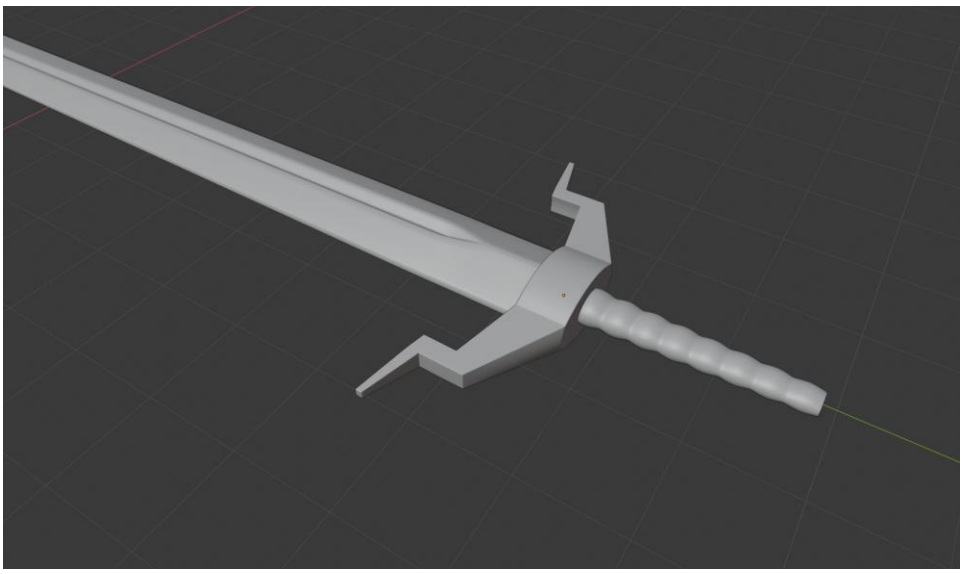


Slika 6.12 Izglađivanje nepravilnosti u topologiji objekta



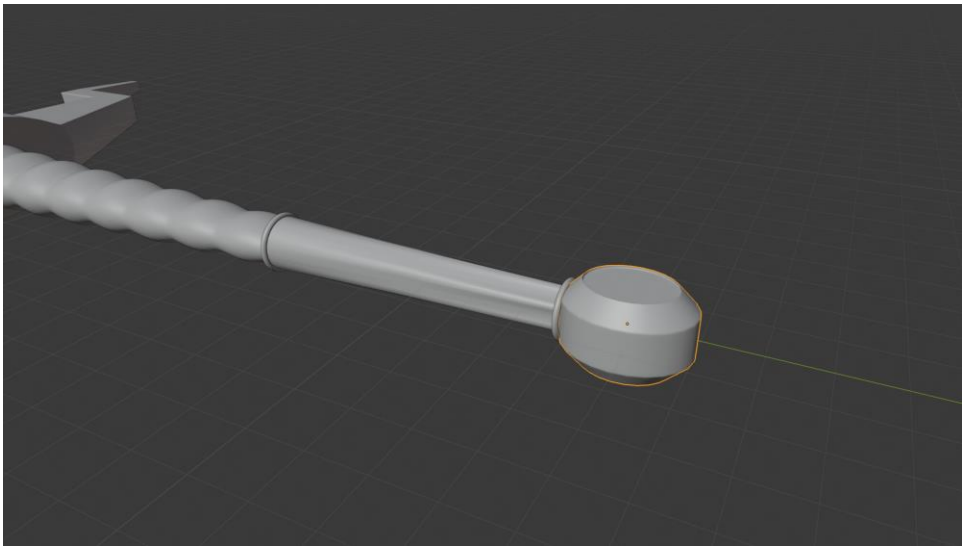
Slika 6.13 Finalan model branika mača

Slijedi modeliranje drške mača za koju je također odabran cilindar te je prilagođen već poznatim alatima. U ovom koraku koristiti će se još jedan alat, *Bevel*. Iako je prije spomenut u kontekstu *Modifier* alata, ovdje se ipak radi o alatu za poligone. Razlika je u tome što je *Bevel* kao alat za poligone destruktivan, što znači da je zapečen (eng. baked) u model, dok u kontekstu *Modifiera* nije destruktivan. Nakon postavljanja novih bridova na objektu putem *Loop cut*, selektiraju se svi bridovi pomoću naredbe *Select* > *Select Loops* > *Edge Loops*, Nakon selektiranja alata, klikom i povlačenjem se odredi željeni rezultat. Sljedeći je korak dodavanje novih bridova te njihovo selektiranje već poznatim korakom te prilagodba veličine. Da bi se sve izgladilo, ponavlja se dodavanje *Subdivison Surface* i *Shade Smooth*.



Slika 6.14 Finalizirana drška mača

Za dodatne detalje između drške i ostalih dijelova mača kao i sami njegov vrh koriste se cilindar, ali i novi objekt, *Torus* koji se zatim prilagođavaju već poznatim alatima, rotiraju i postavljaju na predviđene pozicije.



Slika 6.15 Finalizirani vrh mača

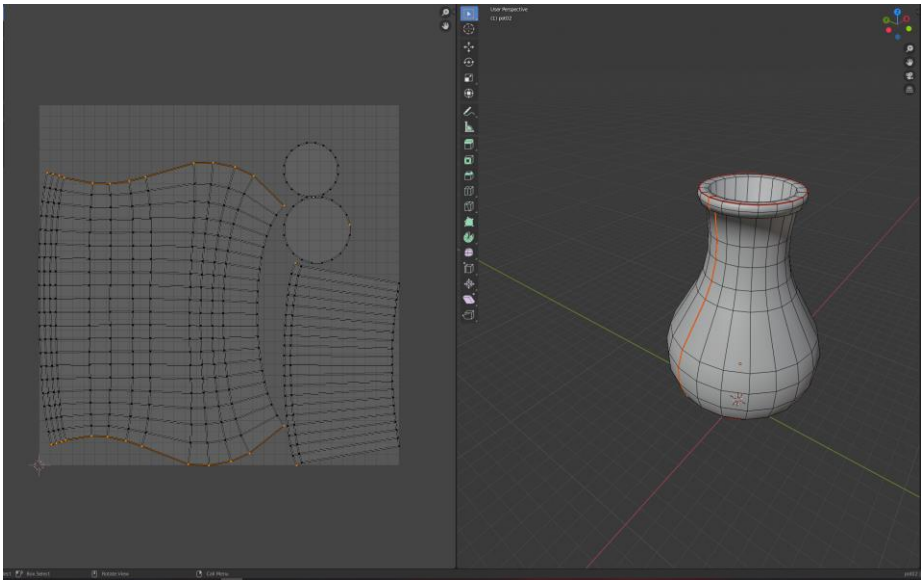
Slijedi UV mapiranje i teksturiranje modela, a koje će biti pojašnjeno kroz drugi izmodelirani dodatak, vazu.

6.2. UV mapiranje

Kod odmotavanja (eng. unwrapping) nekog objekta potrebno je na njega misliti u obliku dvije dimenzije. Vazu dakle trebamo podijeliti i namjestiti na područja 2D teksture. Ovaj se korak odrađuje u *UV Editing* prozoru. Prema zadanom, prilikom manipuliranja cilindričnih objekata, Blender odradi automatsko odmotavanje, ali ne uzima u obzir ništa od dodatnog uređivanja koje je primijenjeno na objektu. Dakle, preporučuje se da se taj korak ne koristi. Brz i jednostavan način da odmotamo UV mape jest označavanje cijelog objekta, pritisak tipke U te odabir opcije *Smart UV Project*. Iako je ovo doista brz i jednostavan način, ne daje uvijek najbolje rezultate.

Ponekad se može dogoditi da se prikaže pogreška „*Object has non-uniform scale*“, odnosno objekt ima neujednačeni *scale*. Ovdje je onda potrebno pritisnuti Ctrl + A, te u izborniku odabrati Scale.

Adekvatniji način za odmotavanje UV mapa jest definiranje gdje želimo da se oblici režu. Potrebno je biti u *Edit mode*, selektirati odabrani brid, te ga označiti kao privid (eng. seam) putem *Edge* > *Mark Seam*. Nakon toga je potrebno odmotati objekt, no koristeći opciju *Unwrap* umjesto *Smart UV Project*. Ukoliko nismo zadovoljni prikazom UV mapa, prijašnji korak ponavljamo dok ne dobijemo jasniju sliku.

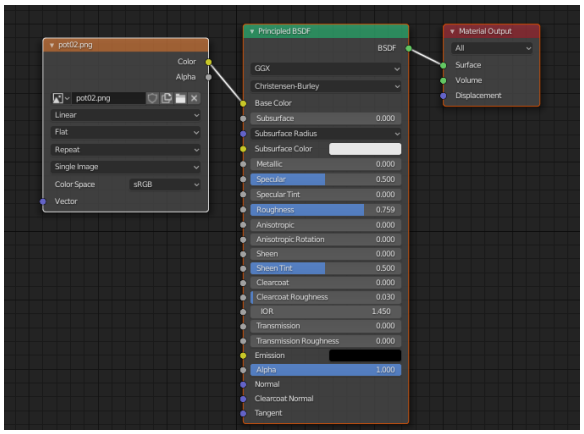


Slika 6.16 UV mapiranje modela vaze

6.3. Teksturiranje

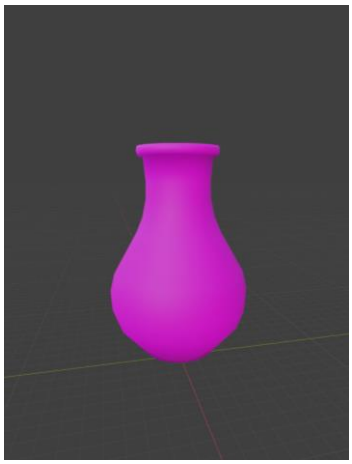
Teksture su za ovaj korak preuzete sa stranice *Textures.com*. Potrebno je biti u *Shading* prozoru. U slučaju dobre UV mape kao što je prikazana iznad, kod teksturiranja je dovoljno dodati sliku teksture ravno u *Shading Editor*, kreiranjem novog materijala klikom na *New* te putem *Add* >

Texture > *Image Texture* dodati teksturu. Zatim je *Color* potrebno prikvačiti za *Base Color* materijala, te putem *Open* otvoriti mapu s teksturama i odabrati željenu. Ovdje se mogu prilagođavati određena svojstva materijala. Ovo je i najjednostavniji korak, no za potrebe ovog rada odlučeno je teksture primijeniti putem opcije *Paint Texture*, za što realističniji doživljaj.



Slika 6.17 Prikaz čvorova (eng. node) u Shading Editoru

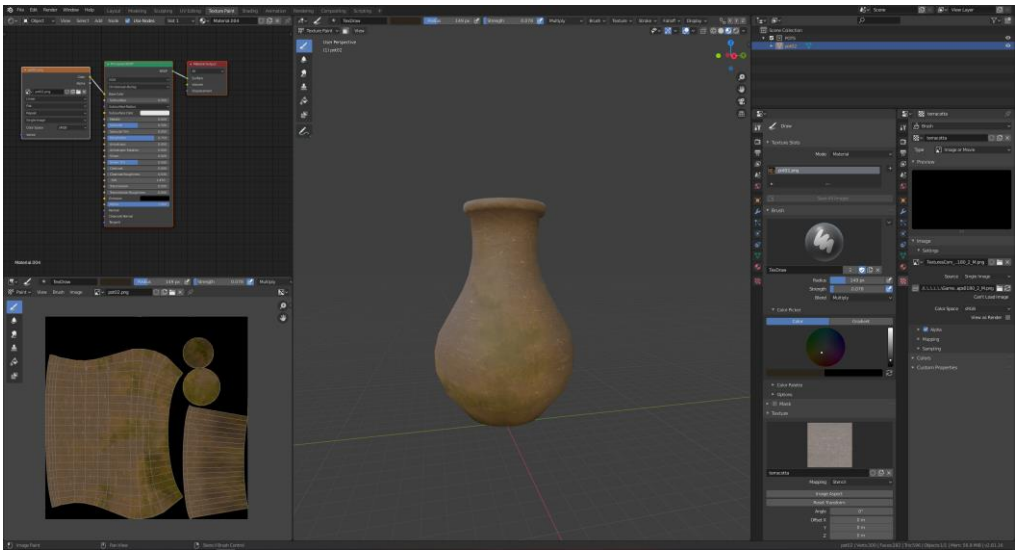
Putem opcije *Paint Texture* možemo kontrolirati gdje se nalaze i kako se ponašaju teksture na objektu. Ovdje se može dogoditi da je objekt ljubičaste boje te da po njemu ne možemo bojati, a javlja se i greška „*Missing Material, Textures detected*“.



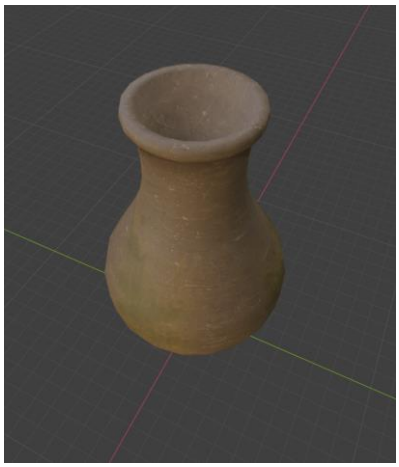
Slika 6.18 Model kojom nedostaju materijali i teksture

Potrebno je dodati novu teksturu pod *Active Tool and Workspace settings*, a gdje će u ovom slučaju biti prikazana informacija „*No Textures*“. Odmah po kreiranju teksture, u *Shading Editoru* pojavljuju se čvorovi za materijal i teksturu. Da bi se po objektu moglo bojati, potrebno je da mu normale gledaju prema gore. To se može provjeriti klikom na *Viewport Overlays* te označavanjem opcije *Face Direction*. ako je sa normalama sve u redu, objekt će biti plave boje. U slučaju da su

normale okrenute prema dolje, potrebno je prebaciti se u *Edit mode*, pritisnuti Shift + N što poziva opciju *Recalculate Normals*, gdje je potrebno označiti opciju *Inside*. Ono što je sada potrebno jest povlačenje teksture s kojom želimo bojati objekt. U *Active Tools and Workspace Settings* odabiremo opciju *Texture*. Zatim kreiramo novu teksturu s odabranom slikom klikom na *New*. Na kartici *Texture Properties* pod tip teksture biramo *Image or Movie*. Kod opcije *Texture* je za mapiranje zadano *Tiling*, što u slučaju ove teksture nije niti loša opcija, no za bolji efekt odabran je *Stencil*. *Stencil* se kontrolira desnim klikom i u kombinaciji s tipkama Shift i Ctrl. Slijedi obilaženje oko predmeta, bojanje teksture, prilagođavanje boje, efekata te zaglađivanje vidljivih prijelaza, bojanje novom teksturom za dodatni efekt te zatamnjenje unutarnjih područja.



Slika 6.19 Texture Paint prozor



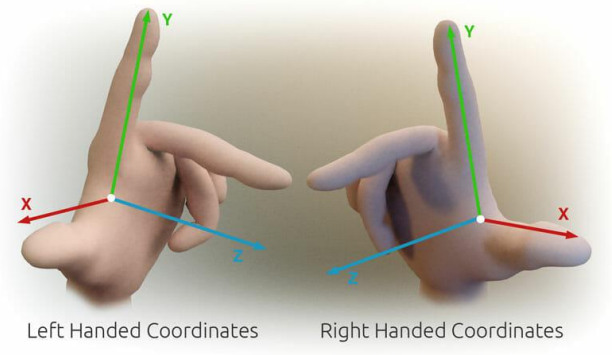
Slika 6.20 Teksturirani model vaze

Ne smijemo zaboraviti da unutar *UV Editor* prozora trebamo spremiti sliku teksture na za to predviđeno mjesto. Prilikom svake promjene koju radimo na objektu, na kartici *Image* se prikaže zvjezdica što nam također govori da je sliku, nakon svake promjene, potrebno spremiti.

6.4. Priprema i izvoz modela iz Blendera za Unity

Koordinatne razlike

Prvo što treba imati na umu jest da postoje dvije glavne razlike između koordinatnog sustava Unity-a i Blender-a. Blender koristi koordinatni sustav s desnicom (eng. right-handed coordinate system), dok Unity koristi sustav s ljevicom. U blenderu je Z os usmjerena prema gore, dok je u Unityu Y os usmjerena prema gore.



Postavke Blender modela

Osnovni koraci prije pokušaja bilo kojeg uvoza u Unity uključuju brisanje zadane kamere i svjetla u Blenderu, obzirom da je potreban samo model. Na panelu transformacije (eng. transforms) vrijednosti za rotaciju trebaju biti postavljene na 0, a za Scale na 1 na svim osima. To se postavlja jer u Unityu želimo biti u mogućnosti transformirati model, a u slučaju da imaju neobične vrijednosti, taj će nam korak biti otežan.

Pozicija ishodišta (eng. origin position)

Za poziciju ishodišta je dobro da je postavljena na značajnom mjestu, odnosno na osnovi modela.

Normale trebaju biti usmjerene prema van

Ponekad se prilikom uvoza modela u Unity može dogoditi da su neke stranice modela „nevidljive“. Ta je pogreška uzrokovana normalama onih stranica koje su okrenute prema unutra.

U *meshu*, svaka stranica ima jedan vektor koje je okomit na tu stranicu. Taj se vektor naziva normala i usmjerava na samo jednu stranu lica, a koristi se za *rendering*. U modelima normale bi

uvijek trebale biti usmjerene prema van, a to se može postići u *Edit* načinu, selektiranjem svih stranica te naredbom *Mesh* > *Normals* > *Recalculate Outside*.

Uvoz .blend ili .fbx datoteka

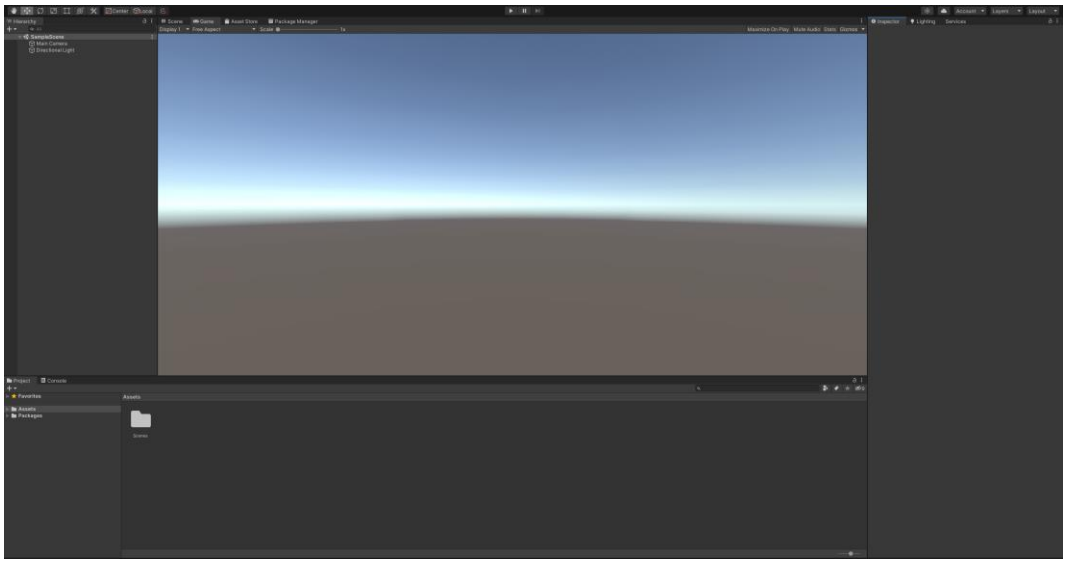
Dvije su glavne mogućnosti za uvoz datoteka iz Blendera u Unity; uvoz datoteke *.blend* izravno u Unity te izvoz *.fbx* datoteke iz Blendera, a zatim uvoz te datoteke u Unity.

Datoteku *.blend* jednostavno povlačimo u *Asset* mapu, a zatim i u scenu.

Za izvoz *.fbx* datoteke u Blenderu potrebno je ići na *File* > *Export* > *FBX (.fbx).* da bismo bili sigurni da ne izvozimo ništa što nije potrebno za projekt (kamera, svjetla, itd.) unutar prozora koji se otvorio potrebno je kod tipova objekta odabrati *Mesh*. Ovisno o tome što se želi izvesti, potrebno je označiti odgovarajući tip objekta. Obzirom da Unity i Blender imaju različite koordinatne sustave, potrebno je označiti *Apply Transform* kako bi koordinate transformacija funkcionirale za Unity. Zadnji je korak klik na gumb *Export* kako bi se model izveze u željenu lokaciju. Kao što je to opisano za .blend datoteke, i *.fbx* jednostavno povučemo u *Asset* mapu te zatim i u scenu.

6.5. Izrada scene u Unity-u

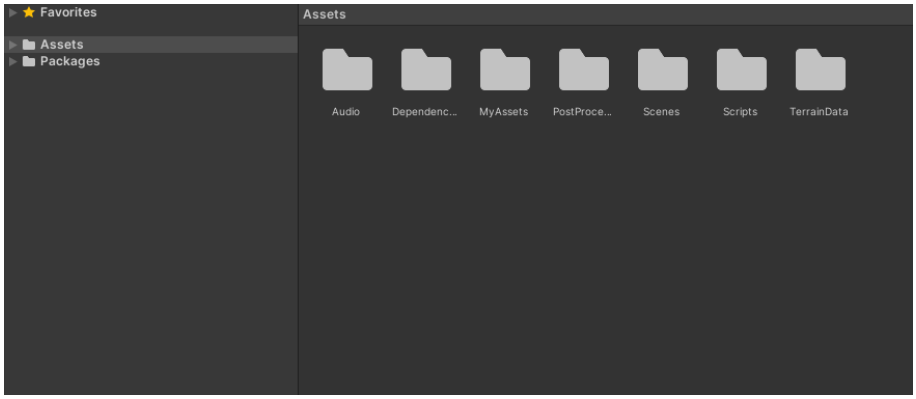
Kod prvog pokretanja aplikacije potrebno je kreirati projekt. Za potrebe ovog rada kreiran je 3D projekt te slijedi prikaz UI-a kao na slici ispod.



Slika 6.21 Početni prikaz novokreiranog projekta u Unity-u

Prije samog slaganja scene, pripremljeni su besplatni dodaci koji su preuzeti sa *Asset Store*-a, a koji će zajedno s izmodeliranim dodacima sačinjavati zamišljeni nivo video igre. Tako su odabrani dodaci kao što su vegetacija, kamenje, šator i efekti osvjetljenja. Preuzimanjem dodatka iz trgovine, iste je u *Packages Manager* potrebno preuzeti te zatim i uvesti klikom na *Import*. Da bi se uspješno izradila aplikacija, projekt mora sačinjavati jednu ili više scena koje se kasnije pretvaraju u aplikaciju. Aplikacija ovog rada sadržava jednu scenu odnosno jedan nivo (eng. level).

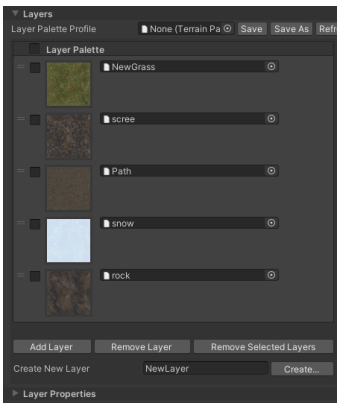
Slijedi priprema strukture projekta što predstavlja jedan od ključnih koraka, obzirom da projekt može postati vrlo opsežan te je potrebno sve logički posložiti. U prozoru projekta kreiraju se nove mape u koje će se ubaciti modeli iz Blendera, preuzeti dodaci, teksture i ostalo.



Slika 6.22 Struktura projekta

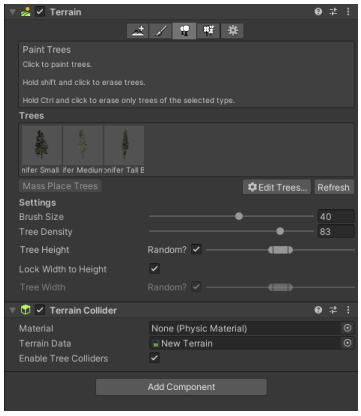
Da bismo na odgovarajući način prilagodili postavke terena, prvo ga je potrebno kreirati. U hijerarhiji objekata desnim klikom idemo na *3D Object > Terrain*. Prije izrade terena, pripremljena je tekstura trave koja će prekrivati cijeli teren, preuzeta s *Asset Store*-a, kao i dodatne teksture za detalje trave i cvijeća te stabla koja će sačinjavati šumu. Za potrebe izrade terena korišteno je nekoliko kistova (eng. brushes) uz dodatnu prilagodbu svakog. Za ovaj je projekt dodatno preuzet paket sa *Package Managera* imena *Terrain Tools* koji sadržava širi spektar kistova za izradu terena. Ovisno o željenom rezultatu, na pojedinom se kistu prilagođava jačina, veličina, rotacija i razmak.

Slijedi teksturiranje terena. Budući da su tereni veliki objekti, uobičajena je praksa koristiti teksturu koja se neprimjetno ponavlja i popločava preko površine. To je ponavljanje uglavnom neuočljivo s gledišta lika blizu tla. Teren se boja u slojevima. Svakom sloju dajemo ime i teksturu. Prvi sloj koji se postavi, biti će postavljen na cijeli teren.



Slika 6.23 Teksturiranje terena

Nakon što se postigne željeni izgled terena, po terenu se postavljanju već pripremljena stabla. Za početak, stabla se postavljaju na mjestu predviđenom za izradu glavne scene nivoa, oko koje će se zatim kasnije nastaviti popunjavati teren.

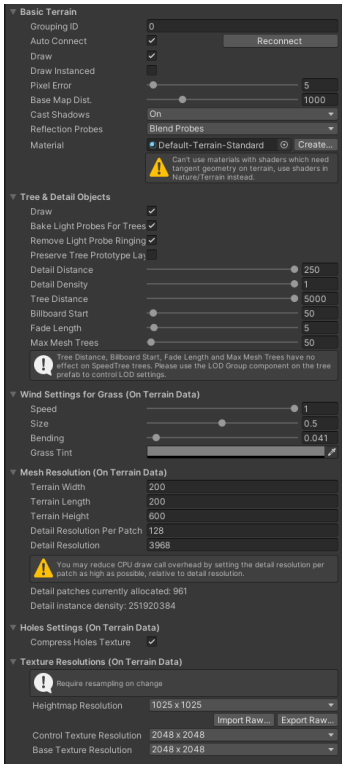


Slika 6.24 Opcija Paint Trees

Stabla su čvrsti 3D objekti koji rastu iz površine. Unity koristi optimizacije poput *billboarding* za udaljena stabla, kako bi se održali dobri performansi renderinga. To znači da projekt može imati guste šume s tisućama stabala, a da pritom zadrži prihvatljiv *frame rate*. Stabla se oslikavaju pomoću gumba *Pain Trees* . U početku, teren nema dostupne prototipove stabala, a kako bi se mogli koristiti potrebno ih je dodati. Klikom na gumb *Edit Trees > Add Tree*, otvara se prozor u kojem možemo odabrati dodatke i dodati stabla kao *Tree Prefab* za upotrebu s kistom. Nakon odabira željenih stabala, potrebno im je prilagoditi postavke i karakteristike. Da bi se uklonila stabla s terena, potrebno je držati tipku *Ctrl* dok traje oslikavanje.

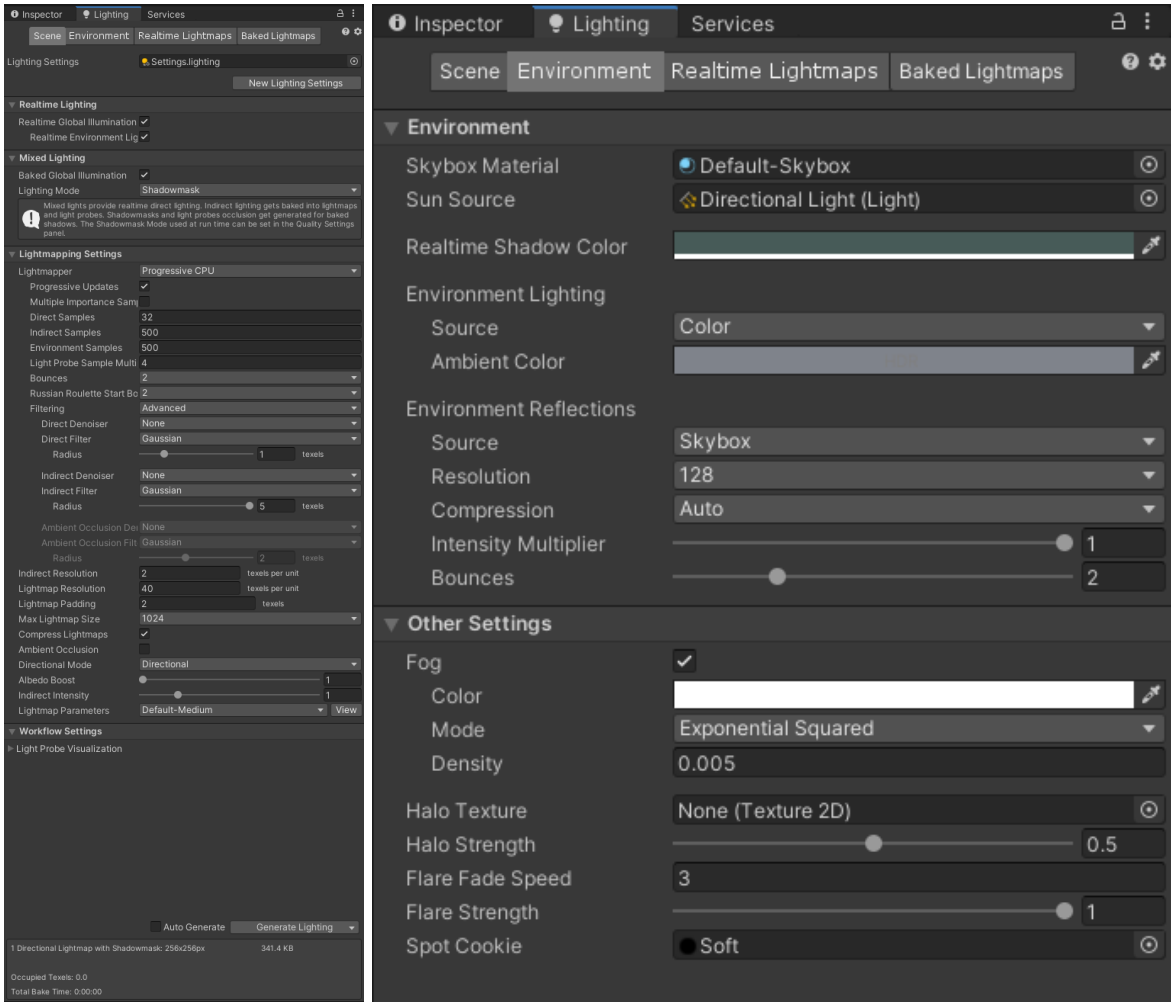
Slika 6.25 Teren

Slijedi postavljanje općih postavki terena. *Detail distance*, koji se odnosi na udaljenost do koje će se renderirati detalji te *detail density*, što se odnosi na broj detalja/travnatih predmeta u određenoj jedinici površine, postavljeni su na maksimum. Sljedeće je postavljanje stupnja do kojeg će vjetar savijati travnate predmete. Ukoliko bi se ostavile zadane vrijednosti, travnati predmeti bi se savijali na neugodan način. *Grass tint* je postavljen na neutralnu, sivu boju kako bi se izbjeglo izmjenjivanje boja travnatih površina prilikom puhanja vjetra.



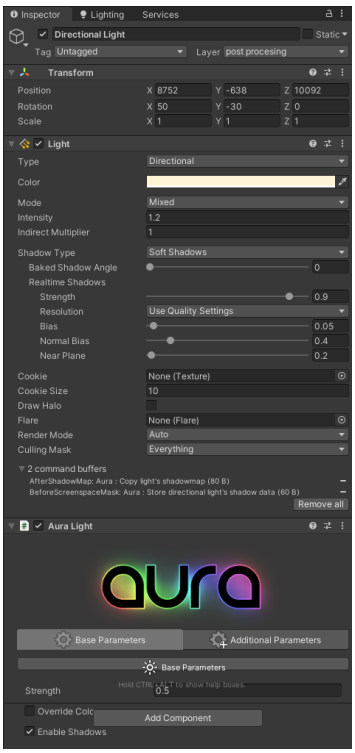
Slika 6.26 Opće postavke terena

Nakon postavljanja terena slijedi namještanje svojstava osvjetljenja. Izvor osvjetljenja je prema vlastitoj preferenciji umjesto na zadani *Skybox* postavljen na boju (eng. color), a ambijentalna boja je dodatno prilagođena. Dodana je i prilagođena opcija zamagljenja scene.



Slika 6.27 Namještanja svojstava osvjetljenja scene

Directional Light je prilagođen i dodana mu je komponenta za dodatni efekt osvjetljenja, *Aura-light*. Aura simulira rasipanje svjetlosti u mediju okoliša i osvjetljenje mikročestica koje su prisutne u ovom okruženju, ali nisu vidljive oku / kameri. Taj se efekt naziva i "volumetrijska magla" (eng. volumetric fog).



Slika 6.28 Prilagodba svjetla i dodavanje efekta za volumetrijsko osvjetljenje

Slika 6.29 Demonstracija postavljenih prilagodbi

Nakon postavljanja terena i šume, slijedi njegovo nadopunjavanje dodacima. Kao što je već spomenuto, ovdje se fokusira na slaganje glavne scene, te zatim ostatka terena. Kako bi se očuvala smislena i jasna hijerarhija objekata, pomoću naredbe desni klik > *EmptyObject* kreira se prazan *GameObject* koji se preimenuje u *Environment*. Unutar tog objekta, ubacuje se još objekata koji su podijeljeni na: *Rocks*, *Bushes*, *Shrooms*, *Campfire*, *MoreTrees*. Dodaci se jednostavno ubacuju u scenu povlačenjem iz *Asset* mape, te pomoću alata *move*, *scale* i *rotate* prilagođavaju i

postavljanju po sceni. Pomoći gumba *Paint Grass*, po terenu se dodatno oslikava pripremljena trava i cvijeće radi što boljeg efekta. Ovdje je ključno ne oslikavati veliku količinu trave po cijelom terenu jer šteti performansu te je bitno pripaziti da travke i cvijeće ne vire iz kamenja, što je učestala i jednostavna pogreška.

Korištene tehnike kojima je kreirana glavna scena proširene su na ostatak okruženja.

Slika 6.30 Prikaz ostatka terena

Dodatni efekti:

U projektu je aktivan efekt vjetra. Vjetar za travnate predmete je namješten u postavkama terena (eng. Terrain Settings), dok su stabala isto imala postavljeno iz *prefabs*. Takvi dodaci pružaju nužno kretanje okolini čineći ju tako življom.

Post Processing

Unity nudi brojne efekte naknadne obrade (eng. post-processing) te efekte na cijelom zaslonu, koju uz jednostavno postavljanje uvelike poboljšavaju izgled aplikacije.

Slika 6.31 Scena prije post processing efekata

Slika 6.32 Scena poslije post processing efekata

Budući da je puno efekta u paketu post-processing, u nastavku će se ukratko opisati efekti koji su se koristili za potrebe ovog projekta:

- Ambient Occlusion – zamračuje područja koja nisu izložena ambijentalnom osvjetljenju.
- Bloom – omogućuje sjaj svijetlim dijelovima scene.
- Vignette – zamračuje rubove slike.
- Color Grading – postupak izmjene ili korekcije boje i osvijetljenosti konačne slike.

Na kameru je postavljen *layer* za opciju *Post-Processing*. Ovu je knjižnicu potrebno instalirati pomoću *Windows* > *Package Manager*, koji otvara *Packages* prozor gdje je potrebno pronaći paket *Post Processing*. Zatim je kod hijerarhije objekata potrebno kreirati novi *GameObject* koji će se

nazivati *Volume*. Za post-processing potrebno je imati *Post-Process Volume* i *Post-Process Layer* (kamera), a oni moraju biti dodijeljeni novom, prilagođenom *layeru,* nazvanom post processing. Prilagođeni *layer* kreira se u zaglavlju Inspektor prozora, klikom na padajući izbornik *Layer* te odabirom naredbe *Add Layer*. Zatim je potrebno *Volume* i *Layer* dodijeliti prilagođenom *layeru post processing*.

Slijedi kreiranje *Post-processing profila,* a koji se kreira desnim klikom unutar *Asset* mape, zatim *Create > Post-processing Profile*. Isti se povlači u *Post-processing Volume* kako bi efekti mogli biti vidljivi. Za potrebe ovog projekta prvo je dodan i prilagođen efekt *Ambient Occlussion*. Slijedi *Bloom, Vignette* i *Color Grading*. Kod *Color Gradinga* je ponajviše rađeno na prilagodbi. Za veći efekt odabran je *ACES mode* te prilagođena nijansa (eng. tint), i povećan *Post-exposure (EV)* kako bi adekvatnije prilagodili taman efekt ACES-a. Prilagođen je i filter boja koji odlično nadopunjuje nijansu.

Postavljanje kontrolera

Da bi se postavio kontroler (eng. Controller) potrebno je kreirati novi prazan *GameObject* te u njega ubaciti kameru i na njega zakačiti skriptu za kretnje. Kamera na sebi ima skriptu za kretnje kamere. Skripta je minimalno modificirana uz pomoć primjera s interneta, kako bi se omogućila simulacija kretanja lika kao što je to u video igrama. Lik se po sceni može kretati pomoći tipaka W, A, S, D i micanjem miša, te može skakati tipkom *Space* i čučnuti s tipkom C. Iz aplikacije standardno se izlazi pritiskom tipke ESC.

Izvoz aplikacije

Nakon što je postavljena cjelokupna scena, pregledana i dodatno prilagođena, aplikacija je spremna za izvoz. Pomoću *File > Build Settings* otvara se prozor u kojem se dodaje trenutna scena projekta. Zatim se izvozi mapu koja sadrži sve potrebne *assete* i *.exe* datoteku. Sve što korisnik sada treba napraviti jest pokrenuti Camp.exe te početi istraživati nivo u kojem se našao.

Teksture i materijali u Unity-u

Tijekom procesa izrade ovog rada iz vlastite znatiželje odlučeno je isprobati teksturiranje u Blenderu i uspješan uvoz teksturiranih modela te teksturiranje i postavljanje materijala u Unity-u.

Za potrebe postavljanja tekstura i materijala unutar samog Unity-a odabran je izrađeni model mača, za kojeg su teksture i materijali preuzeti sa stranice Textures.com. Ovaj je proces vrlo jednostavan i brz. Nakon pronalaska odgovarajućih tekstura te preuzimanja određenih mapa tekstura, unutar *Asset* prozora i lokacije samog dodatka, desnim klikom kreiran je materijal. Njemu se zatim postavljaju preuzete teksture, jednostavnim povlačenjem iz *Asset* mape. Nakon

postavljanja svih tekstura i prilagođavanja svojstava materijala, pojedini se materijal povlači na određeni dio mača. Krajnji rezultat ovakvog načina je prikazan na slici ispod:

Audio

Na *playera* je zakačena komponenta *Audio Source*, i iz *Asseta* odabrani je zvuk povučen na AudioClip unutar *Audio Source* komponente.

7. Zaključak

Osim realizacije zadatka, cilj je ovog rada bio i upoznavanje i svladavanje korištenih alata za izradu zamišljenog nivoa video igre.

Današnji standard kvalitete video igara na izrazito je visokoj razini na kojoj se sama grafika igara sve više i brže približava realnom svijetu. Ipak, da bi se postigli takvi rezultati, za razvoj takve igre potreban je tim od više ljudi.

U radu su detaljno opisani alati korišteni pri izradi nivoa video igre srednjovjekovne tematike, s elementima inspiriranim iz poznate video igre Skyrim. Kroz teorijski dio pokrivene su osnovne informacije o pojedinom alatu, povijest, sučelje, alati te prednosti i mane, a u praktičnom dijelu rada prikazano je razrađivanje zadatka. Ovim se radom nastojalo obuhvatiti cjelokupan proces izrade nekog nivoa za video igru. Osnova svake video igre jesu 3D modeli, koji su u ovom slučaju izmodelirani u Blenderu, te korisničko sučelje putem kojeg korisnik komunicira s igrom izrađeno u Unity-u. Oba su alata pristupačna, zanimljiva i moćna, te zajedno daju odličan finalni produkt te dokazuju kako danas svatko, uz volju i trud, može kreirati vlastitu video igru. Ipak, kako bi igra izgledala dobro i bila kvalitetna, potrebno je puno vremena i više uključenih ljudi.

Iz jednostavnih točkica na blijedo obojenom zaslonu do piksela u boji i hiperrealističkih 3D pejzaža; zahvaljujući tehničkom razvoju priča o video igrama igrače vodi sve dublje u virtualni svijet. Kako će izgledati videoigre budućnosti? Jedno je sigurno; priča će se nastaviti u istom stilu. Već zahvaljujući virtualnoj stvarnosti, igrači gotovo mogu potpuno uroniti u igru. Ali ova je tehnologija još uvijek u povojima. Bolja grafička razlučivost, taktilni kontroleri i lakši uređaji definirat će budućnost industrije igara.

U Varaždinu _____ _____

Datum Potpis

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, MARTINA ŽGELA *(ime i prezime)* pod punom moralnom, materijalnom i kaznenom odgovornošću, izjavljujem da sam isključivi autor/ica završnog/diplomskog (*obrisati nepotrebno*) rada pod naslovom Izrada nivoa za video igru *(upisati naslov)* te da u navedenom radu nisu na nedozvoljeni način (bez pravilnog citiranja) korišteni dijelovi tuđih radova.

Student/ica:
(upisati ime i prezime)

Martina Žgela
(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, MARTINA ŽGELA *(ime i prezime)* neopozivo izjavljujem da sam suglasan/na s javnom objavom završnog/diplomskog (*obrisati nepotrebno*) rada pod naslovom Izrada nivoa za video igru *(upisati naslov)* čiji sam autor/ica.

Student/ica:
(upisati ime i prezime)

Martina Žgela
(vlastoručni potpis)

8. Literatura

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

[2] https://hr.wikipedia.org/wiki/3D_modeliranje, dostupno 01.07.2020.

[3] https://www.blender.org/foundation/history/ dostupno dostupno 05.07.2020.

[4] https://hr.wikipedia.org/wiki/Grafi%C4%8Dko_korisni%C4%8Dko_su%C4%8Delje dostupno 01.07.2020.

[5] https://artisticrender.com/more-than-30-blender-modeling-tools-explained/ dostupno 05.07.2020.

[6] https://en.wikipedia.org/wiki/Unity_Technologies dostupno 15.07.2020.

[7] https://docs.unity3d.com/Manual/UsingTheEditor.html dostupno 16.07.2020.

[8] https://www.pluralsight.com/blog dostupno 02.07.2020.

[9] https://docs.unity3d.com/Manual/index.html dostupno 01.07.2020

[10] https://medium.com/ dostupno 02.07.2020.

[11] https://learn.unity.com/ dostupno 05.08.2020.

[12] https://docs.blender.org/manual/en/latest/index.html dostupno 29.06.2020.

[13] https://www.blender.org/foundation/ dostupno 02.07.2020.

[14] https://www.investopedia.com/articles/investing/053115/how-video-game-industry-changing.asp dostupno 30.08.2020.

[15] https://blog.nationalmuseum.ch/en/2020/01/the-history-of-video-games/ dostupno 29.07.2020.

[16] https://gamedevacademy.org/ dostupno 25.08.2020.

[17] https://assetstore.unity.com/ dostupno 15.07.2020.

Popis slika

Slika 2.1 Vrijednost globalnog tržišta video igara 2012. do 2021., Izvor: Statista.com 3
Slika 4.1 Uvodni prozor Blendera 8
Slika 4.2 Početni zaslon Blendera 9
Slika 5.1 Korisničko sučelje alata Unity 13
Slika 6.1 Kocka – jedan od osnovnih geometrijskih oblika 16
Slika 6.2 Demonstracija alata Loop cut 16
Slika 6.3 Demonstracija manipuliranja točkama 17
Slika 6.4 Prikaz dobivene špice oštrice mača 17
Slika 6.5 Prilagodba ostatka oštrice mača 18
Slika 6.6 Opcije Bevel i Subdivision Surface 18
Slika 6.7 Opcija Shade Smooth 18
Slika 6.8 Finalan izgled izmodelirane oštrice mača 19
Slika 6.9 Rezultat Extrude Region alata 19
Slika 6.10 Dodavanje opcije preslikavanja 20
Slika 6.11 Nezadovoljavajući rezultati opcije Shade Smooth 20
Slika 6.12 Izglađivanje nepravilnosti u topologiji objekta 21
Slika 6.13 Finalan model branika mača 21
Slika 6.14 Finalizirana drška mača 22
Slika 6.15 Finalizirani vrh mača 22
Slika 6.16 UV mapiranje modela vaze 23
Slika 6.17 Prikaz čvorova (eng. node) u Shading Editoru 24
Slika 6.18 Model kojom nedostaju materijali i teksture 24
Slika 6.19 Texture Paint prozor 25
Slika 6.20 Teksturirani model vaze 25
Slika 6.21 Početni prikaz novokreiranog projekta u Unity-u 27
Slika 6.22 Struktura projekta 28
Slika 6.23 Teksturiranje terena 29
Slika 6.24 Opcija Paint Trees 29
Slika 6.25 Teren 30
Slika 6.26 Opće postavke terena 30
Slika 6.27 Namještanja svojstava osvjetljenja scene 31
Slika 6.28 Prilagodba svjetla i dodavanje efekta za volumetrijsko osvjetljenje 32
Slika 6.29 Demonstracija postavljenih prilagodbi 32

Slika 6.30 Prikaz ostatka terena 33
Slika 6.31 Scena prije post processing efekata 34
Slika 6.32 Scena poslije post processing efekata 34

Popis tablica

Tablica 4.1 Prednosti i nedostaci alata Blender 11
Tablica 5.1 Prednosti i nedostaci alata Unity 15