

# Unreal Engine platforma za razvoj edukativnih video igara

## Unreal Engine platform for developing educative video games

Nikola Savanović, Marko Šarac, Dušan Stamenković, Saša Adamović  
Univerzitet Singidunum, Danijelova 32, Beograd, Srbija

**Sažetak**—U ovom istraživačkom radu bavimo se temom koja je vezana za razvijanje igara u okruženju Unreal Engine 4 platforme. Platforma obezbeđuje više dodatnih alata za razvoj video igara sa fleksibilnim interfejsima pogodnim za primenu u edukativnim svrhama na svim nivoima obrazovanja. Rad sadrži uvod u radno okruženje, testirane i preporučene alate koji su podobni za razvoj video igara sa svim kolaborativnim elementima zahtevanim u savremenim pristupima učenja. Na ovaj način, ukazali smo na platformu koja je podrška budućim trendovima učenja za primenu u tradicionalnim rutinama, metodologiji i predmetnim silabusima.

**Cljučne riječi** – unreal engine; edukativne igre; učenje

**Abstract** – In this reaserch we are dealing with the topic of game development in environment Unreal Engine 4 platform. It provides more additional tools for video games development with a flexible interfaces suitable for educational purposes on every education levels. Reaserch contains introduction to the work environment, tested and recomended tools for video games development with all collaborative elements required in modern approaches to learning. This way we pointed to the platform which will support future trends of learning for use in traditional routines, methodology, and subject syllabi.

**Keywords** – unreal engine; education games; learning

### I. UVOD

Unreal Engine 4, kao softver za razvijanje računarskih (PC) igrica je veoma dobar softver za razvijanje veoma kompleksnih igrica. Daje velike mogućnosti autorima igrica da izvrše najkompleksnije zadatke, vrlo dobru optimizaciju, a sistemski nije veoma zahtevna. Pre dvadeset godina razvijanje igrica je bilo veoma kompleksno. Nije bilo moguće razvijati igricu bez jakih računara, skupih grafičkih kartica i na kraju dobrih programera. Danas je to potpuno drugačije.

Zašto je ova tema bitna? Zašto odstupamo od „tradicionalnih“ metoda pravljenja igrica? Neka su od pitanja kojim smo se bavili u radu. Ideja i cilj ovog rada je da se predstavi akademskoj zajednici, u polju informatike i

računarstva, slika šta mogu postići sa jednim ili više alata. U ovom radu smo se osvrnuli na starija rešenja i način na koji su se razvijale igrice.

Ovakva studija je zasnovana na ideji kako olakšati studentima razvijanje video igara. Krajem 20. veka glavni problem za razvijanje igrica ležao je u budžetu kojim jedna kompanija, firma ili pojedinac raspolažu. Bez određenih sredstava nije bilo moguće razvijati video igru. Takva slika se i danas nije značajno promenila, ali je u današnje vreme tehnologija mnogo uznapredovala, lični računari imaju veoma velike performanse i to daje određenu mogućnost pojedincima da se istaknu u ovom polju.

Najskuplja igrica koja je razvijana u vreme pisanja ovog rada je *Destiny* (2014) koja je imala budžet od 500 miliona dolara, od kojih je 140 miliona bilo namenjeno timu za razvijanje igre, zatim, budžet veoma popularne igrice *Call of Duty: Modern Warfare 2* bio je 275 miliona dolara, gde je 50 miliona dolara bilo direktno za programere i developere, a 200 miliona dolara na marketing. Što se tiče veoma poznate igrice *GTA 5* budžet te igrice je bio 265 miliona dolara, gde je za developere bilo predviđeno 137 miliona dolara.

Često kompanije raspoloživim budžetom razvijaju svoje platforme za pravljenje video igara, da bi u kasnijim nastavcima video igre mogli lakše da realizuju nove ideje – imaju već postojeće šablone, mape, nivoe, kretanja i drugo. Na taj način sebi olakšavaju rad na unapređenju fizike kretanja, kretanja tela, raznim zvučnim i vizuelnim efektima itd. Razlozi zbog koga su navedeni iznosi razvoja igara biće razjašnjeni u svakom odeljku ovog rada.

*Unreal Engine* godinama razvija odličnu platformu za razvijanje video igara, konačno, kompanija *Epic Games*, koja je radila na najprodavanijoj igrici *Unreal*, ne tako davne 1998. godine, razvili su svoju potpuno besplatnu platformu za razvijanje igrica. Pristup platformi može svako da ima, potrebno je samo registrovati se na zvaničnom sajtu i preuzeti softver.

### II. PREGLED RAZVOJNIH OKRUŽENJA

U radu je predstavljen kratak pregled platforme koja je postala jedna od vodećih u svetu za razvoj igara. Problem koji

se javlja kod razvoja igara na platformi *Epic Games* je šta uraditi ukoliko nedostaje biblioteka ili skripta koja je potrebna za razvoj igre. Rešenja ovog problema će biti objašnjena u trećem poglavlju ovog rada.

Za potrebe ovog rada predstavljene su funkcionalnosti *Epic Games* platforme za razvoj igrica. *Epic Games* nudi veoma bogat sadržaj za početak projekta. Neke od mogućnosti su: postavljanje prvog lica, trećeg lica (sa ili bez sadržaja koji olakšava početak – osnovna mapa, razni objekti i sl.), leteći objekti, kretanje vozila i drugo. Mi smo se držali standardnog principa – početak igre u trećem licu sa osnovnim sadržajem, koje čine: karakter koji može da se kreće (sa osnovnim podešavanjima, što će biti dalje u radu prikazano), komponente kamere, uglova, pogleda; mišem je dozvoljena kontrola pogleda okoline što je takozvani *Starter Content*.



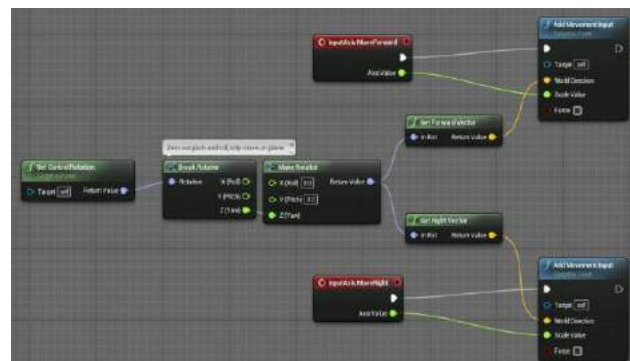
Sl. 1.1 Start 3D

Kao što su rekli Kjetil Gulbrandsen i Kjell Ivar Bekkerhus Storstein[1]:

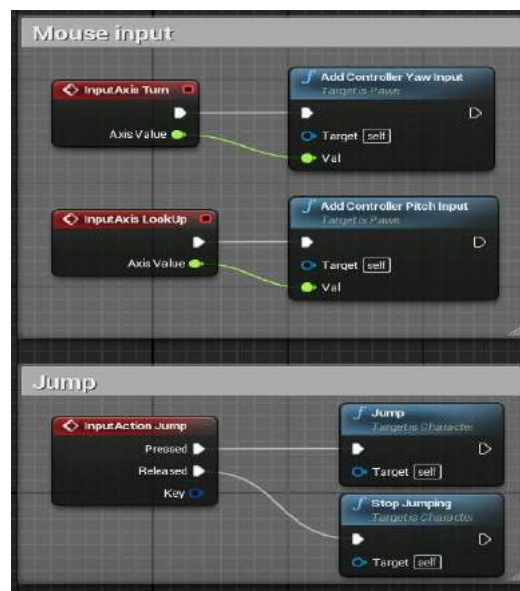
Osećaj igre je strukturan, koji sa različitim modulima doprinosi različitim aspektima igre[1]. To može biti renderovanje, detektovanje kolizije, fizika, umrežavanje, veštačka inteligencija i alati, što su samo neki [1].

Mi smo se u ovom preglednom radu oslonili na rad u *Blueprints Visual Scripting*. U *Unreal Engine 4* okruženju dostupna su dva tipa programiranja okruženja: *Blueprints Visual Scripting* i C++. *Blueprints Visual Scripting* sistem je u *Unreal Engine* okruženju kompletan sistem skriptovanja igre, zasnovano na konceptu korišćenja čvorova kako bi se napravili elementi igračkog okruženja sa svojim *Unreal* editorom[3].

Takav sistem je vrlo fleksibilan i veoma jak, koji pruža mogućnost dizajnerima da koriste virtualno ceo opseg koncepta i alata koji su generalno dostupni samo programerima[3]. Sa *Blueprints* skriptama mogu da se naprave prototipi tipa: postavljanje pravila igre, uslova, raznih ulaza (tastatura, miš, kontroleri), kontrolisanje svih alata u igri (zavisno od okruženja), postavljanje raznih pozicija (polazna tačka igrača, mesto odakle nastavlja).



Sl. 1.2 Blueprints kretanje igrača



Sl. 1.3 Ulaz miša i skok

Na slici 1.3 dat je prikaz vizuelne skripte gde je pomerajem miša moguće pomerati igrača. Prvi događaj (event) je zadužen za usmeravanje igrača mišem (levo i desno), dok je drugi događaj namenjen za pogled, odnosno, dozvoljava ulazu mišem da pravi pogled po mapi. Vrednosti se postavljaju na 0.000000 i one su obavezno tipa float.

Na drugom odeljku slike postoji događaj vezan za skok igrača, on se takođe definiše događajem, ali se u ovom slučaju razlikuje jer je ulaz sa tastature. Skok je na standardnom ulazu taster *space*; u slučaju skoka postoje dva događaja, odnosno jedna petlja:

1. Prvi događaj je događaj koji će se na pritisnuti taster svakako izvršiti i to je sam skok igrača, što će uraditi na prvom *update*-u. Moguće je postaviti *hold time* da skok igrača traje sve dok je pritisnut taster;

2. Drugi događaj je *Stop Jumping* gde se zaustavlja skok karaktera na određenom intervalu, a njegov *update* radi odmah nakon prvog događaja, osim ukoliko nije drugačije definisano. Ako nema drugih definisanih događaja, onda se izvršava *JumpMaxHoldTime*, što je trajanje skoka koje čuva početnu zadatu vrednos;

Prednost *Blueprints Visual Scripting*-a je svakako manje pisanje koda, ali je neophodno znati princip rada samih skripti kako objekat ili objekti u video igri ne bi upali u beskonačnu petlju. Dakle, možemo reći da postavljanje igre je već u samom početku vrlo jednostavno, sa odlično optimizovanom platformom, 3D karakterom i njegovim osobinama.

Jedan primer idealnog *Unreal Scripting*-a bi bio sledeći, gde se je prikazano zamišljeno kretanje, stanje mirovanja i stanje napada[1]:

```
class Karakter
function Kretanje ()
{
    //Nasumično kretanje
}
auto state Mirovanje
{
    Pocetak :
    Sleep (3);
    if (!PrimecenIgrac ())
    Kretanje ();
    else
    GotoState ('Napad');
    Goto 'Pocetak';
}
state Napad
{
    function Kretanje ()
    {
        //Igrač se kreće napred
    }
}
```

Takođe, za modelovanje karaktera (lice, telo i drugo) može se koristiti *MUDBOX Autodesk*. To je jedan od nezaobilaznih alata koji se lako može implementirati u *Unreal Engine 4*. Zbog svojih karakteristika došao je u sam vrh alata za pravljenje video igara. To je softver koji pruža dizajnerima splet alata i komandi za pravljenje i izmene geometrijskih tela, podrazumevano 3D, kao i za pravljenje tekstura mapa i nivoa.



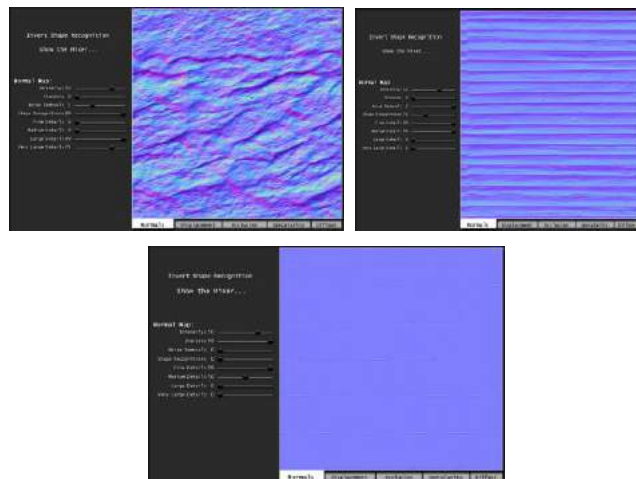
Sl. 1.4 MUDBOX Autodesk[4]

U ovom alatu moguće je modelovati 3D geometrijske oblike i teksture u najvišem mogućem kvalitetu. Sistemski zahtevi za poslednju verziju Autodesk Mudbox 2016 nisu veoma zahtevni. Potrebno je minimalno 4GB RAM memorije, 64-bitna arhitektura procesora i matične ploče, ali ono što je karakteristično za sam softver je da je potreban, ali ne i potpuno neophodan, miš sa tri tastera; to omogućava lakši rad u softveru.

Sledeći softver koji je takođe kompaktilan sa *Epic Games* platformom je *Crazy Bump* – alat, koji služi za podešavanje detalja mapa. Princip rada ovog softvera je sledeći, nakon importovane slike u softver, sliku se konvertuje u *grayscale* sliku i pokreće se kroz *shader*. Upravo *shader* daje određene detalje i korekcije na slici. Ono što je bitno za softver je da on generiše rezultate sa pravom informacijom. Primer dat u [4] prikazuje sledeće



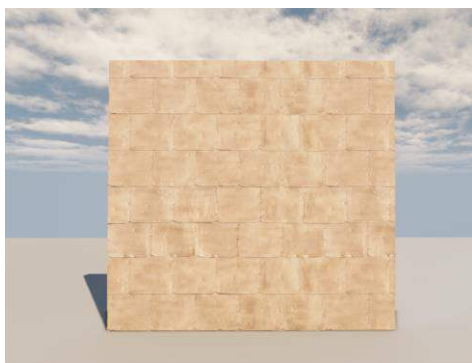
Sl. 1.5 Tekstura sa mnogo detalja[5]



Sl. 1.6 Uklanjanje šuma iz tekstura i njihovo prepoznavanje[5]

Na slici 1.5 je dat prikaz originalnih tekstura koje se nalaze u mapi. Korišćenjem određenih filtera za „razbijanje“ srednjih nivoa senčenja – *High pass*, kao i uklanjanjem šuma iz slike, može se doći do odličnih detalja u mapi. Paralelno sa softverom *Crazy Bump* koristi se obavezno *Photoshop* i preporučena poslednja verzija *CS6*.

Uz nekoliko jednostavnih kombinacija originalne slike i slike koja prolazi kroz filter *high pass* dolazi se do izostravanja teksture. U softveru o *Crazy Bump* se odlično na mapi vrše operacije kao što je uklanjanje šuma, prepoznavanje oblika (*shape recognition*) i obrada detalja. Takođe, s obzirom da su teksture vrlo male veličine, Unreal Engine nam dozvoljava da množimo teksturu sa konstantnom vrednošću 3, a sve to podešavamo kanalima. Izlaz iz takvog množenja za rezultat daje postavku teksture na željenu veličinu mape. Primera radi, mapa može da bude i 100 puta veća, a na samoj teksturi neće biti gubitaka.



Sl. 1.7 Metalna tekstura u igri[5]

Koliko sam softver doprinosi kvalitetu slike, ovu teksturu možemo uporediti sa metalnom teksturom sa ranije definisane slike 1.5. Alternativno softversko rešenje ovog programa je *AwesomeBump 4.0 Beta* verzija.

Još jedan vrlo koristan alat koji je potreban za razvoj igara je *MiKTeX* koji služi za prelom teksta na računaru. Ovaj alat je koristan jer čuva tekst u html formatu i veoma je pogodan za čuvanje tehničke dokumentacije, što je korisno za pripremu instalacije igre, preusmeravanje tekstualnih fajlova na putanje na kojoj određeni fajl mora da se nalazi. Jedan od primera gde je *MiKTeX* je primenu našao leži u postavljanju licence video igre u direktorijum u kome licenca mora da se nalazi.

### III. PRIMENA U EDUKATIVNIM SVRHAMA

U akademskom obrazovanju mogu naći primenu i studentima mogu olakšati rad u razvoju video igara. Studenti iz platforme koju je napravio *Epic Games* mogu dosta da nauče; mogu da shvate princip rada platforme za razvoj igara, da unaprede postojeću platformu ili da se upuste u razvoj sopstvenog rešenja što zahteva, kao što smo rekli u samom uvodu, veliki tim programera.

Takođe, kao što su autori naveli u radu [2], ovakve operacije možemo dovesti u vezu sa platformom, s obzirom da su potrebne i kriptološke i matematičke operacije.

Mi smo se u ovom radu oslonili samo na *Windows* platformu, ali je veoma važno napomenuti da je *Epic Games* platforma vrlo dobro optimizovana za pravljenje video igrica za mobilne uređaje i tablete, zasnovani na *Android OS*, kao i veoma solidnu optimizaciju za *Sony Playstation* i *Xbox*. U akademskom obrazovanju mogu imati višestruku primenu kao

što je shvatanje programiranja i principa kako ono što napišu u kodu (ukoliko se opredele za C++ programiranje) kako se reflektuje na samu animaciju.

U uvodnom delu smo pomenuli problem ukoliko fali neka skripta koja je potrebna da se izvršava, a ne sadrži je *Blueprints*, postoje dva rešenja; prvo rešenje je napraviti alternativnu skriptu u okviru *Blueprints*-a, koja će sadržati više petlji i uspeti da izvrši željenu operaciju, dok je druga mogućnost data od *Epic Games*-a da programer može sam da razvije svoju skriptu i integriše je u okruženje. Prednost ovakvog pristupa je da se u tom slučaju u video igri izbegavaju greške ili petlje iz kojih je skoro nemoguće izaći, ukoliko se loše postavje alternativna rešenja.

### IV. ZAKLJUČAK

U radu su pokazane mogućnosti za razvoj edukativnih igara na platformi Unreal Engine 4. Platforma je vrlo fleksibilna sa aspekta karakterizacije objekata unutar projekta razvijanog na njoj. Takođe, na osnovu eksperimentalnog rada možemo da potvrdimo da su platforma i alati u okviru nje dorasli zahtevima savremenih edukativnih okruženja u visokom obrazovanju, koji obezbeđuju podršku za metode modelovanja, dizajniranja i programiranja različitih scenarija za učenje neke naučne discipline.

Treba napomenuti da su upravo ovakve platforme poslužile za razvoj video igara koje su našle značajnu primenu u vojsci za uspešno savladavanje i učenje različitih scenarija na simuliranim terenima i vojnim akcijama.

### LITERATURA

- [1] Kjetil Gulbrandsen, Kjell Ivar Bekkerhus Storstein, "Evolutionary Game Prototyping using the Unreal Development Kit," Norwegian University of Science and Technology, Department of Computer and Information Science, June 2010. (*references*)
- [2] S. Adamović, M. Šarac, M. Veinović, M. Milosavljević, A. Jevremović, "An Interactive and Collaborative Approach to Teaching Cryptology" [Časopis] = IFETS // Journal of Educational Technology & Society, 1436-4522, 2014
- [3] Unreal Engine 4 docs [link: <https://docs.unrealengine.com/latest/INT/Engine/Blueprints/index.html>]
- [4] Autodesk MUDBOX [link: <http://www.autodesk.com/products/mudbox/overview>]
- [5] Brian – Environment artwork [link: [iamsparky.wordpress.com/crazybump/](http://iamsparky.wordpress.com/crazybump/)]