

Univerza  
v Ljubljani

Fakulteta  
*za gradbeništvo  
in geodezijo*



**LUKA GRADIŠAR**

**UPORABA BIM ZA APLIKACIJE VIRTUALNE  
RESNIČNOSTI V GRADBENIŠTVU**

**DIPLOMSKA NALOGA**

**UNIVERZITETNI ŠTUDIJSKI PROGRAM  
PRVE STOPNJE GRADBENIŠTVO**

Ljubljana, 2017

Univerza  
v Ljubljani

Fakulteta za  
*gradbeništvo in  
geodezijo*



Jamova cesta 2  
1000 Ljubljana, Slovenija  
telefon (01) 47 68 500  
faks (01) 42 50 681  
fgg@fgg.uni-lj.si

UNIVERZITETNI  
ŠTUDIJSKI PROGRAM  
PRVE STOPNJE  
GRADBENIŠTVO

Kandidat/-ka:

**LUKA GRADIŠAR**

## **UPORABA BIM ZA APLIKACIJE VIRTUALNE RESNIČNOSTI V GRADBENIŠTVU**

## **USING BIM FOR VIRTUAL REALITY APPLICATIONS IN CIVIL ENGINEERING**

**Mentor/-ica:**

doc. dr. Tomo Cerovšek

**Predsednik komisije:**

**Somentor/-ica:**

**Član komisije:**

## **STRAN ZA POPRAVKE**

**Stran z napako**

**Vrstica z napako**

**Namesto**

**Naj bo**

*»Ta stran je namenoma prazna«*

## **BIBLIOGRAFSKO – DOKUMENTACIJSKA STRAN IN IZVLEČEK**

**UDK:** UDK 004.946:624(043.2)

**Avtor:** Luka Gradišar

**Mentor:** doc. dr. Tomo Cerovšek

**Naslov:** Uporaba BIM za aplikacije virtualne resničnosti v gradbeništvu

**Tip dokumenta:** Diplomaska naloga – univerzitetni študij

**Obseg in oprema:** 26 str., 31 sl.

**Ključne besede:** BIM v gradbeništvu, virtualna resničnost, Unreal Engine, Autodesk Revit, vizualizacija v realnem času, vizualno programiranje

### **Izvleček:**

V diplomski nalogi se obravnava uporaba virtualne resničnosti v gradbeništvu. V nalogi so najprej analizirane možnosti in tehnologije, ki nudijo vizualizacijo v realnem času, ki je potrebna za doseganje virtualne ali navidezne resničnosti. Pri praktični uporabi je uporabljeno programsko okolje, Unreal Engine, ki podpira virtualno resničnost in programiranje okolja. Omogoča tudi izvoz samostojne aplikacije za konkretno stavbo, oziroma virtualno okolje, ki omogoča uporabniku interakcijo z njem. Za namen diplomske naloge so bili raziskani in predstavljeni različni načini uporabe informacijskega modela zgradbe z uporabo programa Unreal Engine. Na podlagi tega je bila izdelana aplikacija v kateri so predstavljene nekatere rešitve, ki predstavljajo prototip uporabe okolja na gradbenih projektih. Preko izdelane aplikacije lahko uporabnik vstopi v interaktivno računalniško okolje in interaktivno upravlja nekatere funkcije ter strukturno pregleduje model stavbe. Rešitev je pomembna tako za naročnike kot ponudnike inženirskih storitev.

*»Ta stran je namenoma prazna«*

## **BIBLIOGRAPHIC– DOKUMENTALISTIC INFORMATION AND ABSTRACT**

**UDK:** UDK 004.946:624(043.2)

**Author:** Luka Gradišar

**Supervisor:** assist. prof. Tomo Cerovšek, Ph.D.

**Title:** Using BIM for virtual reality applications in civil engineering

**Document type:** Graduation Thesis – University studies

**Scope and tools:** 26 p., 31 fig.

**Keywords:** BIM in civil engineering, virtual reality, Unreal Engine, Autodesk Revit, real-time visualization, visual programming

### **Abstract:**

The thesis deals with the use of virtual reality in civil engineering. The thesis first analyzes options in technology that provide real-time visualization, which is needed to achieve virtual or augmented reality. For practical use, the software environment, Unreal Engine, is used, which supports virtual reality and programming of the environment. It also allows you to export a standalone application for a specific building, or a virtual environment, which enables the user to interact with it. For the purpose of the thesis, various ways of using the building information model, were explored in Unreal Engine and presented. On this basis, an application has been developed, in which some solutions are presented, which represent the prototype of the environment use on construction projects. Through the designed application, the user can enter the interactive computer environment and interactively manage certain functions and structurally inspect the building model. The solution is important for both users and engineering service providers.

*»Ta stran je namenoma prazna«*



## **ZAHVALA**

Za pomoč in svetovanje ob raziskovanju in pisanju diplomskega dela se iskreno zahvaljujem mentorju doc. dr. Tomu Cerovšku.

Zahvala gre tudi družini za podporo v času študija in pisanja diplomske naloge.

*»Ta stran je namenoma prazna«*

## KAZALO VSEBINE

<b>STRAN ZA POPRAVKE .....</b>	<b>I</b>
<b>BIBLIOGRAFSKO – DOKUMENTACIJSKA STRAN IN IZVLEČEK .....</b>	<b>III</b>
<b>BIBLIOGRAPHIC– DOKUMENTALISTIC INFORMATION AND ABSTRACT .....</b>	<b>V</b>
<b>ZAHVALA.....</b>	<b>VII</b>
<b>1 UVOD .....</b>	<b>1</b>
1.1 Opis problema .....	1
1.2 Namen in cilj.....	1
1.3 Metode dela.....	2
<b>2 ORODJA ZA IZDELAVO VIRTUALNEGA OKOLJA .....</b>	<b>3</b>
2.1 Virtualna resničnost.....	3
2.2 Vizualizacijski programi .....	3
2.2.1 Lumion .....	3
2.2.2 Twinmotion.....	3
2.2.3 Unity .....	4
2.2.4 Unreal Engine .....	4
2.3 Primerjava vizualizacijskih programov .....	4
2.4 BIM orodje .....	5
2.5 Pomožni programi .....	5
<b>3 APLIKACIJE .....</b>	<b>6</b>
3.1 Svetlobne študije .....	6
3.2 Interakcija z okoljem.....	8
3.3 Spreminjanje materialov površin .....	9
3.4 Spreminjanje opreme .....	9
3.5 Gradnja okolja v virtualnem urejevalniku.....	10
3.6 Uporaba opisov in oznak.....	12
3.7 Prikaz konstrukcijskih elementov .....	13
<b>4 PREDSTAVITEV APLIKACIJE V PROGRAMU UNREAL ENGINE .....</b>	<b>16</b>
4.1 Zahteve strojne opreme .....	16
4.2 Odpiranje in zapiranje stavbnega pohištva .....	17
4.3 Stikala za luči .....	19
4.4 Prikaz konstrukcijskih slojev .....	20
4.5 Spreminjanje materiala.....	22
<b>5 ZAKLJUČEK.....</b>	<b>24</b>
<b>VIRI.....</b>	<b>25</b>

*»Ta stran je namenoma prazna«*

## KAZALO SLIK

Slika 1: Povezava med BIM in VR .....	1
Slika 2: Rešitev povezave med BIM in VR .....	1
Slika 3: Pregled osenčenja pod izbranim kotom sonca .....	6
Slika 4: Prikaz prižiganja luči s stikalom .....	7
Slika 5: Odpiranje in zapiranje vrat.....	8
Slika 6: Odpiranje in zapiranje senčil.....	8
Slika 7: Primer izbire in primerjave različnih parketov preko uporabniškega vmesnika .....	9
Slika 8: Prikaz spreminjanja pohištva v virtualnem okolju [12] .....	10
Slika 9: Izbira in postavitev elementov v virtualnem urejevalniku [16] .....	10
Slika 10: Prilagajanje dimenzij elementa v virtualnem urejevalniku [17].....	11
Slika 11: Dodajanje materiala elementom v virtualnem urejevalniku [17].....	11
Slika 12: Prikaz 3D armaturnega načrta z dodatnimi opisi .....	12
Slika 13: Primer prikaza različnih konstrukcijskih slojev preko uporabniškega vmesnika .....	13
Slika 14: Prikaz konstrukcije .....	14
Slika 15: Prikaz izolacije med konstrukcijskimi elementi .....	14
Slika 16: Prikaz konstrukcije in izolacije obdane s ploščami .....	15
Slika 17: Prikaz fasadne izolacije .....	15
Slika 18: Prikaz zaključnih slojev .....	15
Slika 19: Prikaz območja v katerem program zazna uporabnika .....	17
Slika 20: Prikaz kode, ki zazna uporabnika v definiranem območju in mu omogoči vnos .....	17
Slika 21: Prikaz kode, ki omogoča odpiranje in zapiranje vrat.....	18
Slika 22: Prikaz kode za drsna vrata .....	18
Slika 23: Prikaz kode, ki zaznava klik na stikalo .....	19
Slika 24: Prikaz kode, ki prižiga ali ugaša luči .....	19
Slika 25: Primer izbire svetil, za določno stikalo .....	20
Slika 26: Prikaz kode, ki preko klika na gumb spreminja vidljivost slojev z izbrano oznako ..	20
Slika 27: Primer podajanja oznake elementu .....	21
Slika 28: Prikaz kode, ki omogoča prikaz uporabniškega vmesnika .....	21
Slika 29: Prikaz kode, ki podaja vrednost spremenljivki ob sprožitvi dogodka .....	22
Slika 30: Prikaz kode, ki spreminja vidljivost podanega drsnika z materiali .....	23
Slika 31: Prikaz kode, ki spremeni material elementu z določeno oznako .....	23

*»Ta stran je namenoma prazna«*

## 1 UVOD

### 1.1 Opis problema

Arhitekturno vizualizacijo so običajno predstavljali fotorealistični renderji slikovnega ali video formata, kateri so zahtevali daljše čase generiranja slike. Želja je bila za hitrejše doseganje rezultatov z ohranjanjem fotorealizma. Razvoj zmogljivejše programske in strojne opreme je omogočil doseganje fotorealistične vizualizacije, ki ne zahteva dolgih časov generiranja slike. Z programsko opremo, ki ponuja tehnologijo vizualizacije v realnem času je arhitekturna vizualizacija postala relativno enostavnejša in hitrejša, saj se popravke opazuje brez čakanja.[1] Naslednjo stopnjo vizualizacije predstavlja virtualno interaktivno okolje, katerega je možno deliti z uporabnikom.

Virtualna resničnost omogoča uporabniku vstop v virtualno okolje, v katerem lahko prosto raziskuje objekt in dejansko spozna počutje v njem. To omogoča pregled nad končanim izdelkom ter možnost odpravljanja pomanjkljivosti in napak pred začetkom gradnje. Uporaba navidezne resničnosti predstavlja zahtevnejši problem, saj zahteva natančno zaznavanje okolice. Z uporabo dodatnih orodij za zaznavanje okolice in uporabnika, se lahko navidezna resničnost uporablja na gradbišču, saj se lahko prikaže končani objekt ali konstrukcijski detajl, kar pripomore k natančnejši izvedbi.

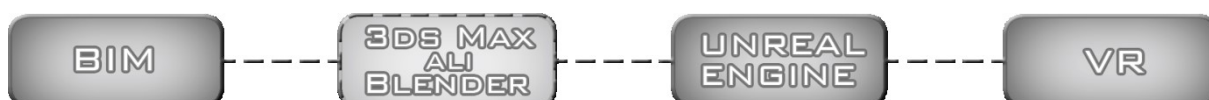
Naročniku oziroma uporabniku to predstavlja vključitev v proces projektiranja ali gradnje. Izdelano aplikacijo lahko projektant deli z naročnikom ali v primeru nezadostne opreme naročnik pristopi k projektantu, pri kateremu lahko nato raziskuje virtualno okolje. Za projektanta to predstavlja dodatno delo izdelave aplikacije, oziroma strošek v primeru zunanjega ponudnika storitev, vendar pa se s tem omogoči hitrejša povezava med dopolnjevanji objekta po željah naročnika. Projektantu tako ni potrebno predstaviti zasnove z uporabo slik in posnetkov, preko katerih naročnik ne more pridobiti pravilne predstave in počutja v objektu. Na gradbišču pa se lahko preko 3D vizualizacije ali navidezne resničnosti pomaga k lažji predstavitvi detajlov, kar običajno ni možno doseči preko 2D načrtov.

### 1.2 Namen in cilj

Za obravnavo aplikacije virtualne resničnosti v gradbeništvu je potrebno raziskati programe, kateri nudijo tehnologijo vizualizacije v realnem času, ki pa je potrebna za doseganje virtualne ali navidezne resničnosti. Za ugotavljanje možnih načinov uporabe vizualizacijskih programov v gradbeništvu je bilo potrebno raziskati možne rešitve na spletu in izdelati aplikacijo, preko katere se predstavi ugotovitve.



Slika 1: Povezava med BIM in VR



Slika 2: Rešitev povezave med BIM in VR

### **1.3 Metode dela**

Diplomska naloga zajema pregled literature za spoznavanje virtualne resničnosti. Za namen doseganja virtualne resničnosti so bili preko dokumentacije in uporabe raziskani različni vizualizacijski programi. Iz ugotovitev je bil izbran program v katerem so bila raziskana orodja in izdelana aplikacija. Za izdelavo aplikacije je bilo uporabljeno vizualno programiranje. V pomoč izdelave so bili pregledani spletni video posnetki, dokumentacija, in uporabniška pomoč. Preko tega procesa so bile raziskane aplikacije in orodja, katera se lahko uporabljajo v gradbeništvu.



## **2 ORODJA ZA IZDELAVO VIRTUALNEGA OKOLJA**

### **2.1 Virtualna resničnost**

Napredek "*real-time*" tehnologije je omogočil novo dimenzijo v vizualizaciji. Virtualna resničnost (VR) je interaktivno tri dimenzionalno računalniško grajeno okolje, katero je dosežno preko računalniške tehnologije, ki podpira generiranje slike v realnem času. Poleg tega znana tudi navidezna resničnost (AR), katera pa združuje virtualno okolje z pravim okoljem.[2][3] Trenutno najbolj uporabljeni programi za doseganje virtualne resničnosti predstavljajo programsko opremo namenjeno za ustvarjanje video iger, katere predstavnik je Unreal Engine.[1]

Za vstop v virtualno okolje se uporabljajo sistemi, ki spodbujajo človeške čute za iluzijo realnosti, kot so očala, rokavice, ročaji, vsesmerne pohodne steze itd. Najbolj znana predstavnik sistemov sta HTC Vive in Oculus Rift, ki omogočata tudi zaznavanje gibanja uporabnika. Poleg tega pa se na trgu pojavlja vedno več očal od različnih proizvajalcev. Za opazovanje virtualnega okolja se lahko uporabljajo nekateri mobilni telefoni ali tablice, vsaj vsebujejo smerne senzorje, ki omogočajo ogled 360° panoram in videov.[2][3]

Poleg uporabe za video igre, ima virtualna resničnost visok potencial v uporabi na ostalih področjih kot so znanost in medicina, arhitektura, izobraževanje, strojništvo, in gradbeništvo. Z razvojem opreme in novih tehnologij se manjša cena sistemov in virtualna resničnost postaja vedno bolj dosegljiva za ustanove ali posameznike.[2][3]

### **2.2 Vizualizacijski programi**

#### **2.2.1 Lumion**

Lumion je enostaven in učinkovit program namenjen arhitekturni vizualizaciji. Omogoča urejanje terena, letnih časov, lege sonca, vremena, poleg tega pa vsebuje knjižnico materialov in modelov, vendar osnovna (cenejša) verzija vsebuje zmanjšan obseg knjižnice. Velika prednost programa je zmožnost iterativnega dopolnjevanja modela, saj se lahko preko programskega dodatka poveže z programom Revit. Podpira izvoz slik, videov in 360° panorame. [6]

Cena standardne verzije programa znaša 1499€, Pro verzija pa 2999€, ki vsebuje povečano knjižnico modelov in materialov ter nekaj dodatnih funkcij.[6]

#### **2.2.2 Twinmotion**

Twinmotion deluje na podlagi tehnologije programa Unreal Engine, prav tako kot Lumion je namenjen predvsem arhitekturni vizualizaciji in vsebuje podobne funkcije: od urejanja terena, vegetacije, vremena, dodajanje prometa. Poleg tega pa nudi še dodajanje in prikaz faz gradnje. Omogočena je povezava v realnem času z programom Revit ali ArchiCad, kar pomeni, da se spreminjanje projekta posodablja tudi v samem programu. Poleg izvoza slik, videov in 360° videov, se lahko izvozi datoteko, ki se jo deli z uporabnikom in omogoča proste ali začrtane prehode skozi objekt. [5]

Cena enojna licence je 1650€, omrežne licence pa 2950€.[5]

### 2.2.3 Unity

Unity je najbolj razširjen program za ustvarjanje video iger, podprtih na več različnih sistemih. Unity je uporabljen na različnih področjih, saj je omogočeno spreminjanje izvirne kode in vsebuje programiranje v več programskih jezikih.[7]

Osebna verzija je zastoj za posameznike in podjetja, katera ne presegajo 100.000\$ letnega zaslužka ali zaslužka od prodaje aplikacije ustvarjene z programom Unity. V primeru presega pogojev je potrebno nadgraditi na Plus ali Pro verzijo, katere obsegajo dodatne storitve in so plačljive glede na naročnino.[7]

### 2.2.4 Unreal Engine

Unreal Engine, prvotno namenjen ustvarjanju video iger, vendar zaradi možnosti programiranja znotraj programa ter spreminjanja izvirne kode, se je njegova uporaba razširila na več področji. Program nudi poleg programiranja v programskem jeziku C++, tudi vizualno programiranje z tako imenovanim sistemom "*Blueprints*", ki omogoča enostavnejšo programiranje. Njegova uporaba je razširjena na področju virtualne resničnosti, ponuja tudi manipuliranje modelov v virtualni resničnosti znotraj urejevalnika.[4]

Uporaba programa je zastoj, v primeru prodaje aplikacije pa je zahtevano 5% zaslužka od prodaje po prvih 3000€.[4]

## 2.3 Primerjava vizualizacijskih programov

Primerjava programov predstavlja osebna mnenja in ugotovitve od spoznavanja z programi.

Prednosti programa Lumion in Twinmotion se izražajo v njuni namembnosti za arhitekturno vizualizacijo, saj omogočata direktno povezavo z nekaterimi BIM programi. Pri pomanjkljivostih v ustvarjanju kompleksnejših materialov se dopolnjujeta z lastno knjižnico različnih materialov in modelov, kateri zelo pripomorejo pri prihranku časa. Programa sta zelo uporabna za hitro poustvarjanje izjemnih vizualizacij. Vendar sta omejena z vidika uporabniške interakcije z okoljem, saj je njun končni rezultat slika ali video, z izjemo programa Twinmotion, kateri v najnovejši verziji ponuja zunanjo datoteko, ki uporabniku omogoča proste ali načrtane sprehode. Opazno je, da se z vsakim letom programi dopolnjujejo z novimi funkcijami in obstaja možnost, da bodo v prihodnosti nudili interaktivne rezultate.

Igralniški programi so zaradi zmožnosti programiranja kompleksnejši, vendar so zaradi prilagodljivosti namenjeni širši uporabi. Zaradi brezplačne dosegljivosti, dolge zgodovine in velike industrije ter široke skupnosti je na voljo veliko orodij in uporabniške pomoči, ki pripomorejo k lažjemu razumevanju in uporabljanju programov. Vsebujejo tudi trg materialov, modelov in ostalih izdelkov, katere prispevajo ostali uporabniki.

Iz različnih primerov na spletu je vidna vedno večja uporaba programa Unreal Engine za doseganje izjemnih arhitekturnih fotorealističnih vizualizacij v realnem času, saj zaradi možnosti shrambe predhodno preračunanih svetlobnih informacij na modele, zmanjša uporabo grafičnega procesorja, vendar to deluje za večinoma statično okolje, za dinamično spreminjanje okolja in svetlobe pa je potrebna kombinacija različnih načinov osvetlitve.[10]  
[11]

## **2.4 BIM orodje**

Model zgradbe, uporabljen v diplomski nalogi, je bil ustvarjen v programu Revit. Izbrana programska oprema je namenjena informacijskemu modeliranju zgradb (BIM). Poleg modeliranja, dodatna orodja tudi omogočajo upravljanje objektov, izdelavo raznih načrtov, popisov, itd. Izbran je bil zaradi enostavne dosegljivosti Autodesk-ove programske opreme za študente in ker predstavlja razširjeno BIM orodje.[8] Za izvoz 3D modela se uporabi funkcija *Export* z izbiro formata .fbx. Vseh dodatnih informacij pa še ni mogoče prenesti v vizualizacijske programe, temveč jih je potrebno samostojno uporabiti.

## **2.5 Pomožni programi**

Izvoženi model se lahko prilagodi v pomožnih programih, ki so namenjeni splošni računalniški grafiki. Program Revit ni namenjen običajnemu 3D modeliranju in se lahko pojavijo nepravilnosti. Zato je priporočena uporaba programov kot sta 3ds Max ali Blender, preko katerih se lahko odpravljajo geometrijske napake in dodajajo detajli. Poleg tega se lahko elementom pripiše standardne materiale, katere je možno uvoziti skupaj z teksturami v program Unreal Engine. Materiali izvoženi iz programa Revit pa niso združljivi z programom Unreal Engine. [8][9]

### 3 APLIKACIJE

Predstavitev objekta pred izgradnjo s pomočjo vizualizacije predstavlja pomembno vlogo za prikaz končnega rezultata naročniku. Poleg slik in videov je preko programa Unreal Engine mogoče izvoziti samostojno aplikacijo, katera ne zahteva nameščanja nobenega dodatnega programskega orodja. To predstavlja naslednjo stopnjo vizualizacije saj je uporabniku omogočeno samostojno raziskovanje objekta in interakcija z njem, tudi v virtualnem okolju. Uporabnik tako bolje spozna objekt in počutje v njem. Vizualizacija v realnem času predstavlja tudi močno orodje za projektanta. Preko tega lahko hitro preveri rezultate in popravi pomanjkljivosti.

Unreal Engine omogoča doseganje fotorealistične vizualizacije, ki ne zahteva veliko zmogljivost računalniške opreme. To je storjeno preko tehnologije programa, ki omogoča vnaprejšnjo generiranje svetlobe in senc, ki se zapišejo na posebne teksture. S tem se v končni aplikaciji razbremeni grafični procesor, saj ni potrebno preračunavanje svetlobe in senc, ki so že bile generirane.[11] Zaradi realistične interpretacije svetlobe je mogoče raziskovati različne vplive osvetlitve in osenčenja.

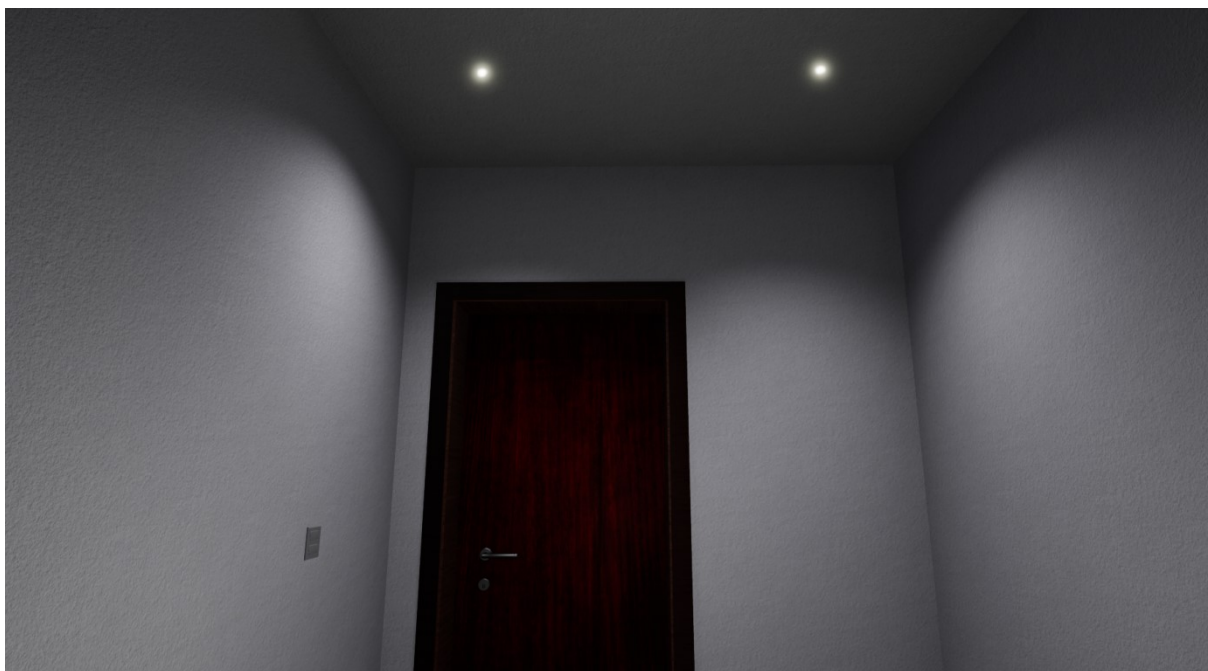
#### 3.1 Svetlobne študije

Zaradi možnosti spreminjanja lege sonca se lahko ugotavlja, kako se svetloba preliva v prostor. Iz lege sonca se lahko ugotavlja potrebno dolžino nadstreška glede na potrebno osenčenje v letnem času, ko ni zaželeno direktna svetloba v zasteklitev. V zimskem času ko je sonce v nižji legi pa je potreba po večji direktni svetlobi, katera poleg osvetlitve nudi tudi gretje prostorov. Poleg nadstreška je možna rešitev tudi listnato drevo, katero v poletnih časih nudi osenčenje, pozimi pa listje odpade. Programi omogočajo iterativno spreminjanje elementov, ki nudijo osenčenje glede na določeno lego sonca. Projektant lahko spreminja dolžino nadstreška, vrsto in lego dreves, in pri tem preklaplja vidljivost listja. Poleg tega lahko naredi aplikacijo preko katere lahko uporabniki sami raziskujejo različne možnosti.



Slika 3: Pregled osenčenja pod izbranim kotom sonca

Za notranjo osvetljavo se lahko v programu Unreal Engine določijo dejanske vrednosti svetilnosti in temperature posameznih svetil. Preko tega se lahko ugotavlja osvetlitev prostorov pri slabi zunanji osvetljavi in glede na ugotovitve ponovno izbira vrsto svetil ali njihovo potrebno število. Pomanjkljivost programa se kaže v nezmožnosti vnašanja svetil z že določenimi vrednostnimi iz zunanjih programov, torej jih je potrebno samostojno vnašati, kar pa je naporno v primeru velikega števila svetil. Svetilom se lahko pripišejo stikala, katerim se lahko znotraj urejevalnika ugotavlja optimalno lego, glede na smer odpiranja vrat ali lego zidovja ter ostalih faktorjev.



Slika 4: Prikaz prižiganja luči s stikalom

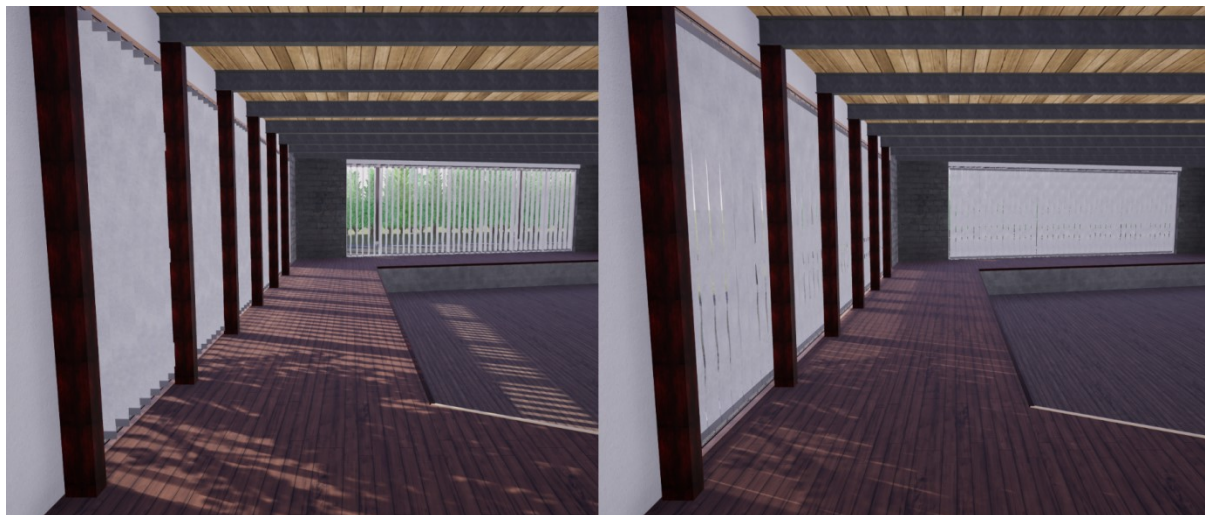


### 3.2 Interakcija z okoljem

Omenjeno je že bilo prižiganje svetil z stikalom. Uporabniku se lahko omogoči tudi odpiranje in zapiranje različnih elementov, kot so vrata, okna, drsna vrata, in senčila. Uporabnik se lahko sprehodi skozi objekt, bližje spozna funkcionalnost končanega objekta, in preko ugotovitev svetuje projektantu za popravke, kot so smer odpiranja in zapiranja vrat, vrsta senčil, izbira med običajnimi ali drsnimi vrati.



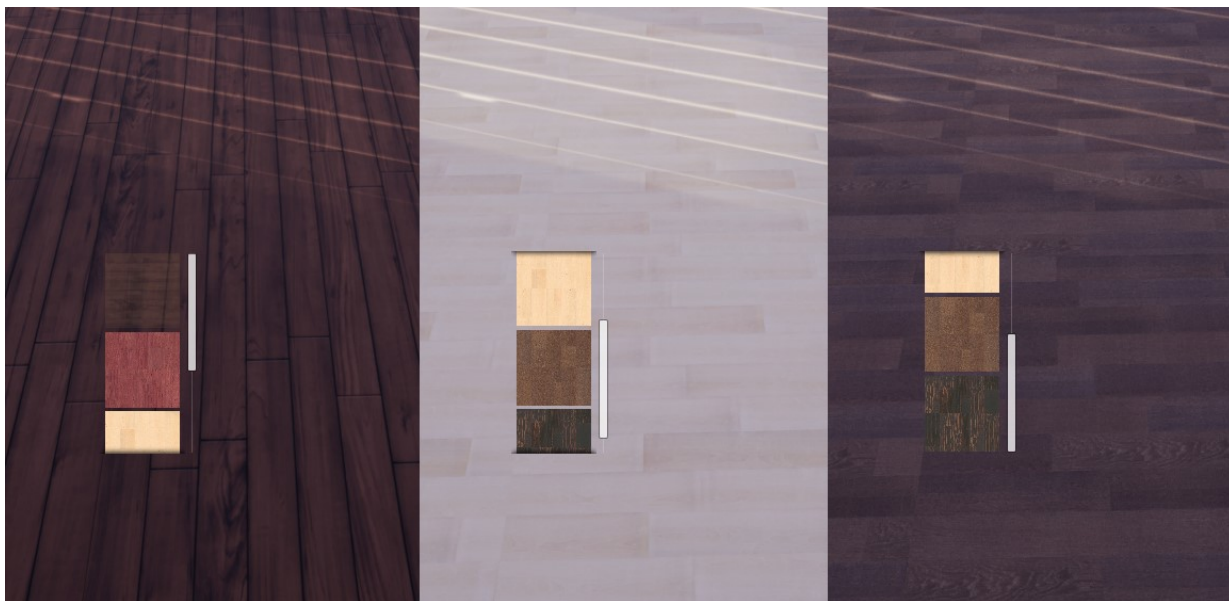
Slika 5: Odpiranje in zapiranje vrat



Slika 6: Odpiranje in zapiranje senčil

### 3.3 Spreminjanje materialov površin

Uporabniku je omogočeno samostojno spreminjanje materialov izbranih površin. Preko tega lahko uporabnik preučuje različne videze končnega rezultata. Aplikacija lahko vsebuje razširjeno zbirko materialov od različnih proizvajalcev s podanimi opisi in cenami. To predstavlja hitro rešitev preverjanja videza in odločanja za kupovanje izbranih materialov ter odpravljanje dvomov izgleda objekta po končani gradnji.



Slika 7: Primer izbire in primerjave različnih parketov preko uporabniškega vmesnika

### 3.4 Spreminjanje opreme

Uporabniku je omogočeno spreminjanje in premikanje različne opreme ali pohištva. Tako lahko uporabnik vnaprej preoblikuje prostor po svojem okusu in se nato lažje odloča za videz po končani gradnji. Tako kot spreminjanje materiala, se lahko tudi spreminjanje opreme uporabi v komercialne namene, kjer bi uporabnik lahko izbral različno opremo iz širše knjižnice izdelkov od različnih proizvajalcev. Spletna stran BIMobject, ki predstavlja zbirko informacijskih modelov stavbnih izdelkov, katere prispevajo proizvajalci, je že predstavila vizijo pregleda modelov v navideznem oziroma virtualnem okolju. [15]



Slika 8: Prikaz spreminjanja pohištva v virtualnem okolju [12]

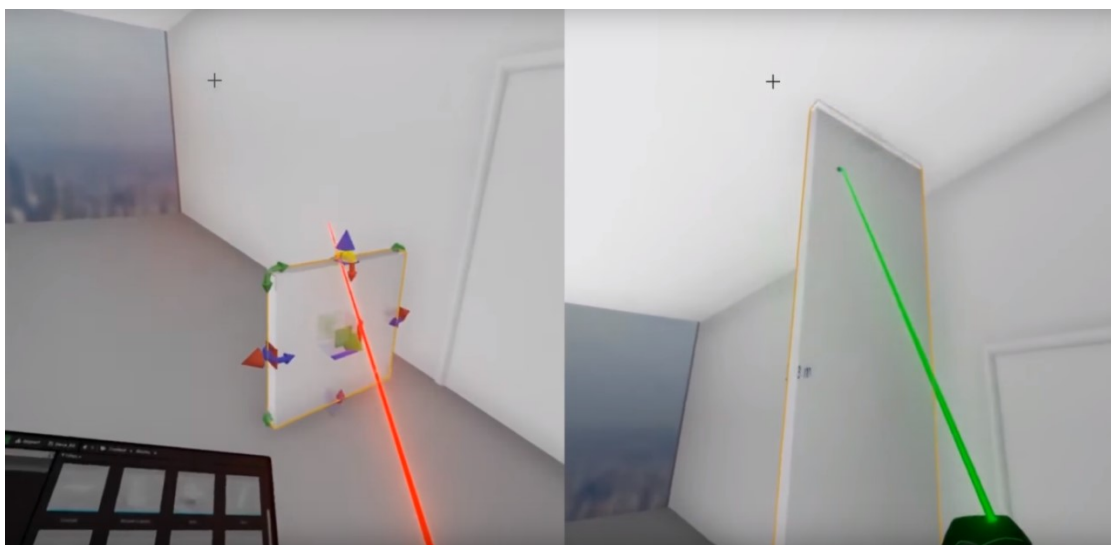
### 3.5 Gradnja okolja v virtualnem urejevalniku

Če je projektantu omogočen dostop do sistemov HTC Vive ali Oculus Rift, je v programu Unreal Engine omogočeno načrtovanje in ustvarjanje okolja v virtualni resničnosti z uporabo virtualnega urejevalnika. Preko orodij je mogoče postavljanje izbranih elementov, dimenzijsko prilagajanje elementov in dodajanje materialov izbranim elementom. Projektant lahko tako zasnuje idejne načrte, katere nato podrobneje obdela. Lahko pa že ureja končno podobo objekta v virtualnem urejevalniku in tako lažje prilagaja izgled v virtualni resničnosti, brez vmesnega preklapljanja oziroma preskakovanja iz "tipkovnice" v virtualno okolje.[16][17]

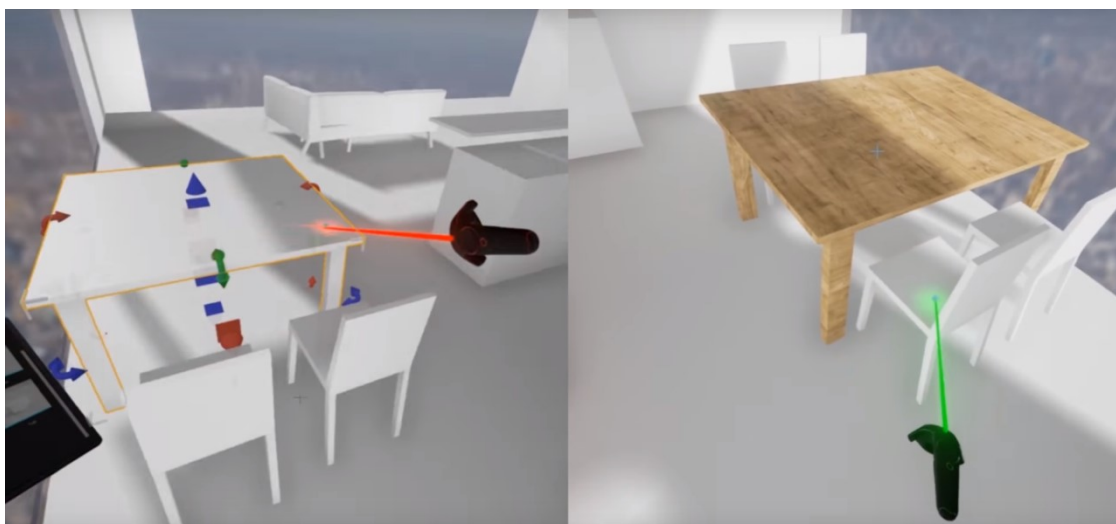


Slika 9: Izbira in postavitev elementov v virtualnem urejevalniku [16]





Slika 10: Prilagajanje dimenzij elementa v virtualnem urejevalniku [17]



Slika 11: Dodajanje materiala elementom v virtualnem urejevalniku [17]

### 3.6 Uporaba opisov in oznak

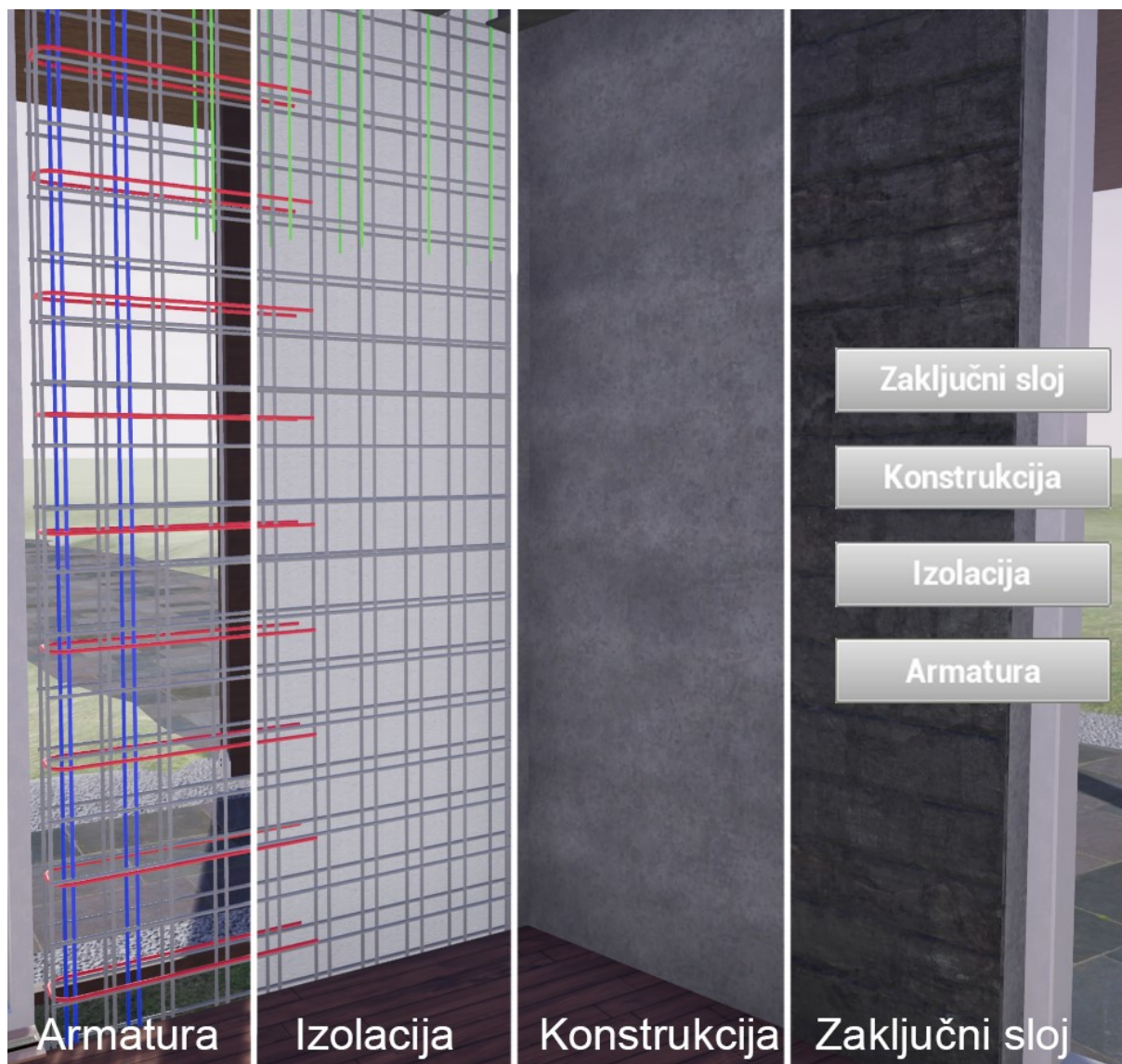
V namen boljše preglednosti in prikaza dodatnih informacij se lahko uporabljajo besedilne oznake. Opise se lahko prikaže poleg elementov ali pa se prikažejo uporabniku ob pogledu na element. Z oznakami se pomaga uporabniku k dodatni interpretaciji različnih elementov. Oznake pripomorejo k lažji vidljivosti manjših elementov ali inštalacij, da uporabniku ni potrebno dodatno iskanje elementov. V podanem primeru so se oznake uporabile za jasnejši vizualizacijski pregled armaturnih načrtov. Uporabniku je tako omogočena lažja predstava postavitve armature, kot pa iz 2D armaturnih načrtov.



Slika 12: Prikaz 3D armaturnega načrta z dodatnimi opisi

### 3.7 Prikaz konstrukcijskih elementov

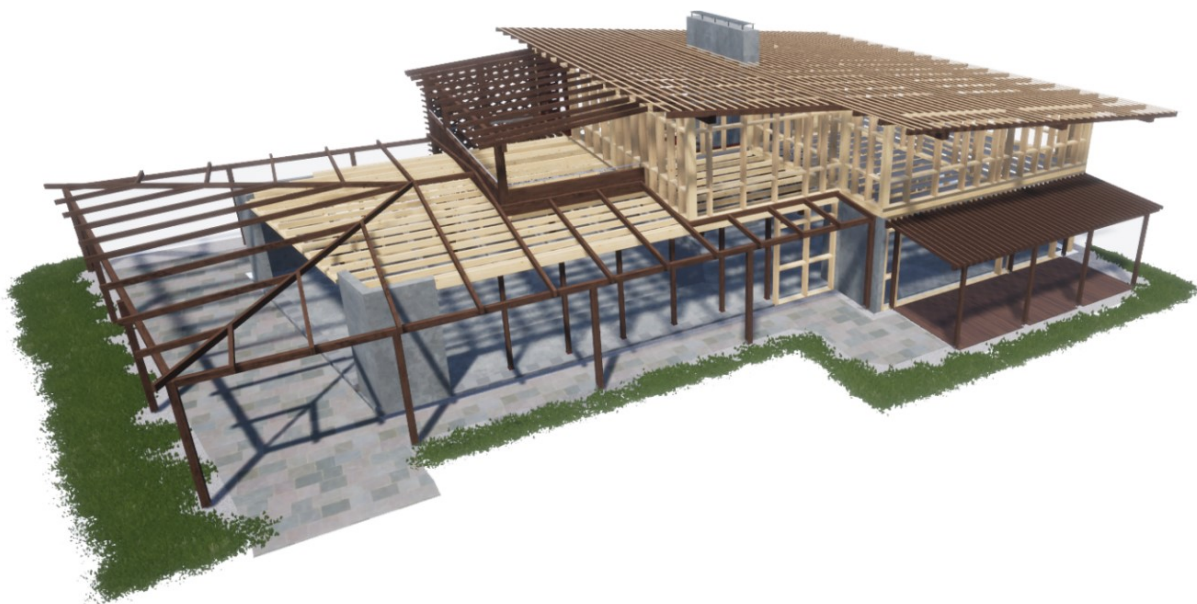
Preko uporabniškega vmesnika je mogoče podati uporabniku interaktivno prikazovanje vidljivosti različnih elementov. Omenjena funkcija bi bila predvsem uporabna na gradbišču, saj uporabnik lahko preklaplja med prikazi izbranih elementov, kot so konstrukcijski sloji, inštalacije, armaturni načrti itd. To omogoča podroben predhoden pregled nad objektom.



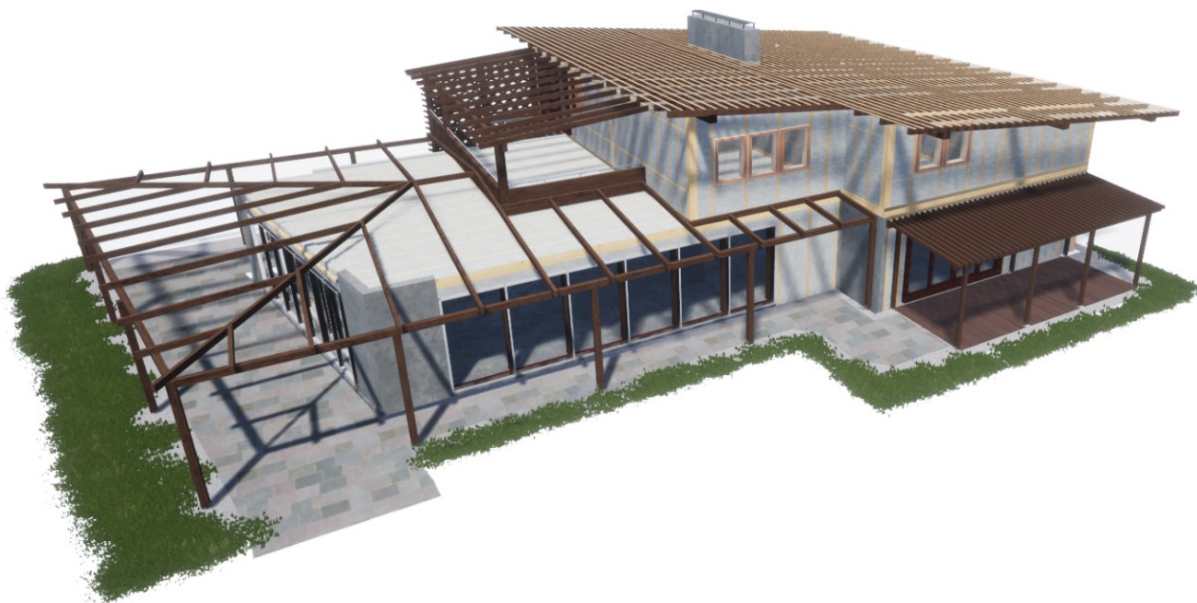
Slika 13: Primer prikaza različnih konstrukcijskih slojev preko uporabniškega vmesnika



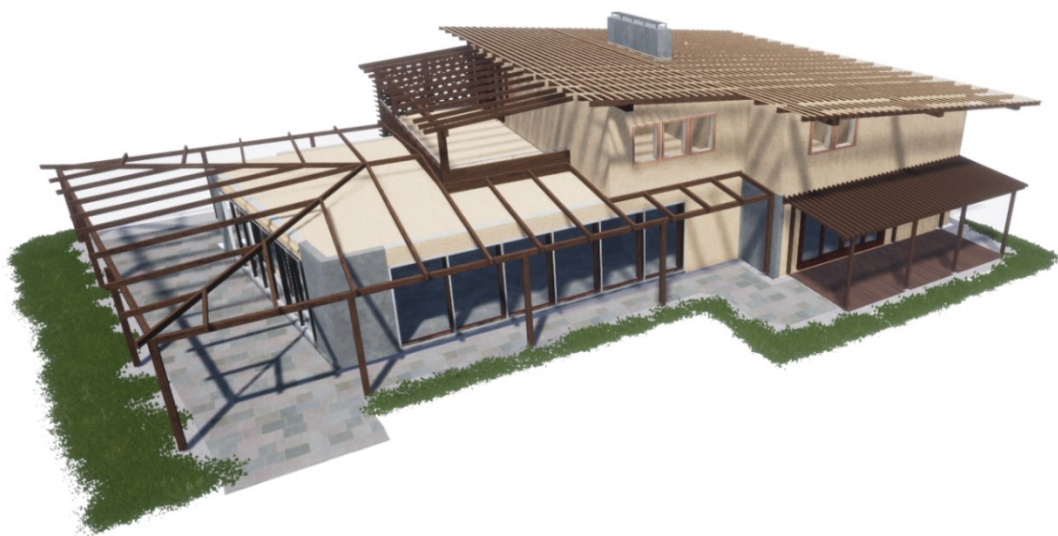
Funkcija prikazovanja različnih konstrukcijskih slojev omogoča prikaz poteka gradnje, saj se lahko prikazuje različne gradbene elemente, kakor si sledijo po fazah gradnje. Projektant oziroma izvajalec lahko tako poda natančna navodila, s katerimi si lahko gradbeniki pomagajo pri gradnji objekta ali pa za nadzor in kontrolo objekta.



Slika 14: Prikaz konstrukcije



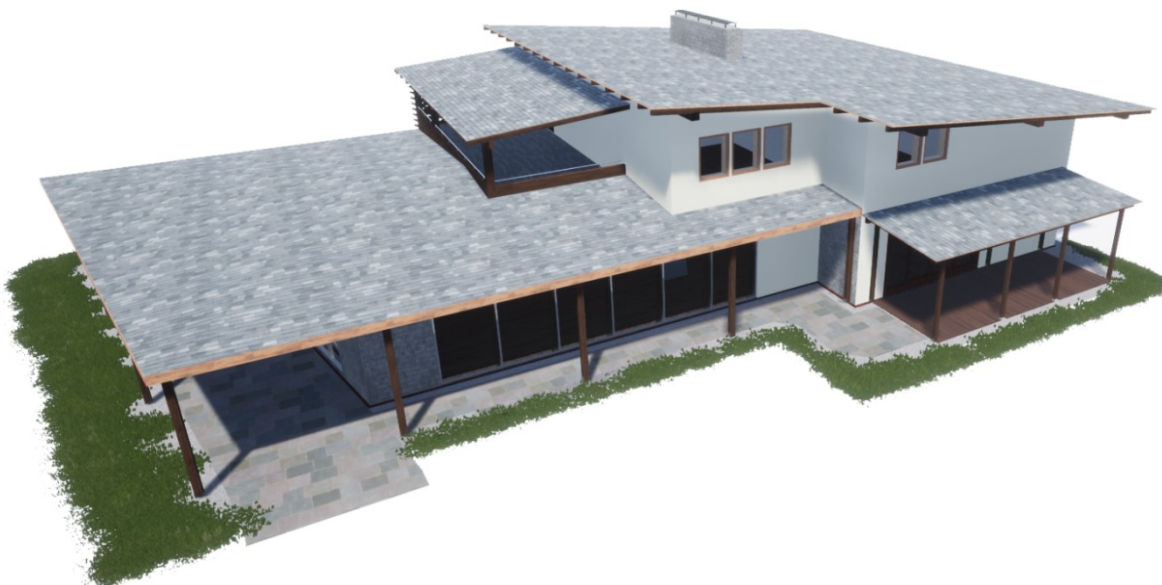
Slika 15: Prikaz izolacije med konstrukcijskimi elementi



Slika 16: Prikaz konstrukcije in izolacije obdane s ploščami



Slika 17: Prikaz fasadne izolacije



Slika 18: Prikaz zaključnih slojev

## 4 PREDSTAVITEV APLIKACIJE V PROGRAMU UNREAL ENGINE

V poglavju so predstavljene različne funkcije in programi, ki so bili ustvarjeni s pomočjo vizualnega programiranja *Blueprint* v programu Unreal Engine za namen raziskave možnih aplikacij.

### 4.1 Zahteve strojne opreme

Razvoj aplikacije je bil izveden na namiznem računalniku z opremo:

- Operacijski sistem: Windows 7 64-bit
- Procesor: Intel Core i7 2600K 3.4 GHz
- Grafični procesor: NVIDIA GeForce GTX 960
- Pomnilnik: 16 GB

Priporočena zahteve za razvoj aplikacij v programu Unreal Engine [18]:

- Operacijski sistem: Windows 7 64-bit ali Mac OS X 10.9.2 ali poznejše verzije
- Procesor: štiri jedrni Intel ali AMD procesorji, 2.5 GHz ali hitrejši
- Grafični procesor: grafične kartice, ki podpirajo DirectX 11
- Pomnilnik: 8 GB ali več

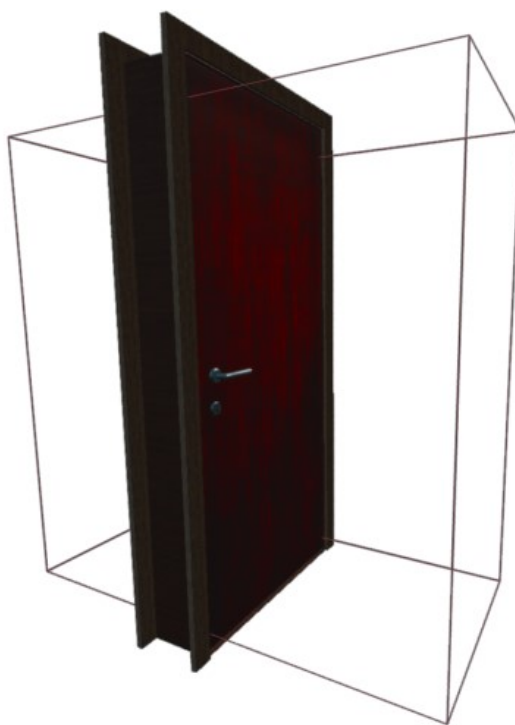
Priporočene zahteve za virtualno resničnost (HTC Vive) [19]:

- Operacijski sistem: Windows 7 ali poznejše verzije
- Procesor: Intel i5-4590 ali AMD FX 8350 ali boljši procesorji
- Grafični procesor: NVIDIA GeForce GTX 1060 ali AMD Radeon RX 480 ali boljše
- Pomnilnik: 4 GB ali več

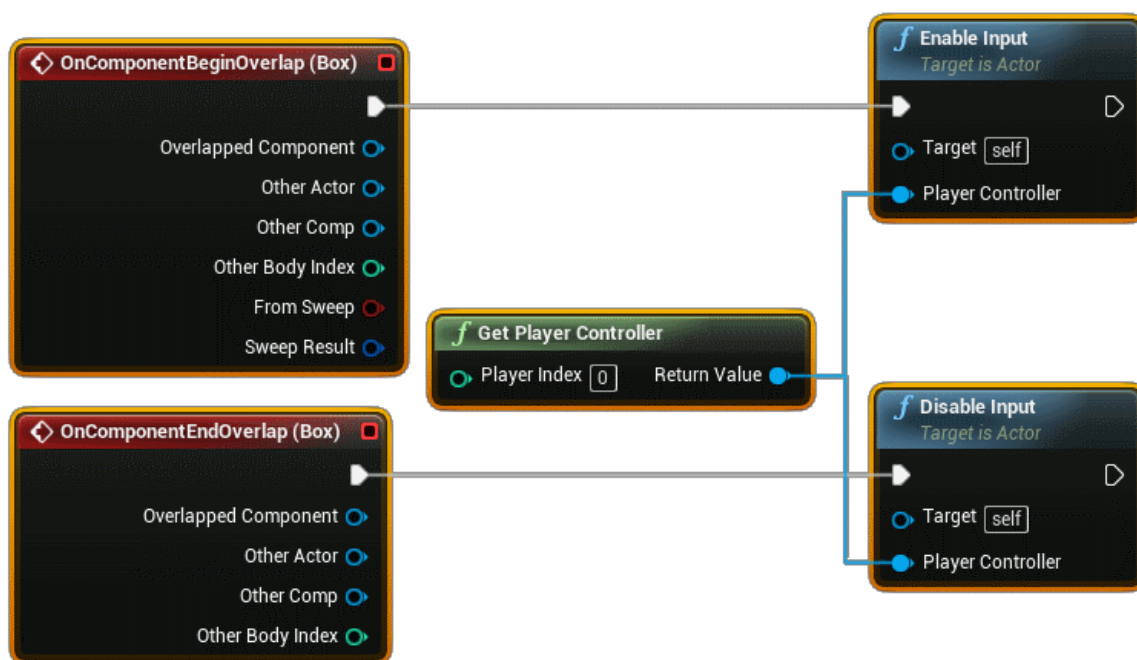
Zahteve izvožene aplikacije so odvisne od velikosti in kompleksnosti aplikacije. Splošno so za optimalno delovanje aplikacije priporočene podobne zahteve strojne opreme, kot za uporabo programa Unreal Engine, s poudarkom na grafičnem procesorju, ki naj ne bo integrirani grafični procesor, temveč grafična kartica novejših generacij. [18] Aplikacija bi morala delovati na večini namiznih sistemov, razlike se lahko pojavljajo v hitrosti delovanja, vendar je uporabniku še vedno omogočeno uporabljanje aplikacije. Virtualna resničnost in fotorealistična vizualizacija zahteva večjo zmogljivost grafičnega procesorja in je potrebna zmogljivejša oprema. Prikaz konstrukcijskih elementov in pregled detajlov pa ne predstavlja visokih potreb v grafičnem prikazu in se lahko aplikacija uporablja na prenosnih računalnikih za namen uporabe na gradbiščih.

## 4.2 Odpiranje in zapiranje stavbnega pohištva

Funkcija odpiranja in zapiranja stavbnega pohištva deluje preko definiranega območja v okolici elementa, v katerem program zaznava vnos od uporabnika. Prikazan je primer odpiranja in zapiranja vrat, vendar se lahko enak pristop uporabi za odpiranje oken, drsnih vrat, senčil, in podobnih elementov.



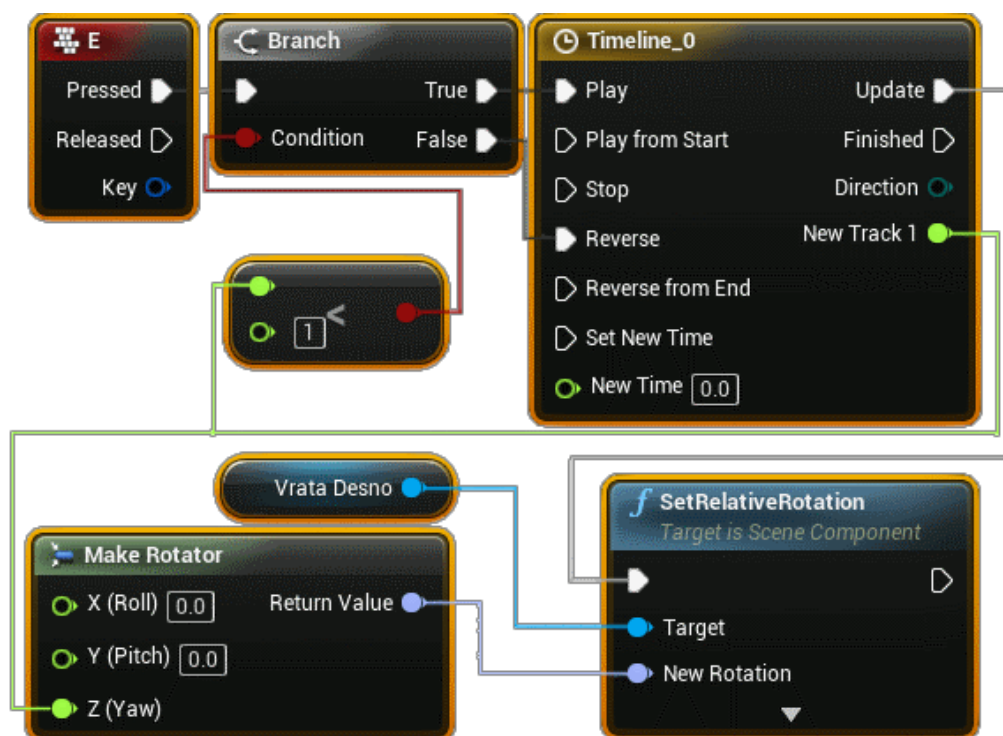
Slika 19: Prikaz območja v katerem program zazna uporabnika



Slika 20: Prikaz kode, ki zazna uporabnika v definiranem območju in mu omogoči vnos

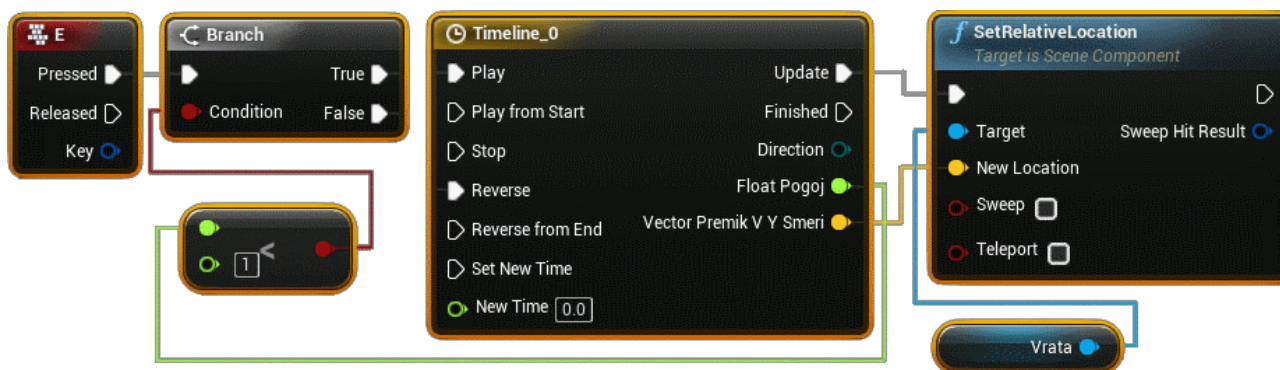


Za odpiranje in zapiranje vrat se poda poljuben vnos, v tem primeru se koda požene, ko program zazna vnos tipke. Potek se nato razdeli preko logičnega testa. Pogoji preverja če je vrednost časovne linije oziroma rotacije manjša kot 1, potem so vrata zaprta. Če so vrata zaprta se program nadaljuje na časovno linijo, v kateri je podana sprememba vrednosti iz 0 na 90 v 2,5 sekunde. Vrednost se prevede na rotacijo okoli z osi, ki predstavlja navpično os. Za zapiranje vrat se ponovno poda vnos in če se je rotacija že iztekla, pogoj zazna, da je vrednost časovne linije 90. Pogoj ni izpolnjen, zato se bo časovna linija predvajala vzvratno.



Slika 21: Prikaz kode, ki omogoča odpiranje in zapiranje vrat

Za drsna vrata pa je potrebno ustvariti časovno linijo vektorskega tipa v kateri se nato spreminja vrednost tisti koordinati v kateri se odpirajo drsna vrata. Poleg tega pa je za uporabo pogoja potrebna časovna linija tipa "float" v kateri se poda spremembo vrednosti iz 0 na neko pozitivno število, da pogoj pravilno deluje.

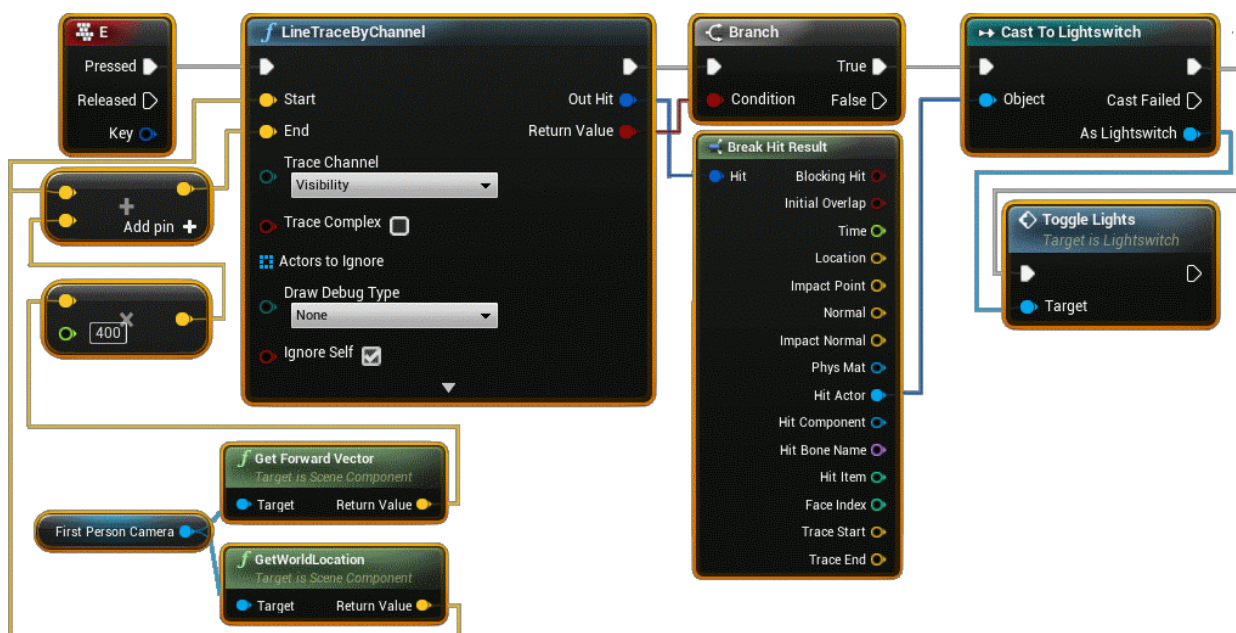


Slika 22: Prikaz kode za drsna vrata



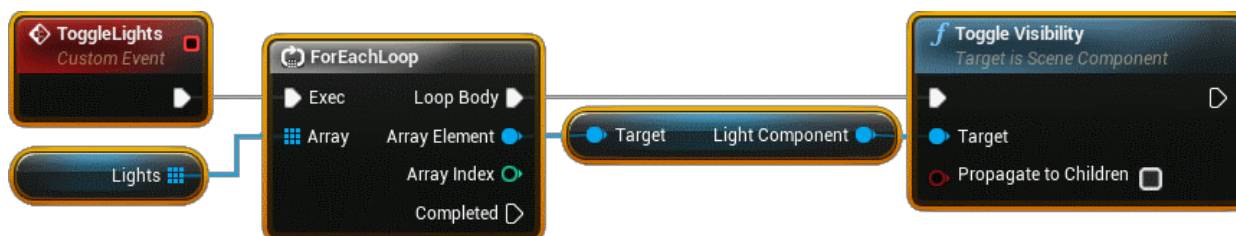
### 4.3 Stikala za luči

Aplikacijo stikala je potrebno razdeliti na dva dela. Za interakcijo s stikalom je potrebno zaznati smer pogleda uporabnika oziroma igralca, zato je potrebno prilagoditi program igralca. Ko program zazna vnosno tipko se preko lokacije in smernega vektorja začrta linija. Ko se bo linija sekala ali dotikala predpisanega elementa (v našem primeru stikala) se bo program nadaljeval in skliceval na drugi del preko posebnega dogodka, ki poveže programa.[13] Problem se pojavi ko program ne zazna vedno interakcije zaradi majhne velikosti stikala, ki se ga odpravi tako, da se poleg stikala doda širši element in se mu pripiše skrito vidljivost.



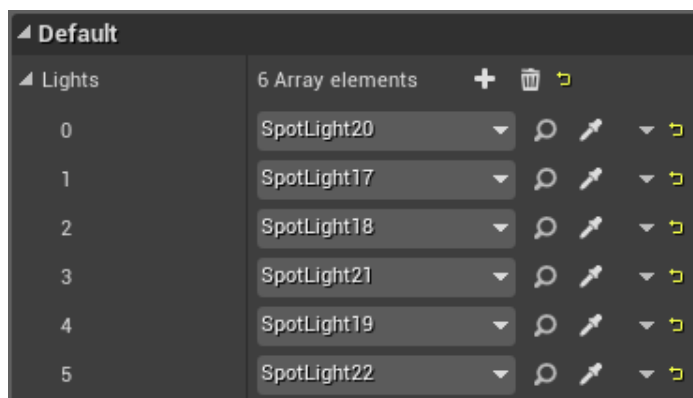
Slika 23: Prikaz kode, ki zaznava klik na stikalo

Potrebno je uvesti novo spremenljivo tipa "*Light*", ki omogoča izbiro svetil v urejevalniku. Ko se požene poseben dogodek, program preko zanke spreminja vidljivost za vsako predpisano luč v spremenljivki.



Slika 24: Prikaz kode, ki prižiga ali ugaša luči

Luči predpišemo tako, da se znotraj urejevalnika izbere stikalo in v podrobnostih doda število elementov in nato za vsakega predpiše izbrano svetilo.



Slika 25: Primer izbire svetil, za določno stikalo

#### 4.4 Prikaz konstrukcijskih slojev

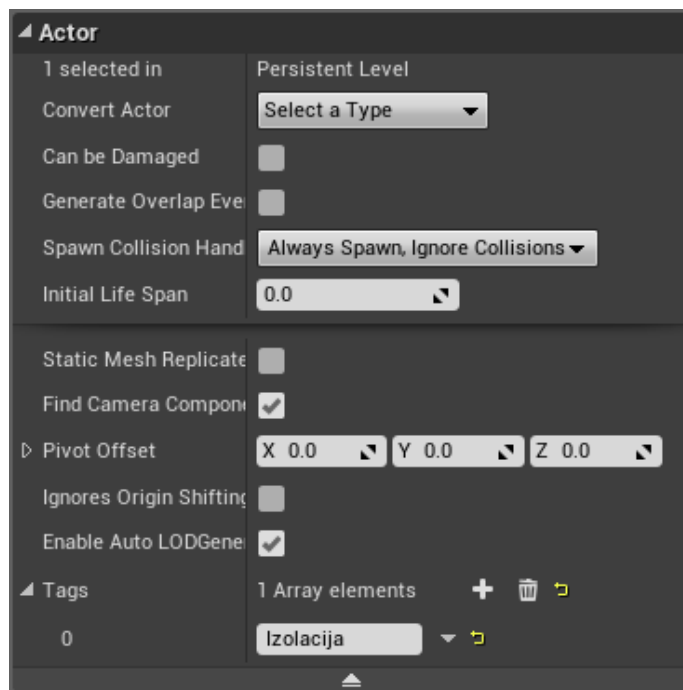
Za prikaz različnih konstrukcijskih slojev je potrebno pridobiti 3D model, kateri jih vsebuje. Da ni potrebno modeliranje vsakega sloja, lahko v programu Revit razdelimo elemente na posamezne sloje. To je mogoče preko funkcije *Create Parts* v oddelku *Create*, ki se prikaže ko so izbrani elementi.

Rešitev funkcije prikaza konstrukcijskih slojev je narejena s pomočjo uporabniškega vmesnika, preko katerega uporabnik z gumbi uravnava prikaz različnih slojev. Za kreiranje uporabniškega vmesnika se uporabi *Widget Blueprint*. Znotraj urejevalnika se v območje vnesejo gumbi, katere se lahko prosto razporedi po območju, ki bo viden uporabniku. Program se nato sklicuje na izbran gumb preko dogodka, ko uporabnik klikne nanj. Nato se uporabi funkcija, ki izbere vse elemente z določeno oznako (npr. Izolacija). Preko zanke se za vsak izbran element pošle informacija za vklapljanje ali izklapljanje vidljivosti.



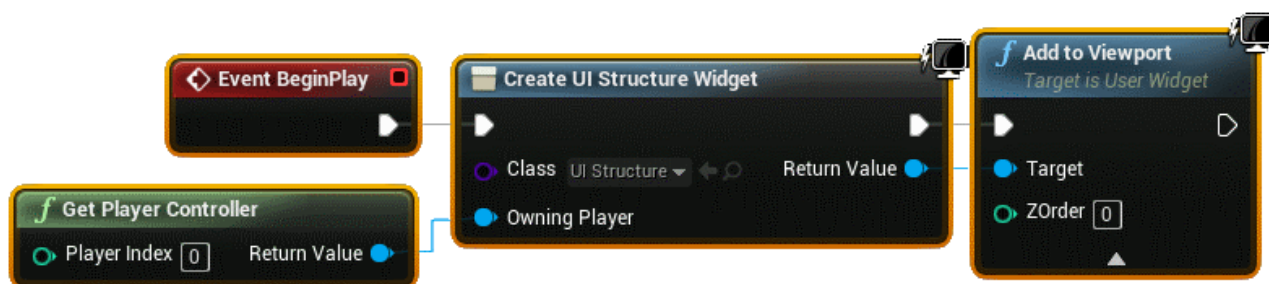
Slika 26: Prikaz kode, ki preko klika na gumb spreminja vidljivost slojev z izbrano oznako

Za podajanje oznak elementom je potrebno v urejevalniku izbrati elemente in v podrobnostih v območju *Actor* dodati oznako. Torej z klikom na gumb lahko uporabnik vklaplja ali izklaplja vidljivost posameznih elementov. S tem postopkom lahko dosežemo prikaz poteka gradnje stavbe.



Slika 27: Primer podajanja oznake elementu

Za prikaz uporabniškega vmesnika je potrebno ustvariti nov *Blueprint*, katerega zvrst je HUD (Heads-Up-Display). Znotraj podamo dogodek, ko se aplikacija zažene se bo ustvaril in prikazal izbrani uporabniški vmesnik. Poleg tega pa je potrebno še izbrati izdelani HUD v *Blueprint*, katerega zvrst je *GameMode*.



Slika 28: Prikaz kode, ki omogoča prikaz uporabniškega vmesnika

## 4.5 Spreminjanje materiala

Spreminjanje materiala nekega elementa je omogočeno preko uporabniškega vmesnika, ki se pojavi, ko je uporabnik v bližini obravnavanega elementa. Program je razdeljen na tri dele. V prvem delu je potrebno definirati novo spremenljivo vrste *Boolean*, ki predstavlja podatek z dvema vrednostnima *"True"* ali *"False"*.<sup>[14]</sup> Spremenljivka je ustvarjena v programu igralca (*FirstPersonCharacter*), na katero se nato program sklicuje.

V drugem delu je potrebno ustvariti nov *Blueprint* v katerem definiramo območje, ki zazna če je uporabnik v njem. Če je uporabnik v območju se bo program skliceval na spremenljivko in ji podal vrednost *"True"*. Ko bo uporabnik izven območja pa bo vrednost spremenljivke *"False"*.

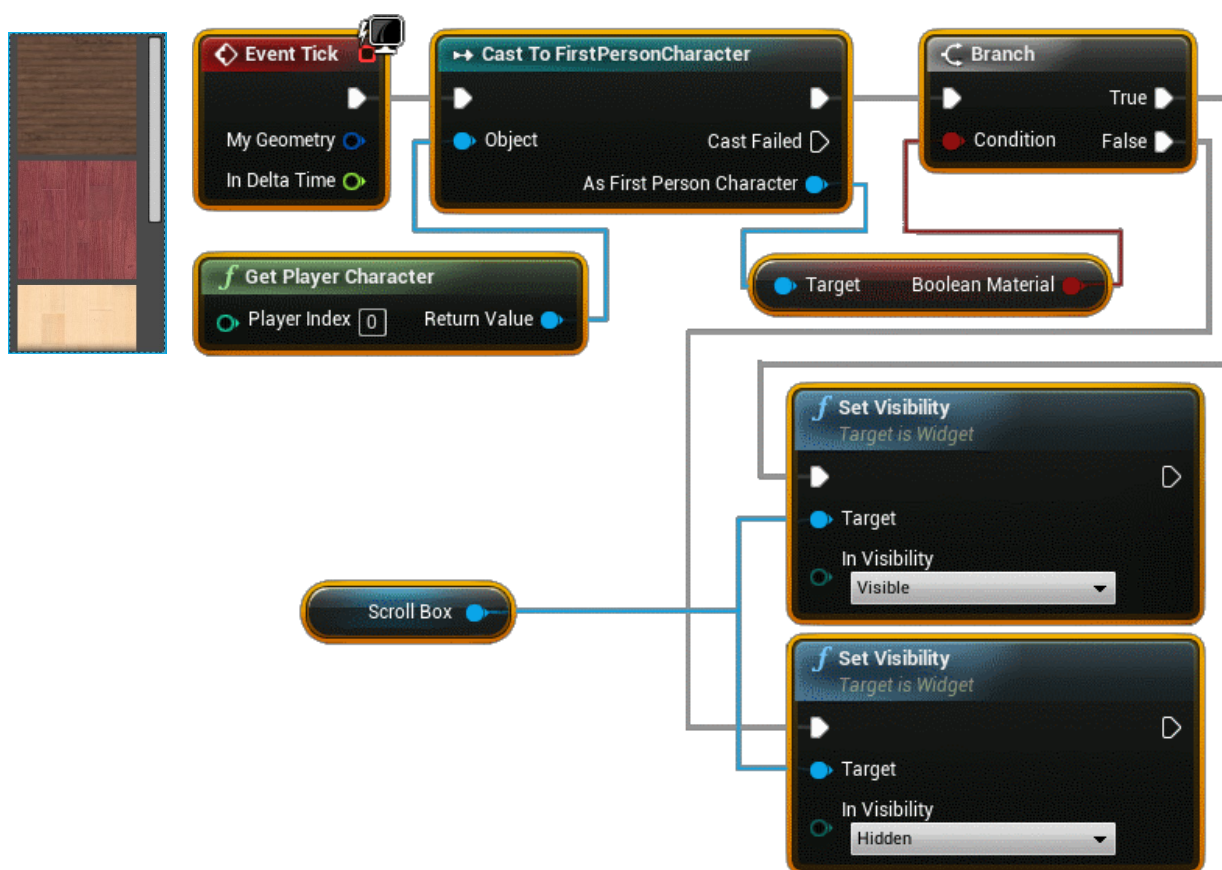


Slika 29: Prikaz kode, ki podaja vrednost spremenljivki ob sprožitvi dogodka

V tretjem delu je potrebno ustvariti uporabniški vmesnik *Widget Blueprint*, če ga aplikacija še ne vsebuje, drugače pa se doda novo vsebino v obstoječ uporabniški vmesnik.

Za izbiro novega materiala je uporabljen drsnik, ki vsebuje gumbe s slikami tekstur izbranih materialov. Program preverja vrednost spremenljivke definirane v prejšnjem delu. V primeru vrednosti *"True"* se bo drsnik prikazal, drugače pa ne bo viden. Potrebno je še nastaviti začetno vidljivost drsnika na skrito, to se pa spreminja v podrobnostih drsnika.





Slika 30: Prikaz kode, ki spreminja vidljivost podanega drsnika z materiali

Dogodek se izvede s klikom na gumb, ki nato poišče vse elemente z izbrano oznako. Zanka se nato sklicuje na vsak izbran element in jim pripiše izbrani material.



Slika 31: Prikaz kode, ki spremeni material elementu z določeno oznako

## 5 ZAKLJUČEK

V diplomski nalogi je bilo ugotovljeno, da je za namen uporabe BIM za aplikacije virtualne resničnosti potrebno programsko orodje, ki podpira tehnologijo vizualizacije v realnem času. Zaradi omogočenega programiranja in možnosti izvoza aplikacije, je bilo v namen raziskave izbrano programsko okolje Unreal Engine. Virtualna resničnost v aplikaciji predstavlja zgolj dodatno funkcijo. Glavna značilnost aplikacije je interaktivno raziskovanje okolja v realnem času, katera ne zahteva nameščanja dodatnega programskega orodja in ki zaradi napredovanja tehnologija postaja vedno bolj dostopna do uporabnika.

V programu so bile raziskane nekatere možne aplikacije za namen projektiranja in gradnje. Ugotovljeno je bilo, da lahko projektant preko izdelane aplikacije, vključi naročnika oziroma uporabnika v sam proces projektiranja. Izvožena aplikacija se deli z naročnikom, kateremu omogoča samostojno raziskovanje objekta, interakcijo z okoljem, pregled funkcionalnosti objekta, in prilagajanje videza končnega rezultata. Uporabnik se tako seznani z pričakovanim končnim rezultatom in glede na ugotovitve zahteva popravke. Preko sodelovanja lahko nato projektant iterativno prilagaja in popravlja objekt. Projektant lahko izdelava aplikacijo za namen uporabe na gradbišču, preko katere omogoča podrobni pregled detajlov, inštalacij, in ostalih konstrukcijskih elementov. Uporabnik lahko interaktivno vklaplja in izklaplja vidljivost konstrukcijskih slojev in pri tem opazuje fazni potek gradnje ali pa si ogleda armaturne načrte v 3D prostoru. Z razvojem novih orodij in tehnologij, uporaba virtualnega interaktivnega okolja predstavlja velik potencial v prihodnosti zasnove in gradnje objekta.

Poleg tega pa je diplomska naloga predstavljala uvod v spoznavanje novih programskih orodij in procesov, s katerimi se nameravam še podrobneje spoznavati v nadaljnji študijski in službeni poti.

## VIRI

- [1] Lasse, R. 2014. Striving For Real-Time Photorealism.  
<https://www.unrealengine.com/en-US/blog/striving-for-real-time-photorealism> (Pridobljeno 25.8.2017.)
- [2] Virtual Reality Society. 2017. What is Virtual Reality?  
<https://www.vrs.org.uk/virtual-reality/what-is-virtual-reality.html> (Pridobljeno 22.8.2017.)
- [3] Augment News. 2015. Virtual Reality vs. Augmented Reality.  
<http://www.augment.com/blog/virtual-reality-vs-augmented-reality/> (Pridobljeno 22.8.2017.)
- [4] Unreal Engine. 2017. Game Engine Technology by Unreal.  
<https://www.unrealengine.com/en-US/what-is-unreal-engine-4> (Pridobljeno 15.5.2017.)
- [5] Twinmotion. 2017. Home 2018 - Twinmotion.  
<https://twinmotion.abvent.com/en/> (Pridobljeno 12.8.2017.)
- [6] Lumion3d. 2017. Specifications | Lumion.  
<https://lumion3d.com/specifications.html> (Pridobljeno 15.5.2017.)
- [7] Unity. 2017. Unity - Products.  
<https://unity3d.com/unity>. (Pridobljeno 15.5.2017.)
- [8] Blender. 2017. Free And Open 3D Creation Software.  
<https://www.blender.org/> (Pridobljeno 25.8.2017.)
- [9] Autodesk. 2017. 3Ds Max | 3D Modeling, Animation & Rendering Software | Autodesk.  
<https://www.autodesk.com/products/3ds-max/overview>. (Pridobljeno 25.8.2017.)
- [10] Docs.Unrealengine. 2017. Lightmass Basics.  
<https://docs.unrealengine.com/latest/INT/Engine/Rendering/LightingAndShadows/Lightmass/Basics/>. (Pridobljeno 28.8.2017.)
- [11] Gnomon. 2017. Unreal Engine By Epic Games: Lighting And Rendering. Video.  
<https://youtu.be/qu3NfoXNZG4> (Pridobljeno 15.8.2017.)
- [12] Vrtisan Architectural Visualisation. 2016. Interactive Virtual Reality Architectural Visualisation - Vrtisan - Examples Of Interactivity. Video.  
[https://youtu.be/ey9F0k\\_Nmf0](https://youtu.be/ey9F0k_Nmf0) (Pridobljeno 28.8.2017.)
- [13] Docs.Unrealengine. 2017. Linetracebychannel.  
<https://docs.unrealengine.com/latest/INT/BlueprintAPI/Collision/LineTraceByChannel/> (Pridobljeno 29.8.2017.)
- [14] Docs.Unrealengine. 2017. Blueprint Variables.  
<https://docs.unrealengine.com/latest/INT/Engine/Blueprints/UserGuide/Variables/> (Pridobljeno 29.8.2017.)
- [15] Bimobject. 2017. Bimobject VR.  
<https://vr.bimobject.com/> (Pridobljeno 31.8.2017.)

[16] Docs.Unrealengine. 2017. Unreal Engine VR Editor.

<https://docs.unrealengine.com/latest/INT/Engine/Editor/VR/> (Pridobljeno 10.9.2017.)

[17] VRtisan Architectural Visualisation. 2016. Architectural Design in Virtual Reality - Full Process - VRtisan - Unreal Engine VR Editor .

[https://youtu.be/X7h\\_JOPHoxk](https://youtu.be/X7h_JOPHoxk) (Pridobljeno 29.8.2017.)

[18] Wiki.unrealengine. 2017. Recommended Hardware.

[https://wiki.unrealengine.com/Recommended\\_Hardware](https://wiki.unrealengine.com/Recommended_Hardware) (Pridobljeno 13.9.2017)

[19] Vive. 2017. Vive Ready Computers.

<https://www.vive.com/us/ready/> (Pridobljeno 13.9.2017)