

TARTU ÜLIKOOL
Arvutiteaduse instituut
Informaatika õppekava

Max Frolov

**Eestikeelse õppematerjali loomine Unity mängumootorile
Bakalaureusetöö(9 EAP)**

Juhendaja:
Madis Vasser,Phd

Tartu 2021

Eestikeelse õppematerjali loomine Unity mängumootorile

Lühikokkuvõte: Selle bakalaureusetöö tulemusena valmis eestikeelne õppematerjal, mis käitleb erinevaid teemasid, mis on vajalikud mängu loomiseks mängumootoriga Unity. Õppematerjal annab läbijale esialgsed teadmised Unity kasutamiseks ja jaguneb teemadeks, mis käsitlevad mängu loomise elemente. Õppematerjali lõpptulemusena valmib 2D mäng, mida luues arendatakse arusaamist iga sammu taga olevate põhimõtete kohta.

Võtmesõnad: Õppematerjal, Unity, 2D mäng

CERCS: P175 Informaatika, süsteemiteooria; S281 Arvutiprogrammide kasutamise meetoodika ja pedagoogika

Creating educational material in Estonian for the Unity game engine

Abstract: As a result of this bachelor's thesis, educational material in Estonian was created, which deals with various topics that are necessary for creating a game with the Unity game engine. The training material provides the participant with initial knowledge of using Unity and is divided into topics that deal with the elements of game creation. The end result of the learning material is a 2D game, which is created to develop an understanding of the principles behind each step.

Keywords: Study material, Unity, 2D game

CERCS: P175 Informatics, system theory; S281 Computer-assisted education

Sisukord

Sissejuhatus	4
2. Mängumootorid	5
2.1 Unity tutvustus	5
2.2 Unity lühiajalugu	6
2.3 Alternatiivsed mängumootorid	8
2.3.1 Unreal Engine	8
2.3.2 Godot Engine	9
2.3.3 CryEngine	10
3. Õppematerjal	10
3.1 Õppematerjali vajadus	10
3.2 Olemasolevad õppematerjalid	11
3.3 Ingliskeelsete terminite tõlkimine	11
3.4 Õppematerjali ülesehitus	13
3.5 Eesmärk ja õpiväljundid	15
3.6 Õppematerjali testimine ja täiendamine	16
4. Kokkuvõte	17
Viidatud kirjandus	18
Lisad	21
Õppematerjal	21
Praktiline töö	21
Litsents	22

Sissejuhatus

Mänguarenduse valdkond kasvab ja areneb kiirelt, uusi tehnikaid ja tehnoloogiaid arendatakse pidevalt välja. Uue tehnoloogiaga kaasas käimiseks võib osutuda vajalikuks olemasoleva õppematerjali täiustamine või uue materjali loomine. Eestikeelset infot mänguarenduse kohta on vähe, kuid aasta-aastalt tunneb selle vastu huvi aina rohkem eesti keelt kõnelevaid inimesi. Alustamine võib aga olla keeruline ja esialgu harjumatu, mistõttu võib esmalt vaja minna abi. See rõhutab eestikeelse juhendmaterjali olulisust, mis kirjeldaks kiiresti põhitõdesid ja näitaks, kui lihtne on luua oma esimest mängu.

Käesoleva bakalaureusetöö eesmärgiks on anda eestikeelne ülevaade mängumootorist Unity, luua kasutajasõbralik juhend ja praktiline rakendus, järgides juhendi spetsifikatsioone. Õppematerjali järgides kujuneb arusaam iga mänguarenduse sammu taga peituvatest põhimõtetest. Õpitakse kasutama Unity tarkvara ja mõistma samal ajal mängude loomise olulisimaid aspekte.

Õppematerjal loodi mõeldes algajatele, kellel on elementaarsed programmeerimisoskused ja põhiteadmised programmeerimisest. Bakalaureusetöös käsitletakse Unity mängumootori ajalugu ja konkureerivaid mängumootreid. Töö koosneb lisaks materjali õpiväljundite kirjeldusest ning alternatiivsete õppematerjalide ülevaatest. Lõputöö teema valiti seetõttu, et eestikeelne õppematerjal Unity kohta on puudulik.

2. Mängumootorid

Selle peatüki eesmärk on tutvustada erinevaid mängumootoreid. Samal ajal aitab see lugejal mõista, miks Unityt kasutatakse ja kuidas see ühtlustub teiste mängumootoritega. Lisaks antakse lühike ülevaade Unity mängumootori ajaloost.

Tüüpiline mängumootor sisaldab palju erinevaid komponente, millest igaüks vastutab mängu teatud aspektide eest. Levinud komponentide hulka kuuluvad füüsikamootor (ingl k **physics engine**), ressurshaldur (**resource manager**), renderdushaldur (**render manager**), sisendihaldur (**input manager**), helihaldur (**audio manager**) ja stseeni haldur (**scene manager**). Ressurshaldur on kettalt ressursside toomiseks ja mällu laadimiseks, renderdushaldur graafiliste ressursside kuvamiseks graafika riistvara abil, helihaldur heli taasesitamiseks, sisendihaldur, mis loeb kasutaja sisendit sisendseadmetelt, ja füüsikamootor vastutab selle eest, et mängumaailma objekte mõjutaksid määratud jõud või füüsikaseadused. Need olulised komponendid on enamiku mängude toimimiseks vajalikud. Näiteks ei piisa lihtsalt graafikaressursi laadimisest failist ilma eesmärgita või heliressursi esitamisest ilma põhjusega. Failidest ressursside laadimise ja nende esitamise vahel eksisteerib mängus vahepealne haldus- ja kompositsioonikiht ehk stseeni haldur. See kiht otsustab, kuidas ressursse omavahel seostatakse ja kasutatakse, et anda neile mängija jaoks tähendus seoses mängu tegevuste ja sündmustega aja jooksul.[7]

2.1 Unity tutvustus

Unity on 2D/3D mängumootor ja võimas platvormideülene integreeritud arenduskeskkond, mis on mõeldud peamiselt arvutimängude loomiseks ja arendamiseks. Unity on võimas tööriist, mida paljud mänguarendajad kasutavad oma loomingu loomiseks ja jõustamiseks [1]. Kuigi mängu on võimalik luua ka ilma mängumootorite abita, on suur osa mängustuudiodid ja arendajaid otsustanud seda kasutada, kuna see lihtsustab oluliselt rakenduste arendusprotsessi. Mõned mängumootorite peamised eelised on arenduskiirus, võimalus näha muutusi reaajas ja mängu erinevatele platvormidele teisaldamise lihtsus. Mängumootor muudab graafiliste esitluste loomise palju lihtsamaks, kuna osa koodist on juba olemas. Arendajad saavad vajalikke muudatusi või täiendusi tehes olemasolevat koodi uuesti kasutada, mis säästab aega, vähendab kulusid ja võimaldab tootel kiiremini turule jõuda. Mängumootorite abil saab luua arvutimänge, aga ka erinevaid simulatsioone, õpikeskkondi ja visualiseerimiskendusi [5,6].

Mängumootorina on Unity turul kõige populaarsem, kuna see töötab mitmel platvormil ja on väga kasutajasõbralik nii algajatele kui ka asjatundjatele. Tarkvara pakub mängijatele 2D- ja 3D-mängude loomisel mitmeid eeliseid. Unity teeb veelgi mugavamaks aga see, et see sisaldab õitsvat varapoodi, see on sisuliselt koht, kuhu arendajad saavad oma loomingut üles laadida ja kasutajatele kättesaadavaks teha. See õpetus kasutab teiste arendajate loodud

varasid, mis hõlmavad joonistusi, taustapilte, karaktereid, häält jms, mis lihtsustab mängu kontseptsioonide õpetamist [2]. Unity eeliseks on ka see, et see ei nõua palju programmeerimiskogemust, kasutades lihtsaks peetavat C# programmeerimiskeelt, liides on väga intuitiivne ja arhitektuur on lihtne, nimelt saab Unity luua ainult objekte ja komponente [3].

Unity on tuntud indie- ja mobiilmängude arendamise poolest. 2019. aasta septembri seisuga kasutas Unity mootorit 52 protsenti tuhandest populaarseimast mobiilmängust ja samas ettevõtte andmetel 60% kogu AR/VR sisust [8]. Unity on populaarne väiksemate projektide jaoks, kuna see on tasuta. Unity mängumootori isiklik litsents võimaldab üksikisikutel luua kommerts mängu tasuta, tingimusel et nad ei ole viimase 12 kuu jooksul teeninud rohkem kui 100 000 dollarit tulu Unity kasutamisest [9]

2.2 Unity lühiajalugu

Unity asutasid 2005. aastal kolm kolleegi David Helgason, Joachim Ante ja Nicholas Francis eesmärgiga muuta mänguarendus suuremale hulgale arendajatele kättesaadavamaks. Unity ilmus algselt Mac OS X operatsioonisüsteemile ning hiljem lisati tugi Microsoft Windowsile ja veebibrauseritele. Esialgne toode avaldati 6. juunil 2005. aastal [10].

Unity 2.0 ilmus 2007. aastal koos ligikaudu 50 uue funktsiooniga, sealhulgas optimeeritud maastikumootoriga üksikasjalike 3D-keskkondade jaoks ja reaajas dünaamiliste varjude lisamisega, lisades samal ajal funktsioone, mis võimaldasid arendajatel paremini koostööd teha. Selle tegi võimalikuks Unity Asset Server – versioonihaldussüsteem, mis jälgib mänguvarasid ja skripte ning võimaldab importida sätteid ja muid metaandmeid. Samuti tehti värskendusi võrgukihti, mis võimaldas arendajatel luua mitme mängijaga mängu, kasutades kasutajadatagrammi protokoll (ingl k **User Datagram Protocol**, UDP), mis pakub võrguadressi tõlkimist (ingl k **Network Address Translation**, NAT) ja kahte võrgusuhtlusmeetodit [11]. Kui Apple 2008. aastal oma App Store'i käivitas, lisas Unity iPhone'i toe. Mitu aastat ei olnud mängumootorile iPhone'is konkurentsi ja see aitas iOS-is saada mängude arendamise domineerivaks platvormiks [12].

Unity 3.0 ilmus 2010. aasta septembris koos uute funktsioonidega, mis parandasid lauaarvutite ja videomängukonsoolide mootori graafikavõimet. Need funktsioonid olid tol ajal kallid, nii et Unity 3.0 võeti hästi vastu. Lisaks Androidi toele sisaldas Unity 3 ka uusi tööriistu, millest silmapaistvaim oli **Illuminate Labs' Beast Lightmap tool**, mida kasutati sellistes mängudes nagu Mirror's Edge ja Killzone 2 tõhusate ja kvaliteetsete 32-bitiste varjukaartide loomiseks [13]. Üle 1,3 miljoni arendaja on Unity tööriistu kasutanud oma iOS-i, Androidi, konsooli, arvuti ja veebimängude jaoks vapustava graafika loomiseks [14]. Ajakirja Game Developer 2012. aasta maikuu küsitlus leidis, et Unity on parim mobiiliplatvormide mängumootor [15].

Unity Technologies andis Unity 4.0 välja novembris 2012. See versioon lisas DirectX 11 ja Adobe Flashi toe, uue animatsioonitööriista nimega Mecanim ja juurdepääsu Linuxi eelvaatele. Facebook integreeris Unity mängumootorit kasutavate mängude tarkvaraarenduskomplekti 2013. aastal [16]. 2016. aastal töötas Facebook koos Unityga välja uue arvutimängude platvormi. Unity pakkus tuge Facebooki mänguplatvormidele, nii et arendajad said mängu Facebooki kiiremini eksportida ja avaldada [17].

Unity 5 kuulutati välja 2014. aastal, kuid ilmus sama aasta oktoobris. Unity 5 parandab mängumootori valgustust ja heli, võimaldades arendajatel lisada oma mängu mis tahes ühilduvasse veebibrauserisse, ilma et oleks vaja mängijate jaoks pistikprogramme [18]. Mängumootori versioon 6 lisas uusi valgustus- ja osakesteefekte, uuendas mootori üldist jõudlust ning lisas tuge Nintendo Switchile, Facebook Gameroomile, Google Daydreamile ja Vulkaniga graafika rakenduse programmeerimisliidesele (ingl k Application Programming Interface, API) loomuliku toe.

2016. aasta detsembris teatas Unity Technologies, et muudab Unity versioonide nummerdamissüsteemi järjestuspõhistelt identifikaatoritelt väljalaskeaastale, et viia oma versioonid vastavusse nende sagedasema väljalaske kadentsiga [19]. Unity 2017.2 on tõstnud esile plaanid muuta Unity mitteinseneridele kättesaadavamaks. See hõlmas selliseid tööriistu nagu Timeline, mis võimaldas arendajatel animatsioone mängudesse tõmmata, ja Cinemachine, mängude nutikas kaamerasüsteem. Need kaks tehnoloogiat võimaldasid automatiseerida paljusid mängude kujundamise protsesse, vähendades oluliselt vajalikku koodi, nii et peaaegu igaüks saaks neid kasutama hakata [20].

Unity 2018 sisaldab võimsat Scriptable Render Pipeline'i, mida arendajad saavad kasutada kvaliteetse graafika loomiseks. See hõlmab kõrglahutusega renderduskonveieri (ingl k Rendering Pipeline) konsooli- ja arvutikogemuste jaoks, aga ka kergeid renderduskonveiereid mobiilside, virtuaalreaalsuse, liitreaalsuse ja segareaalsuse jaoks. Unity 2018 sisaldab ka masinõppevahendeid, nagu imitatsioonõpe, kus mängud õpivad päris mängijate harjumustest [21].

2020. aasta seisuga töötas Unity mängumootoriga loodud tarkvara enam kui 1,5 miljardil seadmel. Unity ütleb, et selle mängumootoriga tehtud rakendused moodustavad 50% kõigist mobiilmängudest ja neid laaditakse alla rohkem kui 3 miljardit korda kuus [22]. Suuremateks mängudeks, mis tehti mängumootoriga Unity on näiteks Pokemon Go ja Activisioni Call of Duty Mobile. 2020. aasta juunis tutvustas Unity sega- ja liitreaalsuse nimega Mixed and Augmented Reality Studio (MARS). See tööriist pakub arendajatele rohkem funktsioone liitreaalsuse (ingl k augmented reality,AR) rakenduste loomiseks [23]. 2020. aastal ostis Unity Finger Food Advanced Technology Groupi, mis nägi selles vahendit, et tugevdada oma mitte-videomängude kasutamist ja pakkuda klientidele täiendavat disainiabi. Ettevõtte läks börsile 2020. aasta septembris, et kasutada oma mängumootorit teistes tööstusharudes.[24]

2.3 Alternatiivsed mängumootorid

Saadaval on palju erinevaid mängumootoreid ja arendustööriistu, millest igaühel on erinev hinnapoliitika ja võimalused. Saadaval on palju populaarseid mängumootoreid, sealhulgas CryEngine ja Godot. Suurim konkurent on aga Unreal Engine. See peatükk annab ülevaate loetletud mängumootoritest, nende funktsioonidest ja erinevustest võrreldes Unity mootoriga.

2.3.1 Unreal Engine

Unreal Engine on mängumootor, mille on välja töötanud Epic Games. See ilmus esmakordselt esmaisikulise tulistamismänguga Unreal aastal 1998. Vaatamata asjaolule, et see töötati välja esimese isiku tulistamismängude jaoks, pääses mängumootor kiiresti žanripiiridest ja sai laialdaselt kasutust kõigis erinevat tüüpi mängudes. Mõned kaasaegsed näited edukatest Unreal Engine mängudest on Final Fantasy VII Remake, PUBG: BATTLEGROUNDS, Borderlandsi seeria, Tekken 7 ja Epic Games'i enda Fortnite. Mootoril on eelvalmistatud mängumallid, mis sobivad suurepäraselt selliste žanrite jaoks nagu esimese isiku laskurid ja muud interaktiivsed projektid, nagu tööstuslik visualiseerimine.

Unreal Engine, mis on kirjutatud C++ keeles, pakub suurt teisaldatavust ja toetab mitmesuguseid erinevaid platvorme, sealhulgas Microsoft Windows, macOS, Linux, Android, iOS, praeguse ja eelmise põlvkonna konsoolid ning mitmed virtuaal reaalsuse peakomplekte [25]. Kuni Unreal Engine'i neljanda versioonini oli peamine skriptikeel UnrealScript. Lõppkokkuvõttes peeti C++ ja Blueprints'i visuaalset skriptisüsteemi kasulikumaks kui alternatiivi. Unreal Engine pakub ka mitmesuguseid sisseehitatud komponente, sarnaselt Unity omadele.

Mängumootor on hästi dokumenteeritud ja sellel on aktiivne foorumikogukond ning abistavad ingliskeelsed juhendid Epic Gamesilt ja mootori entusiastidelt. Unreal Engine'i varade pood on nagu Unity varade pood, kust saab osta ja alla laadida varasid, malle ja pistikprogramme, mida oma projektides kasutada. Mängumootoril pole nii suurt sõltumatut arendajate kogukonda kui Unityl, sest Unity on populaarsem kui Unreal Engine. Mängumootori graafilised omadused muudavad selle suurepäraseks valikuks kvaliteetsete mängude loomiseks, mistõttu otsustavad paljud stuudiod seda kasutada fantastiliste projektide (nt filmide või mängude) arendamiseks. Viimase kahe aastakümne jooksul on Unreal Engine'ist saanud üks populaarsemaid ja töökindlaimaid mängumootoreid maailmas.

Seda on kasutatud mitmesuguste mängude loomiseks, sealhulgas mõned kõige populaarsemad mängud turul.

2.3.2 Godot Engine

Godot Engine on tasuta ja avatud lähtekoodiga mängumootor, mida arendab ja hooldab sõltumatute panustajate kogukond. Godot on olnud alates 2014. aastast, kuid mootor on tõeliselt populaarsust kogunud alles viimastel aastatel. Godoti mängumootor on suurepärase, kui otsida midagi tasuta ja avatud lähtekoodiga, mis tähendab, et saab mootorit muuta ja luua mängu nii, nagu soov on. Mootor on piisavalt mitmekülgne, et toetada nii 2D- kui ka 3D-mängu, seega sobib see suurepäraselt igat tüüpi mängude jaoks. Godot'ga on tehtud mõned edukad mängud, nende hulgas on Cruelty Squad, Beat Invaders, Wrought Flesh ja Ex-Zodiac.

Godot Engine'i redaktoril on tööriistad sisu loomiseks ja haldamiseks ning sellel on ka sisseehitatud skriptiredaktor. Erinevalt Unity'st ja Unrealist ei kasuta Godot komponendipõhist lähenemist, selle asemel on erinevad objektide klassid, mida nimetatakse funktsioonikastideks (ingl k node). Godot's saab iga objekt või funktsioonikast olla ainult kindlat tüüpi ja sellele võib olla lisatud konkreetne skript. Saab luua korduvkasutatavaid stseene, ühendades funktsioonikastid kokku. Need stseenid käituvad sarnaselt Unity monteeritava valmisobjektiga [26]. Skripti saab lisada mis tahes funktsioonikastile tema käitumise laiendamiseks. Godotil on spetsiaalselt loodud skriptikeel nimega GDScript, mis on oma süntaksi poolest sarnane Pythoni omaga [27]. Godot'l on eraldi 2D- ja 3D-renderduskonveier. 3D-renderdamine järgib füüsikapõhist lähenemist varjutamisele ja valgustusele, mis hõlmab enamiku Unity ja Unreal Engine sagedamini kasutatavatest funktsioonidest. 2D-renderdamisel on ka mõned spetsiifilised funktsioonid, nagu 2D-valgustus, maskeerimine ja varjud. Projekti ekspordi tugi Godot Engine'is on saadaval töölauplatvormidele, mobiilile ja veebile.[28]

Godot toetub entusiastide kogukonnale, kes toetab selle arendamist ja arendajaid, kes seda kasutavad. Need inimesed panustavad oma aega ja vaeva, et muuta Godot parimaks võimalikuks tööriistaks mängu arendamiseks. Godot'l on saadaval üksikasjalik ametlik dokumentatsioon ja avalik GitHubi leht, nii et igaüks võib seda näha.

2.3.3 CryEngine

CryEngine'i kujundas Saksa arendaja Crytek. CryEngine'i on varem kasutatud paljude kvaliteetsete mängude loomiseks ja seda täiustatakse pidevalt. CryEngine on tuntud oma esteetiliselt nauditavate ja kvaliteetsete kõrge eraldusvõimega mängude poolest. Aja jooksul värskendavad nad oma süsteemi nii, et uued konsoolid saaksid toetatud. Mõned CryEngine'iga tehtud mängud on järgmised: Sniper – Ghost Warrior 2 ja Snow Warhorse. Seda mängumootorit kasutavad ka suured mängufirmad nagu Ubisoft. Ubisoft säilitab oma CryEngine'i modifitseeritud versiooni, tuntud kui Dunia Engine, mida kasutatakse nende hilisemates Far Cry seeria iteratsioonides [29].

Skriptimist CryEngine'is saab teha C++, C # ja lua abil. Seetõttu saavad mänguarendajad kasutada keelt, mida nad eelistavad. Aastal 2016 avalikustas Crytek uusima versiooni CryEngine 5 ja teatas samas uuest litsentsimismudelitest, kus inimesed saavad mängumootori kasutamise eest maksta mida iganes nad tahavad. Seda tehes oli võimalik kindlaks teha, kui suur osa makstud rahast läks mängumootori loojale ja kui suur osa Crytekile. 20. märtsil 2018 läks Crytek üle tulude jagamise mudelile, kus kui mäng teenis rohkem kui 5000 dollarit, pidi maksta 5% kogutulust [30]. Mängumootoreid Unreal Engine ja Unity on üsna lihtne õppida, kuid CryEngine on vähem kasutajasõbralik, kuna saadaval on vähem ressursse ja kogukondi, isegi kui kvaliteet on hea [31].

3. Õppematerjal

Järgnev peatükk annab ülevaate sellest, miks on käesoleva bakalaureusetöö käigus loodud juhend vajalik, millistele põhimõtetele õppematerjal on üles ehitatud ning alternatiivsetest õppematerjalidest. Lisaks on antud juhendis kasutatud ingliskeelsete terminite selgitused ning loetletakse õppematerjali täitmisega saavutatud õpitulemused.

3.1 Õppematerjali vajadus

Vajadus õppematerjali järele tekkis seetõttu, et Unity mängude arendamiseks ei olnud piisavalt õppematerjale saadaval eesti keeles. Tartu Ülikool pakub arvutimängude loomise ja disaini kursust (ainekoodiga MTAT.03.263), kus õpetatakse Unityt kasutama, kuid kõik juhendid on inglise keeles.

Õppematerjali saab kasutada ka eraldi mängumootori Unity õppeaine osana Tartu Ülikoolis või mõnes muus õppeasutuses. Loodud õppematerjal on kasulik ka neile huvilistele, kes koolis ei õpi, kuid soovivad selle mängumootoriga tutvuda.

3.2 Olemasolevad õppematerjalid

Autorile pole teada eestikeelseid õppematerjale, mis kasutaksid mängumootorit Unity. Selles peatükis on lühidalt ülevaade mõnest ingliskeelsest materjalist.

Tartu Ülikool pakub kahte Unity programmeerimise kursust, mis mõlemad on inglise keeles. Üks neist on Tartu Ülikooli arvutimängude loomine ja kujundamine (ainekood MTAT.03.263), mis on kättesaadavad Tartu Ülikooli kursuste keskkonnast courses ning millele pääseb ligi Tartu Ülikooli kasutajatunnuse ja parooliga. Antud õppematerjal tutvustab Unityt ja koosneb ülesannetest, mida saab täita erinevat tüüpi mängude loomiseks [32]. Lisaks on ka õppematerjal Unity kohta aines mängumootorid (ainekoodiga LTAT.02.218), mis on ka kättesaadav Tartu Ülikooli keskkonnast courses. Mõlemad ained õpetavad 2D mängu loomise konsepte ning neid läbides tuleb luua oma mäng projektina [33].

Unity ametlikul kodulehel on erinevad ingliskeelsed juhendid, mille on välja töötanud Unity töötajad ise [34]. Saadaval on üle 750 tunni tasuta õppesisu, mis aitab oma eesmärged saavutada. Saadaval on palju erinevaid juhendeid, mis aitavad lahendada konkreetseid probleeme või mängu nullist üles ehitada.

Loodud õppematerjalide kõrval leidub ka palju kasutajaskonna poolt loodud materjale, millest kõige enam laetakse üles videomaterjale videojagamiskeskonnas YouTube [35].

3.3 Inglisekeelsete terminite tõlkimine

Kuna eestikeelseid õppematerjale Unity mängumootori jaoks on vähe, puudub selle jaoks vastav eestikeelne terminoloogia. Selles õpetuses käsitletakse ainult Unityga mängude loomise konkreetseid termineid (Tabel 1). Mõningaid konkreetseid mängude nimesid ei tõlgita, seega otsige need konkreetset sõnad kindlasti Unity dokumentatsioonist.

Tabel 1. Terminoloogia tõlked ja seletused.

Tõlge	Inglisekeelne mõiste	Seletus
Vara	Asset	Unity vara on üksus, mida saate oma mängus või projektis kasutada. Vara võib pärineda mis tahes failist, mida Unity toetab (nt 3D-mudel, helifail, pilt või midagi muud).
Plaatide kaart	Tilemap	Maatrikskujuline blokkidest koosnev kaart. Tilemapi komponent on süsteem, mis salvestab ja haldab plaatide varasid 2D tasemete loomiseks. Plaadid on varad, mis on paigutatud

		plaatide kaartidele 2D-keskkonna loomiseks. Iga plaat viitab valitud spraidile, mis seejärel renderdatakse plaadi asukohas.
stseen	Scene	Stseenid sisaldavad teie mängu objekte. Neid saab kasutada menüüde, individuaalsete tasemete ja muu loomiseks. Mõelge igat stseeni kui ainulaadset mängutasemet. Igas stseenis loote oma keskkonna, takistused ja kaunistused, seejärel ühendate need kõik kokku, et luua oma mänguosa.
sprait	Sprite	Spraidid on pildid, mis kujutavad mängu varasid, nagu tegelasi, vaenlasi, mürske ja muid objekte.
Skript	Script	Skriptimine ütleb mänguobjektidele, kuidas käituda. Mänguobjektidele lisatud skriptid ja komponendid ning see, kuidas need omavahel suhtlevad, loovad teie mängu.
Mänguobjekt	GameObject	Mänguobjektid on Unity põhiobjektid, mis esindavad tegelasi, rekvisiite ja maastikke. Nad ei saavuta iseenesest palju, kuid toimivad komponentide konteineritena, mis rakendavad tegelikku funktsionaalsust.
Põrkur	Collider	Põrkurid on Unity'i komponent, mis võimaldab füüsikamootoril kokkupõrgetega hakkama saada. Füüsikaline mootor simuleerib kokkupõrkeid objektide abil. Saame kasutada põrkeid, et teha kindlaks, millised objektid üksteisega põrkuvad või kuidas objektid kokkupõrkel käituvad.
Komponent	Component	Unity'is on komponendid tükid, mille kinnitate oma mänguobjekti külge. Mänguobjektidel on osi või komponente, mille kinnitate või lisate neile, et panna need tegema asju, mida soovite.
Monteeritav valmisobjekt	Prefab	Valmisobjektid on spetsiaalset tüüpi komponendid, mis võimaldavad täielikult konfigureeritud mänguobjekte salvestada projekti taaskasutamiseks. Neid varasid saab stseenide või muude projektide vahel uuesti kasutada, ilma et peaksite neid uuesti konfigureerima.
Jäik keha	Rigidbody	Jäigad kehad on komponendid, mis võimaldavad mänguobjektidel reageerida reaajas füüsikale. Sellesse kategooriasse kuuluvad kõik füüsikalised nähtused, nagu jõud ja gravitatsioon, mass, takistus ja impulss.
Funktsioon	Function	Funktsioon on väidete kogum, mis täidab teatud ülesannet. Funktsioon võib võtta argumentide komplekti, mille alusel tegutseda, ja tagastada väärtuse, mis esindab selle sooritatud ülesande tulemust.
Renderdaja	Renderer	Renderdaja on see, mis paneb objektid ekraanile ilmuma. Renderdusmootoreid saab keelata, et muuta objektid nähtamatuks, ning nende kaudu saab materjalidele juurde pääseda ja neid muuta.

Inspektor	Inspector	Inspektor kuvab üksikasjalikku teavet valitud mänguobjekti kohta, sealhulgas kõiki lisatud komponente ja nende omadusi.
Kiht	Layer	Unity kihid aitavad näidata mänguobjekti omadusi, näiteks milliseid mänguobjekte peaks ignoreerima või mis on kaamerale nähtamatud. See süsteem on kasulik keerukate süsteemide loomiseks või mängufunktsioonide, näiteks kokkupõrketuvastuse, toetamiseks.
Laskekeha	Projectile	Mänguarenduse kontekstis on see tavaliselt väike mänguobjekt, mille on loonud mõni teine objekt, mis liigub mööda etteaimatavat rada sageli, kuid mitte alati otse. Objekt hävib, kui see põrkub kokku tahke esemega ja põhjustab kahju, kui see teine objekt on mürsu põhjustanud objekti vastane.
Sündmuste süsteem	Event System	Sündmuste süsteem pakub võimalust saata sündmusi rakenduse objektidele sisendi põhjal. Sündmuste süsteem sisaldab mõnda komponenti, mis töötavad koos sündmuste saatmiseks.
Kasutajaliides	User interface	Unity kasutajaliides on tööriistakomplekt mängude ja rakenduste arendamiseks. See on kasutajaliidese süsteem, mis kasutab mänguobjekte ja mänguvaadet kasutajaliideste korraldamiseks, paigutuseks ja stiiliks.
Redaktor	Editor	Unity redaktor võimaldab lisada kohandatud menüüsid, mis näevad välja ja käituvad nagu sisseehitatud menüüd.

3.4 Õppematerjali ülesehitus

Õppematerjal on jagatud osadeks, kus iga osa käsitleb erinevat teemat. Kõik peatükid kokku moodustavad ühtse terviku. Õppematerjali ülesehitus sai valitud Unity ametliku õpetuse järgi, mis oli loodud algajatele. [36]

Läbides materjali algusest lõpuni, saab lihtsa pealtvaatega arvutimängu (vt Joonis 1), mille maailmakujundus on enda loodud, lisades ohtlikud vastased, mängija poolt juhitava animeeritud tegelase, lisades relva. Samal ajal luuakse lihtne kasutajaliides ja dialoog teise mängutegelasega.



Joonis 1. Ekraanitõmmis valminud mängust

Soovitud tulemuse saavutamiseks on kasutajal soovitatav peatükid järjest läbi töötada, kasutades igas peatükis erinevaid tööriistu ja tehnikaid. Lõppkasutajale mõeldud õppematerjal on leitav bakalaureusetöö lisa 1 juures.

Peatükkide teemad on lühidalt järgmised:

- 1. peatükk (Projekti ülesseadistamine) on sissejuhatav peatükk, kus tutvustatakse kuidas projekti üles sättida ja milliseid Unity varasid projektis kasutatakse.
- 2. peatükk (Mänguobjekti liikuma panemine) käsitleb tegelase spraidi lisamist mängu, skripti loomist ja antakse ülevaade skriptimiskeelest C#. Lisaks õpetatakse kuidas tegelast liikuma panna skripti abil.
- 3. peatükk (Maailma disain) käsitleb mängumaailma loomist kasutades Unity plaatide kaarti, lisaks õpetatakse kuidas kihtide järjekorrad töötavad.
- 4. peatükis (Tegelase ja mänguobjektide interaktsioon mängu maailmaga) tutvutakse lähemalt Unity füüsika süsteemiga, st kuidas lisada kokkupõrkeid ja jäikaid kehakomponente nii, et füüsikaseadused rakenduksid.
- 5. peatükis (Tegelasele elude lisamine ja vastased mängumaailmas) lisatakse tegelasele elud ja mängijategelasele lisandub mängumaailmas uus vastane, kes liigub iseseisvalt ja kahjustab mängijat temaga kokku puutudes.
- 6. peatükk (Animatsioonid) tutvustab lugejale animatsioone ja nende loomist animaatori abil.
- 7. peatükis (Kaamera muutmine filmikaameraks) vaadatakse lähemalt Unity Cinemachine'i paketti, mida saab kasutada kaamera juhtimise automatiseerimiseks.
- 8. peatükk (Tegelase viskerelva loomine) õpetab kuidas viskekeha lisada tegelasele ja hävitada, kui see vastase või mõne mänguobjektiga kokku puutub.
- 9. peatükis (Visuaalne kasutajaliides) lisatakse lihtne kasutajaliides mängutegelase elude kuvamiseks.

- 10. peatükis (Dialogi loomine teise tegelasega) lisatakse uus tegelane mängumaailma, kellega mängija saab interakteeruda.
- 11. peatükis (Mängu ehitamine ja käivitamine) käsitletakse mängu eksportimist mängumootorist eraldiseisvaks rakenduseks.

3.5 Eesmärk ja õpiväljundid

Selle õppematerjali eesmärk on õpetada Unity mängumootori põhitõdesid, aidates samal ajal luua lihtsat pealtvaates mängu. Õppematerjal on suunatud algajatele mänguloojatele. Õppijal on soovitatav eelnev programmeerimise kogemus, et materjal oleks paremini mõistetav.

Õpiväljundid on kirja pandud laias laastus kogu õppematerjali mõistes ning need on ka peatükkide kaupa liigendatud. Üldised õpiväljundid kirjeldavad seda, mida õpilane õpib pärast õppematerjali läbimist. Õpitulemused peatükkide kaupa näitavad, mida õpilased pärast iga peatüki läbimist õpivad.

Õppematerjali õpiväljundid on peatükkide kaupa järgmised (tabel 2):

Tabel 2. Õppematerjali õpiväljundid.

Peatükk	Õpiväljundid
Projekti ülesseadistamine	<ul style="list-style-type: none"> • Õpilane oskab üles seadistada projekti kasutades Unity hubi. • Õpilane oskab lisada Unity vara poest pakette oma projekti ja luua uut stseeni.
Mänguobjekti liikuma panemine	<ul style="list-style-type: none"> • Õpilane oskab mängumaailma lisada objekte. • Õpilane oskab luua skripti ja teab tema ülessehitust. • Õpilane teab skripti põhi funktsioonide tähendust. • Õpilane õpib kasutama sisendsüsteemi, et skripti abil tegelane liikuma panna.
Maailma disain	<ul style="list-style-type: none"> • Õpilane õpib kasutama plaatide kaarti, et kujunda oma mängu maailma • Õpilane õpib lisama spraiite plaatide paletti, millega kujundatakse mängu disain. • Õpilane õpib kasutama Unitys kihte ja mille jaoks need vajalikd on. • Õpilane õpib maailma kujunduse muutmist realistlikumaks. • Õpilane õpib kasutama monteeritavat valmisobjekti.

Tegelase ja mänguobjektide interaktsioon mängu maailmaga	<ul style="list-style-type: none"> • Õpilane tutvub lähemalt Unity füüsika süsteemiga ja oskab seda rakendada. • Õpilane oskab lisada pörkjaid objektidele ja muuta neid vajadusel jäikadeks kehadeks. • Õpilane õpib optimiseerima tegelase ja mängumaailma vahelist interaktsiooni.
Tegelasele elude lisamine ja vastased mängumaailmas	<ul style="list-style-type: none"> • Õpilane oskab luua skriptis funktsioone • Õpilane oskab luua vastaseid ja neid iseseisvalt liikuma panna. • Õpilane oskab tegelasele elusid lisada või eemaldada, kontakteerudes vastasega.
Animatsioonid	<ul style="list-style-type: none"> • Õpilane õpib kasutama animaatorit animatsioonide lisamiseks objektidele. • Õpilane oskab muuta tegelase animatsiooni vastavalt mängija sisendile.
Kaamera muutmine filmikaameraks	<ul style="list-style-type: none"> • Õpilane tutvub lähemalt Unity pakettiga Cinemachine ja õpib automatiseerima kaamera liikumist. • Õpilane õpib kuidas muuta kaamera seadeid nii, et näha oleks ainult mängumaailma sees olev ala.
Tegelase viskerelva loomine	<ul style="list-style-type: none"> • Õpilane õpib looma laskekeha • Õpilane oskab mõnest mänguobjektist instantseerida laskekehasid ja panna need liikuma. • Õpilane oskab panna laskekehad kontakti ainult teatud mänguobjektidega.
Visuaalne kasutajaliides	<ul style="list-style-type: none"> • Õpilane tutvub lähemalt komponendiga canvas ja oskab selle abil luua kasutajaliidest • Õpilane oskab muuta kasutajaliidest dünaamiliseks.
Dialogi loomine teise tegelasega	<ul style="list-style-type: none"> • Õpilane oskab luua mängusisest kasutajaliidest. • Õpilane oskab luua dialogi tegelaste vahel, kasutades Unity füüsikasüsteemi funktsiooni raycasting
Mängu ehitamine ja käivitamine	<ul style="list-style-type: none"> • Õpilane oskab mängu eksportida mängumootorist eraldi rakenduseks.

3.6 Õppematerjali testimine ja täiendamine

Bakalaureusetöö jaoks loodud materjali testiti, et näha, kui tõhus see on. Õppematerjali läbis iseseisvalt üks inimene ning leidis protsessi käigus probleeme ja ettepanekuid. Testijal oli Unreal Engine'is mängu loomisel varasem kogemus, kuid Unityga oli vähe varasemaid kogemusi.

See peatükk annab ülevaate probleemidest ja tähelepanekutest õppematerjali käigus.

Juhendi läbimisel ei tekkinud läbijal suuri puudusi või arusaamatusi seoses õppematerjaliga. Puudusteks olid kohati hooletusvead, kus osade objektide loomisel soovitatud nimetust ei mainitud ning ühes kohas polnud viidet pakatile, mida õpetuses kasutati. Tehti mitmeid soovitusi, näiteks veetis osaleja veidi rohkem aega maailma disaini peatükis, kuna kasutatud paletti tööriistu ei selgitatud ja tuli nendega iseseisvalt tutvuda. Lisaks soovitati teatud kohtadesse lisada pilte, et materjali läbimisel segadust vähendada. Testija leidis, et õpetust on lihtne järgida ja selle läbimiseks kulus kuus tundi. Kuna testijal on varajasem mängu loomise kogemus olemas teise mängumootoriga, on mängu loomiseks kuluv aeg suurem algaja mängulooja jaoks. Kõik õppematerjalis avastatud probleemid likvideeriti ja tehti täiendusi vastavalt testija soovitudele.

4. Kokkuvõte

Bakalaureusetöö eesmärk oli koostada õppematerjal Unity mängumootori õppimiseks, mille kohta oli eestikeelset materjali veel vähe. Juhendi sihtrühmaks on algajad mänguloojad, kellel puudub varasem kogemus mängumootoriga. Bakalaureusetöö eesmärgid saavutati edukalt - koostati õppematerjal ja selle valmimisel valmis kasutajal lihtne 2D arvutimäng. Käesolevas bakalaureusetöös tutvustati mängumootorit Unity ning selle kolme konkurenti Unreal Engine, Godot ja CryEngine. Lisaks anti ülevaade mootori Unity ajaloo kohta. Õppematerjali loomisel tõlgiti eesti keelde mängumootori Unity vajalikud terminid. Käesoleva juhendi eesmärk ja õpitulemused olid sõnastatud nii üldiselt kui ka peatükkide kaupa. On toodud ka ülevaade olemasolevatest ingliskeelsetest õppematerjalidest. Bakalaureusetöö käigus hinnati loodud õppematerjali. Testimine viidi läbi ühe inimesega ning materjali läbimise ja tagasiside põhjal tehti õppematerjalis vastavad parandused. Loodud õppematerjal on autori arvates sobilik kasutamiseks õppeasutustes kui ka selle väliselt iseseisvalt läbimiseks.

Viidatud kirjandus

[1] Adam Sinicki, What is Unity? Everything you need to know

<https://www.androidauthority.com/what-is-unity-1131558/> (20.03.2021)

[2]Unity vara ressursid

https://assetstore.unity.com/packages/templates/tutorials/2d-beginner-tutorial-resources-140167?_ga=2.227838558.2108848997.1632729040-1495475781.1627376286#description

(07.08.2022)

[3]Unity mängumootori vikipeedia lehekülg(07.08.2022)

[https://en.wikipedia.org/wiki/Unity_\(game_engine\)](https://en.wikipedia.org/wiki/Unity_(game_engine))

[4]Unity ametlik dokumentatsioon toetatud platvormide kohta(07.08.2022)

<https://docs.unity3d.com/2018.2/Documentation/Manual/UnityCloudBuildSupportedPlatforms.html>

[5] Alan Thorne, 2011, Game Engine Design and Implementation, lk 10-14.

<https://books.google.ee/books?isbn=0763784516>

[6] Marie Dealessandri, artikkel nimega “What is the best game engine: is Unity right for you”(07.08.2022)

<https://www.gamesindustry.biz/what-is-the-best-game-engine-is-unity-the-right-game-engine-for-you>

[7] Unity ametlikul kodulehel olevad mängumootori plaanid ja hinnad(07.08.2022)

<https://store.unity.com/#plans-individual>

[8] Jon Brodtkin, artikkel nimega “How Unity3D Became a Game-Development Beast”(07.08.2022)

<https://insights.dice.com/2013/06/03/how-unity3d-become-a-game-development-beast/>

[9]Peter Cohen, artikkel nimega “Unity 2.0 game engine now available”(07.08.2022)

<https://www.macworld.com/article/187693/unity-18.html>

[10]Artikkel nimega “Unity at 10: For better—or worse—game development has never been easier”(07.08.2022)

<https://arstechnica.com/gaming/2016/09/unity-at-10-for-better-or-worse-game-development-has-never-been-easier/>

[11]Artikkel nimega “Unity 3 brings very expensive dev tools at a very low price”(07.08.2022)

<https://arstechnica.com/information-technology/2010/09/unity-3-brings-very-expensive-dev-tools-at-a-very-low-price/?comments=1>

[12]Pressiteade nimega “Game developers, start your Unity 3D engines (interview)”(07.08.2022)

<https://venturebeat.com/2012/11/02/game-developers-start-your-unity-3d-engines-interview/>

[13]Artikkel nimega "Mobile game developer survey leans heavily toward iOS, Unity"(07.08.2022)

<https://www.gamedeveloper.com/audio/mobile-game-developer-survey-leans-heavily-toward-ios-unity>

[14]David Cohen, artikkel nimega "Mobile game developer survey leans heavily toward iOS, Unity"(07.08.2022)

<https://www.adweek.com/performance-marketing/unity-sdk-out-of-beta/>

[15]David Cohen, artikkel nimega "Facebook Developing New PC Gaming Platform; Teams Up With Unity Technologies"(07.08.2022)

<https://www.adweek.com/performance-marketing/facebook-developing-pc-gaming-platform-unity-technologies/>

[16]Greg Kumparak, artikkel nimega "Unity 5 Announced With Better Lighting, Better Audio, And "Early" Support For Plugin-Free Browser Games"(07.08.2022)

https://techcrunch.com/2014/03/18/unity-5-announced-with-early-support-for-plugin-free-browser-games/?guccounter=1&guce_referrer=aHR0cHM6Ly9lbi53aWtpcGVkaWEub3JnLw&guce_referrer_sig=AQAAAE-aUoV-gntYmesKvZf7VSzUyio11VSBdgTaLmpPehJSGzBIW4c81G3H1otYjMa-X2oHPBwLzuc2VMaXdhCfSjbBK-LHn0yOr-BAPfx_E-UhsGX76f0iB8-ftKuhSRFgXVRI6gAeXr3D4RqdXzf1LV-yP5OQsQO_Q2eXMNCgGGh

[17]James Batchelor, artikkel nimega "Unity dropping major updates in favour of date-based model"(07.08.2022)

<https://www.gamesindustry.biz/unity-dropping-major-updates-in-favour-of-date-based-model>

[18]Pressiteade nimega "Unity Technologies – The World's Leading Game Engine"(07.08.2022)

<https://www.nanalyze.com/2017/10/unity-technologies-leading-game-engine/>

[19]James Batchelor, pressiteade nimega "Unity 2018 detailed in GDC keynote"(07.08.2022)

<https://www.gamesindustry.biz/unity-2018-detailed-in-gdc-keynote>

[20]Pressiteade nimega "Unity files for IPO, reveals \$163 million loss for 2019 and 1.5 million monthly users"(07.08.2022)

<https://venturebeat.com/2020/08/24/unity-files-for-ipo-reveals-163-million-loss-for-2019-and-1-5-million-monthly-users/>

[21]Pressiteade nimega "Unity MARS Augmented and Mixed Reality authoring studio now available"(07.08.2022)

<https://www.auganix.org/unity-mars-augmented-and-mixed-reality-authoring-studio-now-available/>

[22]Jason Schreier,artikkel nimega "Unity to Use IPO to Move Past Games to Real-World Challenges"(07.07.2022)

<https://www.bloomberg.com/news/articles/2020-09-16/unity-to-use-ipo-to-expand-beyond-games-to-real-world-challenges>

- [23]Unreal Engine wikipedia lehekülg(07.08.2022)
https://en.wikipedia.org/wiki/Unreal_Engine
- [24]Ametlik Godoti dokumentatsioon, ülevaade Godoti põhimõistetest(07.08.2022)
https://docs.godotengine.org/en/stable/getting_started/introduction/key_concepts_overview.html
- [25]Ametlik Godoti dokumentatsioon,GDScripti põhitõed(07.08.2022)
https://docs.godotengine.org/en/stable/tutorials/scripting/gdscript/gdscript_basics.html
- [26]Ametlik Godoti dokumentatsioon,funktsioonide loend(07.08.2022)
https://docs.godotengine.org/en/stable/about/list_of_features.html
- [27]Unreal Engine vikipeedia lehekülg(07.08.2022)
<https://en.wikipedia.org/wiki/CryEngine>
- [28]James Batchelor,pressiteade nimega “Crytek adopts royalties model as CryEngine 5.5 arrives”(07.08.2022)
<https://www.gamesindustry.biz/crytek-adopts-royalties-model-as-cryengine-5-5-arrives>
- [29]Artikkel nimega “Unity vs Cryengine: 9 Point super comparison”(07.08.2022)
<https://vionixstudio.com/2020/01/17/unity-vs-cryengine-game-engine-comparison/>
- [30]Arvutimängude loomine ja disain koduleht keskkonnas Courses(07.08.2022)
<https://courses.cs.ut.ee/2021/gamedev/fall/Main/HomePage>
- [31]Mängumootorid koduleht keskkonnas Courses(07.08.2022)
<https://courses.cs.ut.ee/2020/ge/fall/Main/Unity>
- [32]Unity ametlikul kodulehel olevad juhendid(07.08.2022)
<https://learn.unity.com/>
- [33]Videojuhendite esitusloend keskkonnas YouTube(07.08.2022)
https://www.youtube.com/watch?v=nu5nyrB9U_o&list=PLPV2Kylb3jR4KLGCCAcIWQ5qHudKtYeP7
- [34]Unity ametlikul koduleheküljel olev juhend(07.08.2022)
<https://learn.unity.com/project/ruby-s-2d-rpg>

Lisad

Õppematerjal

Peatüki number	Peatüki pealkiri
1	Projekti ülesseadistamine
2	Mänguobjekti liikuma panemine
3	Maailma disain
4	Tegelase ja mänguobjektide interaktsioon mängu maailmaga
5	Tegelasele elude lisamine ja vastased mängumaailmas
6	Animatsioonid
7	Kaamera muutmine filmikaameraks
8	Tegelase viskerelva loomine
9	Visuaalne kasutajaliides
10	Dialogi loomine teise tegelasega
11	Mängu ehitamine ja käivitamine

Õppematerjal ja praktiline töö on kokkupakitult lisatud. Õppematerjal kannab nime Juhend.pdf.

Praktiline töö

Praktiline töö asub kaustas nimega Lõputöö.Mängu käivitamiseks klõpsata failile Lõputöö.exe.Mängu kinni panemiseks kasutage klahvikombinatsiooni **ALT+F4**.

Litsents

Lihtlitsents lõputöö reprodutseerimiseks ja üldsusele kättesaadavaks tegemiseks

Mina, Max Frolov,

1. annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) minu loodud teose “Eestikeelse õppematerjali loomine Unity mängumootorile”, mille juhendaja on Madis Vasser, reprodutseerimiseks eesmärgiga seda säilitada, sealhulgas lisada digitaalarhiivi DSpace kuni autoriõiguse kehtivuse lõppemiseni.
2. Annan Tartu Ülikoolile loa teha punktis 1 nimetatud teos üldsusele kättesaadavaks Tartu Ülikooli veebikeskkonna, sealhulgas digitaalarhiivi DSpace kaudu Creative Commons'i litsentsiga CC BY NC ND 4.0, mis lubab autorile viidates teost reprodutseerida, levitada ja üldsusele suunata ning keelab luua tuletatud teost ja kasutada teost ärieesmärgil, alates **08.08.2022** kuni autoriõiguse kehtivuse lõppemiseni.
3. Olen teadlik, et punktides 1 ja 2 nimetatud õigused jäävad alles ka autorile.
4. Kinnitan, et lihtlitsentsi andmisega ei riku ma teiste isikute intellektuaalomandi ega isikuandmete kaitse õigusaktidest tulenevaid õigusi.

Max Frolov

08.08.2022