Interaktivní 3D model bytového domu

Bakalářská práce

Vedoucí práce: Ing. Dita Dlabolová

Lukáš Boček

Brno 2017

Rád bych poděkoval Ing. Ditě Dlabolové za vedení této bakalářské práce, za pomoc při výběru tématu a za trpělivost, kterou se mnou měla. Dále bych chtěl poděkovat rodině a kamarádům za podporu během studia a při psaní bakalářské práce.

Čestné prohlášení

Prohlašuji, že jsem tuto práci: **Interaktivní 3D model bytového domu** vypracoval samostatně a veškeré použité prameny a informace jsou uvedeny v seznamu použité literatury. Souhlasím, aby moje práce byla zveřejněna v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách ve znění pozdějších předpisů, a v souladu s platnou *Směrnicí o zveřejňování vysokoškolských závěrečných prací*. Jsem si vědom, že se na moji práci vztahuje zákon č. 121/2000 Sb., autorský zákon, a že Mendelova univerzita v Brně má právo na uzavření licenční smlouvy a užití této práce jako školního díla podle § 60 odst. 1 Autorského zákona. Dále se zavazuji, že před sepsáním licenční smlouvy o využití díla jinou osobou (subjektem) si vyžádám písemné stanovisko univerzity o tom, že předmětná licenční

(subjektem) si vyžádám písemné stanovisko univerzity o tom, že předmětná licenční smlouva není v rozporu s oprávněnými zájmy univerzity, a zavazuji se uhradit případný příspěvek na úhradu nákladů spojených se vznikem díla, a to až do jejich skutečné výše.

Brno, 22.05.2017

.....

Abstract

Boček, L. Interactive passage apartment building using tools Blender and Python. Bachelor thesis. Brno, 2017

Thesis deals with interactive passage of the apartment building in town Kunštát na Moravě, grandual modeling using Blender and enriched by scripts in language Python. Describes the different Blender tools, modeling and creating interactivity and executable programs.

Key words:

Blender, Python, 3D model, interactive passage, Game Engine, script

Abstrakt

Boček L. Interaktivní průchod bytovým domem s vyžitím nástrojů Blenderu a Python. Bakalářská práce. Brno 2017

Práce se zabývá interaktivním průchodem bytového domu ve městě Kunštát na Moravě, postupným vymodelováním pomocí programu Blender a obohaceném o skripty v jazyce Python. Popisuje různé nástroje Blenderu, modelování a tvorbu interaktivity a spustitelných programů.

Klíčová slova:

Blender, Python, 3D model, interaktivní průchod, Game Engine, skript

Obsah

1	Úvod	9
2	Cíl práce	10
3	Teoretiký základ 3.1 3D počítačová grafika 3.2 Modelování 3.3 Interaktivita	11 11 11 11
4	Nástroje pro tvorbu modelů4.1Cinema 4D4.2Rhinoceros 3D4.33ds Max4.4SketchUp4.5LightWave 3D4.6Blender4.6.1Historie4.6.2Open source4.6.3Výhody4.6.4Nevýhody4.6.5Uživatelské rozhraní4.6.6Myš a klávesové zkratky4.7Editory Blenderu4.7.1Info Editor4.7.23D View4.7.3Graph Editor4.7.4Dope Sheet4.7.5Timeline Editor4.7.6UV/Image Editor4.7.8Node Editor4.7.9Logic Editor4.7.10Properties Editor4.7.11Python console4.7.12Další editory4.7.13Game Engine4.8Porovnání programů	$\begin{array}{c} 12 \\ 12 \\ 12 \\ 12 \\ 13 \\ 13 \\ 13 \\ 14 \\ 14 \\ 14 \\ 15 \\ 15 \\ 16 \\ 17 \\ 17 \\ 17 \\ 17 \\ 17 \\ 17 \\ 18 \\ 18 \\ 18 \\ 18 \\ 18 \\ 19 \\ 20 \\ 20 \\ 21 \\ 21 \\ 21 \\ 22 \end{array}$
-	ъ <i>к. (</i> . 1•1	

6	Vlas	Vlastní práce 25											
	6.1	Analýza návrhů rekonstrukce											
	6.2	2 Důležité nástroje											
	6.3	Modelování	26										
		6.3.1 Venkovní scéna	27										
		6.3.2 Scéna chodba	30										
		6.3.3 Scéna byt	31										
		6.3.4 Materiály	32										
		$6.3.5 Osvětlení \dots \dots \dots \dots \dots \dots \dots \dots \dots $	33										
	6.4	Logika aplikace	33										
		6.4.1 Možnosti interaktivního průchodu	33										
		6.4.2 Animace dveří \ldots	35										
		6.4.3 Skripty v Pythonu	35										
		$6.4.4 \text{Export} \dots \dots$	40										
7	Dial		11										
1	71	Proktická využití	41										
	7.1 7.2	Co se povedlo	41										
	7.2	Co se povedle	41										
	7.0	Problémy při vytváření aplikace	41 //1										
	7.4		41 //2										
	1.0		44										
8	Záv	ěr	43										
	8.1	Použité nástroje	43										
9	Refe	erence	44										
Př	Přílohy												
\mathbf{A}	A Ukázky modelů												
в	Příl	ohy na CD	50										

Seznam obrázků

1	Jednoduchý materiál v Node editoru	19
2	Venkovní scéna bytového domu	27
3	nastavení nového stromu	29
4	Nastavení terénu	30
5	Schody scény chodba	31
6	Drátěný model křesla	32
7	Chůze implementována v Logic Editoru	34
8	Chůže implementována pomocí Python scriptu	34
9	Animace dveří v Dope Sheet editoru	35
10	Úvodní obrazovka aplikace	48
11	Sedací nábytek v obývacím pokoji	48
12	Ukázka z aplikace 1	49
13	Ukázka z aplikace 2	49

Seznam tabulek

1 Po	rovnání programi	ì						•			•	•		•	•											•	•	•	22
------	------------------	---	--	--	--	--	--	---	--	--	---	---	--	---	---	--	--	--	--	--	--	--	--	--	--	---	---	---	----

1 Úvod

Bytový dům, který bude modelován se nachází ve městě Kunštát na Moravě. Kunštát má v dnešní době kolem 2 500 obyvatel. Město se nachází od Brna přibližně 40 km a náleží do okresu Blansko.

Modelovaný bytový dům má rozměry 60 metrů na délku, 10 metrů na šířku a 15 metrů na výšku. Obsahuje 4 patra, kde se nacházejí byty a sklepy s garážemi. Bytový dům má dva vchody po 11 bytech, ve 2. až 4. patře jsou 3 byty, 2 na stranách a 1 uprostřed. V 1. patře jsou jen 2 byty na stranách, prostřední byt chybí, jelikož na jeho místě jsou schody ven.

3D Počítačová grafika je zobrazována v trojrozměrném prostoru. Na rozdíl od 2D počítačové grafiky přibyla k šířce a výšce navíc hloubka. Na objekty v 3D grafice je možné nahlížet ze shora, zprava, zepředu, stejně jak je tomu v reálném světě. 3D grafika má velmi širokou škálu uplatnění a pořád se vyvíjí. A spolu s 3D grafikou se vyvíjí i programy pro práci ve 3D. Některé jsou placené a některé zdarma, takže si vybere každý podle svých možností. Nejznámější placené programy jsou: Rhinoceros 3D, Cinema 4D a 3ds Max. Na druhé straně nejznámější open-source program je Blender, který slouží pro vytváření interaktivních 3D aplikací.

Samotná bakalářská práce se bude skládat z části teoretické a praktické. V první části bude objasněna problematika týkající se řízení 3D nástrojů. V praktické části bude provedeno samotné modelování 3D objektu a bude programován průchod bytovým domem.

2 Cíl práce

Při rekonstrukci panelových domů je největší problém komunikace mezi lidmi, kteří v těchto domech bydlí. Většina lidí nemá šanci dát najevo svůj názor a rekonstrukce se obejde bez nich. Nejvhodnějším nástrojem, aby se tak nedělo, je model 3D domu.

Cílem bakalářské práce je vymodelovat 3D model bytového domu. Pro modelování bude využíván open-source software Blender. Následně bude do modelu přidána interakce pro uživatele jako je průchod okolo bytového domu a chodby. Interakce bude vytvořena pomocí skriptů, které budou implementované v programovacím jazyce Python. Tento 3D model bude sloužit pro zástupce zvoleného bytového domu jako inspirace pro rekonstrukci. Náhled pro uživatele bude ve výchozí aplikaci, ve které bude možný průchod scénami.

Práce bude sloužit nejen pro zástupce bytového domu, ale i pro jeho obyvatele. Obyvatelé si budou moci dům prohlédnout zvenku, jeho chodbu a jeden zvolený byt. Jeden byt proto, jelikož jsou všechny byty na stejném principu. Bylo by tedy zbytečné byty kopírovat a vkládat, jelikož by to zatěžovalo výslednou aplikaci. Práce by mohla být do budoucna využitelná pro podobné projekty, jako jsou například realitní kanceláře nebo developerské domy. Zadání práce:

- Seznámení se s open source programem Blender a jeho Game Enginem
- Provedení sběru a analýzu požadavků na 3D model
- Vytvoření 3D modelu domu, který bude doplněný o interaktivní prvky
- Otestování výsledného modelu a zhodnocení výsledků práce a praktický přínos práce

3 Teoretiký základ

3.1 3D počítačová grafika

Počítačová 3D grafika pracuje s trojrozměrnými objekty v trojrozměrném souřadnicovém systému (osy X, Y a Z). Pracuje se souřadnicemi bodů a informacemi o křivkách, plochách a úsečkách. Každý model se skládá z bodů (vertexů), hran a ploch. Spojené 2 body tvoří hranu a více spojených hran tvoří plochu (Matoušek, 2015). Z trojrozměrných objektů dostaneme dvojrozměrný obrázek pomocí renderování. Renderování je tvorba reálného obrazu na základě počítačového modelu, nejčastěji 3D. Renderování obsahuje mnoho parametrů a nastavení, které ovlivňují konečný vzhled scény.

3D grafika se nejčastěji využívá při vytváření filmů (animace), ve vědě a průmyslu (počítačová simulace a zobrazení orgánů). Dnes se 3D grafika také využívá při 3D tisku, kdy si uživatel vymodeluje model například letadla a poté si ho vytiskne po částech na 3D tiskárně. (Počítačová 3D grafika, 2017)

3.2 Modelování

Je to proces tvarování a vytváření trojrozměrného modelu. Trojrozměrný model může být vytvořen pomocí modelovacího nástroje, na základě dat, které získáme měřícím přístrojem z reálného světa, anebo také na základě počítačové simulace. Při modelování trojrozměrné scény jsou důležité 4 základní kroky: modelování, osvětlení, texturování, renderování. K těmto 4 základním krokům patří další, například animace, interaktivita. (Počítačová 3D grafika, 2017)

3.3 Interaktivita

Je to aktivita uživatele se stroji nebo technologickým zařízením. Interaktivní zařízení reagují na aktivitu uživatele. Mezi aktivitu uživatele můžeme zařadit například pohyb, dotek tlačítka nebo slovní příkazy. Interaktivita se objevuje i v hudbě, ve filmech a pořadech. (Interaktivita, 2017)

4 Nástroje pro tvorbu modelů

4.1 Cinema 4D

Je to komerční multiplatformní software, který je dostupný jen pod placenou licencí. Cinemu 4D vytvořila německá společnost Maxon Computer. Tento program obsahuje několik verzí, podle toho se odvíjí cena. Nejlevnější variantou je verze s názvem Prime. Nejvyšší verze programu se nazývá Studio, která obsahuje všechny prvky menších verzí a obsahuje navíc speciální nástroje pro tvorbu dynamických animací. Poslední 3 verze se nazývají Broadcast, Visualize a zkušební verze (jedná se o verzi, která je dostupná zdarma po dobu jednoho měsíce). Ke každé verzi se mohou dokoupit takzvané moduly. Tuto možnost využívá spousta lidí, jelikož si koupí to, co potřebují a zbytečně neplatí za věci, které nevyužijí. Například modul MOCCA (tvorba 3D postav a jejich animace), modul BodyPaint 3D (pokročilé texturování a malování přímo na 3D objekty).

Jelikož jde o placený program, vývojáři zaručují kvalitní podporu. Mezi hlavní výhody patří podpora českého jazyka přímo od vývojářů. Dále existuje na internetu spousta návodů, video návodů a dokumentací. Cinema 4D je jednoduchá a intuitivní na ovládání. Proto je často doporučována pro začátečníky. Sám jsem si poprvé ke 3D grafice "přičuchnul" díky tomuto programu. (Cinema 4D, 2017)

4.2 Rhinoceros 3D

Je to jeden z nejvíce používaných komerčních 3D programů na platformě Windows. Byl vyvinut společností Robert McNeel Associates v roce 1992, nejdříve sloužil jako nadstavba AutoCadu. Cena za Rhinoceros 3D je kolem 25 000 Kč, ale je závislá na počtu doplňků a licenci. Vývojáři tohoto programu, ale poskytují slevy pro studenty a učitele, potom se cena dostane na výši kolem 5 000 Kč.

Prostředí je uživatelsky přívětivé a přehledné. Program je rozdělen na 4 části a každá část představuje různý pohled na objekty (3D pohled neboli perspektiva, shora, zepředu a zprava). Každý pohled lze zvětšit a zmenšit, podle toho, jak uživatel potřebuje. Nemá takové možnosti animací jako ostatní 3D modelovací programy, ale vynahrazuje to již výše zmíněným uživatelsky přívětivým prostředím. (Rhinoceros 3D, 2017)

4.3 3ds Max

Je světově nejrozšířenější program pro vytváření animací postav, filmové grafiky, vizuálních efektů a v neposlední řadě počítačových her. Tento program vyvíjí společnost Autodesk. Program 3ds Max je dostupný pouze pod placenou licencí, a tak je využíván spíše v profesionálních studiích. Tento program byl využit při vytváření několika úspěšných filmů, například: Avatar, Harry Potter, Star Wars, Iron Man, Spider-Man, a dalších. Hardwarové požadavky na tento program jsou náročnější. 3ds Max je kompatibilní se všemi aplikacemi společnosti Autodesk a má také vestavěný skriptovací jazyk MaxScript (Kříž, 2010). Menší projekty lze spustit na notebooku, anebo na stolním počítači. Hlavní výhoda je stálá podpora vývojářů a odstraňování chyb nových verzí programu. Naopak menší nevýhodou je absence českého jazyka, což může odradit začínající uživatele. (3ds Max, 2017)

4.4 SketchUp

Jedná se o nástroj od společnosti Trimble, ale původním vývojářem programu byla společnost Last Software, kterou v roce 2006 koupil Google. SketchUp byl vyvíjen jako plug-in do aplikace Google Earth. Díky tomuto propojení lze vymodelovat dům, továrnu, značky a přidat je do 3D vrstvy Google Earth. V roce 2012 koupila SkethUp od Googlu společnost Trimble. Existují 2 verze, jedna je zdarma a nese název Make, naopak druhá verze je placená a také profesionálnější a nazývá se Pro. Tento program nabízí vytváření animací, ale jeho hlavní účel je jiný. Slouží pro prezentaci inžennýrských projektů. Dále pro konstruktéry, tvorbu modelů a 3D tiskárny. Hlavní nevýhoda je horší kvalita objektů, ale lze to vyřešit díky pluginům. (SketchUp, 2017)

4.5 LightWave 3D

Je to legenda mezi 3D modelovacími programy od společnosti NewTek. Umí kombinovat špičkový render s intuitivními nástroji pro animaci a 3D modelování. Je celosvětově oblíbený jako kompletní balík 3D nástrojů. Používá se pro tvorbu tištěné grafiky, grafiky pro web, vizuálních efektů pro film i televizní produkce, vizualizaci a pro vývoj her. Uživatelé tohoto programu posbírali víc cen Emmy, než uživatelé jiných programů. A to hlavně pomocí nadstandardní výbavě programu. Program LightWave 3D obsahuje 999 renderovacích klientů a plnou 64 bitovou kompatibilitu se systémy Windows a Max OS X. Výrobce Lightwave poskytuje bezplatnou technickou podporu. Tento program byl využit při vytváření několika filmů, například: Jurský park, X-Man, Piráti z Karibiku, Titanik, a dalších. (LightWave, 2015)

4.6 Blender

Je to open-source 3D modelovací program. Slouží pro modelování a vykreslování 3D počítačové grafiky. Pro vykreslování používá knihovnu OpenGL. I díky tomu je dostupný pro operační systémy Microsoft Windows, Max OS X a GNU/Linux (Pokorný, 2008). V nejnovějších verzích se již objevuje český jazyk, ale neobsahuje celý překlad, stále se objevuje v programu mnoho anglických slov. I přesto, že je zdarma dokáže se téměř vyrovnat placeným programům, které jsou zmíněny výše. Je uživatelsky dobře nastavitelný, každý si prostředí přizpůsobí svým potřebám a zvykům. Lze přidávat neomezený počet oken, které se dají zvětšovat, zmenšovat, posouvat a také mazat. Blender podporuje 3D modelování, renderování, tvorbu animací a dokonce i editaci videí a tvorbu her. Blender má vestavěný game engine, díky kterému podporuje tvorbu interaktivních aplikací, například: tvorbu her, animace nárazu a spoustu dalších věcí. Blender lze také rozšířit o skripty napsané v programovacím jazyce Python. Skripty si může zkušenější uživatel napsat sám, anebo existuje mnoho internetových fór, kde se dají sdílet s ostatními uživateli. Některé jsou již součástí instalace, stačí jen povolit v nastavení. (About – blender, 2016)

4.6.1 Historie

V roce 1988 Ton Roosendaal založil v Nizozemsku animační studio NeoGeo. Tento rok se považuje jako počátek Blenderu. Za nedlouho se toto studio stalo jedním z popředních animačních studií v Evropě. V roce 1995 se Roosendaalovi začalo zdát, že jsou nástroje v NeoGeo zastaralé, a proto začal pracovat na nové sadě 3D nástrojů pro animaci, dnes známé jako Blender. V roce 1998 se Roosendaal rozhodl, že založí další společnost, ta nesla název NaN (Not a Number), která se zabývala rozvojem Blenderu a na trhu nabízela služby s ním související. Ale společnost NaN se i navzdory prvotním úspěchům nepodařilo udržet. Proto v roce 2002 Ton Roosendaal založil společnost Blender Foundation. Hlavním úkolem této společnosti se stalo rozvíjení Blenderu jako open source. Díky sbírce a kampani skupiny dobrovolníků se podařilo od investorů NaN odkoupit původní zdrojový kód Blenderu. Od této chvíle je Blender šířen jako open source. V roce 2008 vyšel ze studia Blenderu první krátký animovaný film s názvem Big Buck Bunny. (Blender dokumentace – Historie, 2017)

4.6.2 Open source

Je to počítačový software, který má tzv. otevřený zdrojový kód. To znamená technickou dostupnost kódu, tak legální licenci. Blender je vydávaný pod licencí GNU (Genral Public Licence). Je to program, který je volně dostupný pro jakýkoliv účel, například kopírování nebo dokonce upravování jeho zdrojového kódu. Je to absolutní opak licencování, kde je možnost nahrání programu jen na jeden počítač, pokud není určena licence na více počítačů, neumožňuje kopírování a ani uživatel neuvidí zdrojový kód. Mezi nejznámější open source software patří GNU/Linux, Android, MySQL, Blender, GIMP a mnohé další. (Ewox, 2017)

4.6.3 Výhody

Největší a nespornou výhodou je to, že Blender je ke stažení zcela zdarma, a dokonce i pro komerční účely. Blender je multi-platformní program. Je používán na několika platformách, například Max OS X, GNU/Linux a v neposlední řadě pro Microsoft Windows. Také po hardwarové stránce je na tom Blender dobře, v porovnání s ostatními 3D programy má velmi nízké hardwarové nároky. Každý rok vychází alespoň jedna nová verze, což je další obrovská výhoda. Na jednom počítači, je možnost v pořádku používat současně a nezávisle na sobě několik verzí Blenderu. Je možné tedy současně pracovat v jednom okně a ve druhém renderovat jinou práci. Jedná se o velmi užitečnou funkci, když starší projekt nejde spustit v novější verzi, tak projekt jednoduše spustíme ve verzi starší a pracujeme vesele dále a není nutné program reinstalovat na předešlou verzi, nebo používat jiný počítač. Všechny verze jsou ke stažení na oficiálních stránkách blender.org.

Velkým bonusem je použití Python skriptů. I přes to, že je Blender zdarma, tak dokáže konkurovat placeným programům jako jsou Cinema 4D a 3ds Max.

4.6.4 Nevýhody

Jako každý program i Blender má své nevýhody, ale není jich mnoho. Na zásadní chybu jsem při používání Blenderu nenarazil. Pouze jedna chyba stojí za zmínku. Při pokusu o uložení program, hlásil chybu "Nelze uložit". Nevím, jestli to byla přímo chyba programu anebo něčeho jiného. Každopádně jsem přišel asi o pěti hodinovou práci.

Mezi obecné nevýhody patří přehozená funkcionalita tlačítek myši. U většiny programů jsou objekty označovány levým tlačítkem myši, u Blendru je to přesně naopak. Objekty jsou označovány pravým tlačítkem myši. Při otevření projektů ze starší verze ve verzi nové se v ne tak častých případech stávalo, že byl projekt rozházený. Ale jak je zmíněno ve výhodách, je možnost projekt otevřít ve starší verzi.

V nové verzi je to již opraveno, ale ve starších verzích Blender neupozorňoval při zavírání programu na neuložený projekt. Stávalo se, že při nechtěném kliknutí na tlačítko "zavřít" se projekt jednoduše zavřel a uživatel tak přišel o práci.

4.6.5 Uživatelské rozhraní

V Blenderu se pracuje s editory. Každý editor je využíván pro jinou práci s daty. Například editor 3D view slouží pro zobrazení 3D modelu, editor Outliner slouží pro přehled všech stavebních bloků v scéně anebo Properties editor, který slouží pro úpravu vlastností objektu. Obrazovka v Blenderu je rozdělena na oblasti, každá oblast představuje jeden editor. Obrazovku si může každý uživatel přizpůsobit podle svých představ a potřeb. Pro účinnější a rychlejší práci v Blenderu slouží layouty, kterých vývojáři připravili 9. Jsou vhodné pro různé typy úloh a to sice skriptování, animování a modelování. (Blender dokumentace – obrazovky/uživatelské rozhraní, 2017)

- 3D View Full -3D pohled používaný pro zobrazení náhledu na scénu
- Animation tvorba animací objektů, změny barvy, a další
- Compositing slouží pro slučování a kombinaci různých částí scén

- Default základní layout, základní rozdělení obrazovky
- Game Logic slouží pro programování her
- Motion Tracking používá se pro sledování pohybu objektů
- Scripting slouží především pro psaní skriptů v jazyce Python
- UV Editing slouží pro mapování 2D obrázku na 3D objekty
- Videon Editing stříhání a editace animací

4.6.6 Myš a klávesové zkratky

Vývojáři Blenderu zvolili pro označování objektů pravé tlačítko myši, čímž se program Blender liší od ostatních 3D modelovacích programů, kde se používá levé tlačítko myši. Levé tlačítko v Blenderu slouží ke změně 3D kurzoru. Hlavní účel 3D kurzoru je výchozí pozice pro vkládání nových objektů. Tento kurzor dále slouží pro transformace (například zvětšování a zmenšování objektů).

Velká výhoda je používání prostředního tlačítka myši neboli "kolečka". Při scrollování se klasicky přibližuje a oddaluje náhled. Ale jestliže se kolečko stiskne, tak se při pohybu myši mění úhel náhledu. Další použití "kolečka" je s použitím klávesy Shift,posouvá náhled do stran. V neposlední řadě při použití tlačítka Shift a scrollování ,kolečka" se náhled posouvá nahoru a dolů.

Podstatnou část práce v Blenderu tvoří klávesové zkratky. Klávesové zkratky nejsou nezbytné, ale při používání vám práci nejen usnadní, ale i značně zrychlí. Tato nevýhoda se po osvojení promění ve velikou výhodu. Možnost kliknout na klávesu a táhnout myši velmi ušetří čas oproti zdlouhavému hledání tlačítka na téže funkce. Jestliže chcete použít tyto zkratky musí být kurzor myši v 3D okně, jinak nebudou fungovat.

Přehled nejčastěji používaných klávesových zkratek a klávesových zkratek s použitím myší a jejich funkce:

- Num 0 zobrazení 3D náhledu z pohledu kamery
- Num 1 zobrazení 3D náhledu zepředu
- Num 2 pootočení 3D náhledu ve směru dolů
- Num 3 zobrazení 3D náhledu z boku
- Num 4 pootočení 3D náhledu ve směru doleva
- Num 5 přepínání mezi perspektivním a ortografickým zobrazením
- Num 6 pootočení 3D náhledu ve směru doprava
- Num 7 zobrazení 3D náhledu ze shora
- Num 8 pootočení 3D náhledu ve směru nahoru

- Num 9 otočení 3 D náhledu o 180°
- Tab přepínání mezi objektovým módem a editačním módem
- R rotace objektu
- S změna velikosti objektu
- G posun objektu
- A označení objektu (v objektovém módu), označení hran, vertexů, plošek (v editačním módu)
- Del/X vymazání objektů
- Shift +D duplikace objektu (v objektovém módu), duplikace hran, vertexů, plošek (v editačním módu)
- Ctrl + Šipka nahoru zvětší se aktuální okno na celou obrazovku
- Ctrl + Šipka dolů celé okno se zmenší na přechozí velikost
- T zobrazení a skrytí panelu s nástroji
- N zobrazení panelu s nastavením

4.7 Editory Blenderu

4.7.1 Info Editor

Při spuštění Blenderu je Info editor k nalezení na horní straně okna. Uživatel vidí jen jeho záhlaví, kde je k nalezení ukládání, otevření a vytvoření nového souboru, možnosti pro renderování, aktuální informace o Blenderu (aktuální verze, kolik soubor obsahuje objektů, světel a další) a několik výběrových polí, pomocí kterých nastavíme prostředí Blenderu, jak potřebujeme. Nalezneme zde Screen layout. Seznam možných layoutů je zmíněn v kapitole Uživatelské rozhraní. Dále je zde možnost vybrat scény, které jsou v jednom souboru. V neposlední řadě je zde Rendering engine, kde je možné přepínat mezi jednotlivými renderovacími enginy (Blender Render, Blender Game a Cycles Render). (Blender dokumentace – Info Editor, 2017)

4.7.2 3D View

Bez tohoto editoru by se moc práce neudělalo. Je to jeden z nejdůležitějších editorů, které v Blenderu jsou. 3D View editor zobrazuje změny, které se v modelu provádí a slouží k přepínání módů (editační, objektový, a další). Dále se využívá pro modelování a vykreslování textur. Editor má po stranách ovládací prvky, které se dají skrýt a zobrazit pomocí klávesových zkratek N a T. Mezi dva nejdůležitější módy pro práci v tomto editoru jsou Editační a Objektový mód, které se přepínají pomocí klávesové zkratky Tab. V objektovém módu se pracuje s celými objekty a provádí se zde práce typu rotace, vytváření, zvětšování, přiřazování textur a materiálů celým objektům a mnohé další úkony. V editačním módu se pracuje s jednotlivými vertexy, hranami a plochami. Podle toho, co je potřeba. I v editačním módu lze upravovat velikost jednotlivých částí a přiřazovat textury a materiály jednotlivým plochám. Dále se v tomto editoru mění zobrazení objektů (drátěné, texturované, ohraničující). (Blender dokumentace – 3D View, 2017)

4.7.3 Graph Editor

Graph Editor je hlavní nástroj, který slouží pro správu animací. V tomto editoru jsou dostupné dva módy, prvním z nich je F-Curves a druhým je Drivers. Mód F-Curves umožňuje editaci kanálů za pomocí funkčních křivek (konstantní, beziérová a lineární). Mód Drivers spravuje animace za pomocí změny velikosti jednotlivých atributů. V editoru je zobrazena časová osa X, ta slouží pro zobrazení jednotlivých snímků a hodnota proměnných na ose Y. Graph editor obsahuje možnost přidávání marketů. Markety jsou značky, které slouží pro označení a popis oblastí animace, což zvyšuje přehlednost. (Blender dokumentace – Graph Editor, 2017)

4.7.4 Dope Sheet

Tento editor je určen především pro správu více animací naráz. Slouží pro lepší orientaci a dohled na všechny animace na jednom místě. Jsou zde k nalezení informace o animaci (název, údaje o atributech objektů a mnohé další informace), umožňuje vkládat značky označující důležitá místa v animaci a je možné v něm změnit více snímků současně. (Blender dokumentace – Dope Sheet, 2017)

4.7.5 Timeline Editor

Timeline editor je užitečný, když se provádí kontrola jednoduchých animací. Slouží jako takový jejich přehled. Editor je používán pro vkládání klíčových snímků na časovou osu. V editoru je možné sledovat a měnit aktuální čas snímku a nastavení počátečního a koncového snímku animace. V editoru jsou tlačítka pro přehrání, pozastavení a přeskočení určité části animace. (Blender dokumentace – Timeline Editor, 2017)

4.7.6 UV/Image Editor

UV/Image Editor slouží k několika účelům, tím hlavním je spravování obrázků. Používá se především pro UV mapování a texturování složitějších modelů, jako je například obličej. V průběhu texturování je možné využít mapování, to znamená nanášení textur za pomocí předpřipravených funkcí. UV mapování je v Blenderu jedním z nejlepších způsobů nanášení textur, poskytuje kontrolu vložení 2D textury na 3D objekt. Při této technice je důležité správné rozložení vymodelovaného objektu na jednotlivé plošky anebo větší části, které budou texturovány, tento krok

se nazývá unwrapping (rozbalení modelu). Rozložení objektu probíhá podle funkce, která označí jednotlivé hrany jako důležité a nazývá se Mark Seam. Následně se automaticky rozbalený a označený objekt zobrazí v UV/Image editoru. Poslední částí je texturování, které následuje. Tento editor umožňuje rozbalený objekt exportovat jako obrázek, který může být uložen v různých formátech. Usnadňuje to vytváření a editaci textury a její vložení zpět na správné místo. (Blender dokumentace – UV/Image Editor, 2017)

4.7.7 Textový Editor

V textovém editoru se vytvářejí textové dokumenty jako jsou například "readme" texty, které popisují obsah souboru. Primárně se využívají na psaní vlastních skriptů. Pro ulehčení práce jsou používány předpřipravené šablony (templates). Šablony uživateli usnadní značně práci, jelikož nemusí psát začátek skriptku, který je ve většině případů stejný. Navíc taky poskytuje zvýraznění syntaxe a zobrazení čísel řádků, což napomáhá při debugování kódu. (Blender dokumentace – Text editor, 2017)

4.7.8 Node Editor

Node editor slouží k vytváření materiálů. Materiály jsou vytvářeny pomocí uzlů (nodes), ty se za sebe mohou řetězit. Každý uzel vykonává určitou operaci a tím ovlivňuje vzhled výsledného materiálu. V node editoru lze vytvořit komplexní materiály. Typů uzlů je spousta, jako například Vstup: textura, materiál, hodnota anebo RGB barva (popsána pomocí červené, zelené a modré), další možností je barva, která slouží pro úpravu odstínu, sytosti barvy, gamma korekce, správu barvy a mixování více barev, dále vektor, který je používán k mapování textur, různé druhy konvertorů a v neposlední řadě výstup materiálu. (Blender dokumentace – Node Editor, 2017)



Obrázek 1: Jednoduchý materiál v Node editoru

4.7.9 Logic Editor

Logic Editor slouží pro nastavování logiky her a interaktivních aplikací v Blenderu. Nastavují se v něm vlastnosti objektů a jejich chování. Nastavení logiky se provádí pomocí třech částí (logic bricks) a jsou propojeny pomocí linek, ty jim umožňují propojení a komunikaci mezi sebou. První částí logického editoru je senzor, Controller a Aktuator. Senzory čekají na nějakou akci, typický senzor je standartní vstup z klávesnice, joystick, myš, dotek, vlastnost, a další. Druhou částí je Regulátor (Controller), do nějž jdou data ze senzorů a vyhodnocuje je pomocí logických operací, kterými jsou například and, nand, or nor, xor a další. Po vyhodnocení regulátor pošle signál do třetí části, jestliže je výsledek kladný pošle signál s kladnou hodnotou, a když je výsledek záporný odešle signál se zápornou hodnotou. Třetí část se nazývá akční člen (actuator). Ten na základě získaných impulzů vykonává akci. Mezi tyto akce patří například změna scény, spuštění animace, pohyb objektu, nastavení kamery a mnohé další. (Blender dokumentace – Logic Editor, 2017)

4.7.10 Properties Editor

Properties editor je složen z několika záložek, které slouží k nastavení vlastností projektu. Jednotlivé záložky jsou:

- Render provádí se zde nastavení požadavků pro výsledek renderování. Umožňuje nastavení výsledného formátu obrázku nebo animace a nastavení výsledného rozlišení
- Scene v této záložce se upravují vlastnosti scény, jako je například úprava barev, zvuku a jednotky, ve kterých se pracuje
- World slouží k nastavení okolí, například barvy prostředí, stíny osvětlení a možnost přidání mlhy, síly gravitace
- Object zde se spravují vlastnosti objektů, jejich název, orientaci, velikost a jejich umístění ve scéně
- Constrains kontrola a omezení chování objektů mezi sebou
- Modifiers přidání nového modifikátoru a jeho nastavení. Nejpoužívanější modifikátory jsou array (pole), boolean (sloučení, průnik a rozdíl), mirroring (zrcadlení) a subdivision surface (vyklazení nerovností)
- Material a Textury spravování parametrů, které se týkají vzhledu objektů a jejich vlastností, například barva objektu, textura, druh materiálu, odraz světla
- Particles jsou to částicové systémy. Tisíce malých bodů, které lze využít u tvorby vlasů, trávy, ohně a mlhy

 Physics – možnost simulování fyzikálních vlastností objektu, jako je déšť a voda. Nastavení pohybující postavy, u které se nastavuje délka a výška kroku, výška výskoku a rychlost pádu po výskoku

(Blender dokumentace – Properties editor, 2017)

4.7.11 Python console

Je to konzole, která je interaktivní. V této konzoly je informace o verzi Pythonu, kterou Blender podporuje. Společně s modulem bpy (Blender Python modul) je možné ovládat Blender pomocí příkazů v jazyce Python. Nabízí možnost vytvářet a modifikovat objekty. Například při použití příkazu v konzoli "bpy.context.object.location.x += 0.5" je objekt posunut na ose X o 0,5 bodu. (Blender dokumentace – Python console, 2017)

4.7.12 Další editory

Zmíněné editory nejsou všechny, které se v Blenderu nacházejí. Existuje celá řada dalších editorů, jako například: NLA editor (Non-Linear Action), Outliner, Movie Clip editor, Video Sequence a další.

4.7.13 Game Engine

Je to nástroj, který se využívá při vytváření interaktivních aplikací a her, je to tzv. herní engine. Byl vytvořený v programovacím jazyce C++ a poskytuje velké množství funkcí. Nabízené funkce jsou renderování, animování modelů, umělá inteligence, skriptování a mnohé další. Blender Game Engine je určený pro vytváření real-time aplikací, což znamená, že renderuje scénu kontinuálně a díky tomu poskytuje interakci s uživatelem. Umožňuje to hlavně Logický editor pomocí senzorů, regulátorů a akčních členů. Výsledné aplikace lze spustit přímo v Blenderu pomocí interního přehrávače anebo je možné vyexportovat spustitelný soubor pro různé operační systémy. (Blender dokumentace – Game Engine, 2017)

Game engine není jediný engine v Blenderu. Jsou zde i konvenční enginy, které se nazývají Blender Render a Cycles Render. Oba se využívají pro tvorbu 2D obrázků a animací. První jmenovaný se v dnešní době už nevyvíjí a postupně se od něj upouští, ale stále si najde své věrné příznivce. Druhý jmenovaný je náročnější na hardware z důvodu komplikovanějšího zpracování světel. Je velmi vhodné použít při renderování GPU (Grafický čip). Výsledný obrázek nebo animace je mnohem realističtější. (Blender dokumentace – Cycles Render Engine, 2017)

Další možností je externí engine. Při této možnosti je Blender využit jen jako 3D modelovací program a vytvořené objekty vyexportovat do mnoha formátů. A následně importovat do externího enginu. Mezi nejznámější patří Unity Engine (Unity dokumentace, 2017), Stingray od společnosti AutoDesk a Unreal Engine. (Blender dokumentace – Importing and Exporting, 2017)

4.8 Porovnání programů

Všechny programy, které byly zmíněny jsou na vysoké úrovni. Každý program se liší a používá se pro jiné účely a v jiných pracovních oborech. V práci byly zahrnuty placené programy, freeware, složitější a profesionální programy. Z popsaných programů byl vybrán Blender, hlavně kvůli tomu, že je zcela zdarma, má širokou základnu uživatelů a má vlastní game engine, který je důležitý při vytváření interaktivní aplikace, nemusí se instalovat další modul anebo využívat engine třetích stran. Většina ostatních programů vlastní engine nemají, ale podporují export modelů pro enginy třetích stran. Svůj vlastní engine má již zmíněný Blender a 3ds Max, který se nazývá Stingray. Samozřejmě při výběru vhodného programu záleží na preferencích uživatele a také na tom, co zrovna potřebuje. Srovnání všech programů, které byly zmíněny výše je v Tabulce 1.

Program	Vývojář	Engine	Cena
Cinema 4D	Maxin	-	20 000 Kč – 80 000 Kč
3ds Max	AutoDesk	Stingray	35 000 Kč/rok
Blender	Blender Foundation	BGE	freeware
SketchUp	Trimble Navigation	-	freeware
LightWave	NewTek	-	18 000 Kč
Rhinoceros 3D	Robert McNeeel & Associates	-	5 000 Kč – 25 000 Kč

Tabulka 1: Porovnání programů

Pro tuto práci lze vybrat z několika 3D modelovacích programů. Zde jsou jejich největší výhody a nevýhody.

Cinema 4D

- Výhody: vysoký výkon, jednoduché ovládání
- Nevýhody: vysoká cena

3ds Max

- Výhody: stabilní vývoj, široká základna uživatelů
- Nevýhody: vysoká cena, nepodporuje češtinu

Blender

- Výhody: zdarma, vlastní game engine
- Nevýhody: přehozená funkcionalita tlačítek, špatná kvalita češtiny
 SketchUp
- Výhody: zdarma, jednoduché ovládání
- Nevýhody: horší kvalita modelů, málo funkcí

LightWave

- Výhody: jednoduchost, bezplatná technická podpora
- Nevýhody: vysoká cena, nepodporuje češtinu

Rhinoceros 3D

- Výhody: sleva pro studenty a školy
- Nevýhody: špatná možnost animací, nepodporuje češtinu

5 Metodika

Cílů práce bude dosaženo po dosažení následujících kroků:

- Seznámení se s programem Blender a jeho Game Enginem
- Sběr požadavků a jejich analýza
- Modelování 3D modelu a nasazení vhodné interakce
- Export a otestování výsledné aplikace

Prvním krokem bude seznámení se s prostředím Blenderu a jeho game enginu, kde proběhne vytváření interaktivního průchodu. Pro nastudování práce bude použita především dokumentace, kde je vše dobře vysvětleno. Dalšími vhodnými materiály jsou různé tutoriály, jak modelovat v Blederu. Pro nastudování Pythonu byly též použity různé tutoriály.

Dalším krokem bude sběr požadavků, který bude proveden na schůzi bytového domu, která se koná každý měsíc. Na této schůzi bude společně se zástupcem bytového domu přednesen návrh na rekonstrukci. Každý, kdo se zúčastní bude moci přednést své návrhy. Po sepsání všech nápadů, se kterými obyvatelé přišli, bude provedena analýza se zástupcem bytového domu. Při analýze budou vybrány jen kvalitní a realizovatelné nápady.

Jako další krok je modelování, pro které bude použita verze Blenderu 2.78. Práce bude vytvářena pomocí fotografií bytového domu, které budou sloužit jako základ bytového domu. Následné okolí a návrhy rekonstrukce, které budou vybrány, budou modelovány podle paměti a následně konzultovány. Po nastudování všech důležitých materiálů bude postupně modelována budova bytového domu, jeho chodby a bytu. Pro vytvoření interaktivního průchodu bude nutné nastavit kameru a zajistit její pohyb po scéně. Pohyb kamery, načítání přechodu mezi scénami bude naprogramováno pomocí skriptů v programovacím jazyce Python.

V posledním kroku bude výsledná aplikace vyexportována po dokončení scény a připojení skriptů. Aplikace bude spustitelná na operačních systémech Windows a volně přístupná ke stažení pro obyvatele bytového domu. Následná aplikace bude otestována na funkčnost a absenci chyb.

6 Vlastní práce

Po teoretickém úvodu, ve kterém jsme se seznámili s nástroji pro modelování ve 3D a především s nástrojem Blender a programovacím jazykem Python. V této části se budeme věnovat modelování budovy. Dále si ukážeme, jak naprogramovat v jazyce Python interaktivní průchod scénou, který bude realizován skriptem a jak vytvořit objektům animaci.

6.1 Analýza návrhů rekonstrukce

Návrhů, které obyvatelé domu přednesli bylo spousta, ale většina byla nerealizovatelná. Například nápad pro přistavění dalšího vchodu nelze realizovat, jelikož na něj není prostor ani peníze. Mezi realizovatelné nápady bylo vysázení více stromů okolo bytovky, opravení chodníku a jeho rozšíření kolem bytovky až k parkovišti. Mezi dalšími nápady bylo zateplení domu s novou fasádou a opravení i rozšíření parkoviště, které je malé, a v nedobrém stavu.

Analýza byla provedena se zástupcem bytového domu. Nejprve jsme vyřadili nesmyslné nápady, které byly chvílemi úsměvné, jelikož jsou ve vchodě vtipálci. Mezi nejvtipnější patřil vlastní pivovar na střeše bytovky. Dále jsme podle možností realizace a také z pohledu finanční stránky vybrali zateplení domu, oprava a rozšíření chodníku, zvětšení a oprava parkoviště, výměna vchodových dveří a zasazení stromů. Bytový dům má vlastní finanční rezervu, do které každý měsíc dávají obyvatelé domu určitou částku. Tato rezerva není však tak velká, aby pokryla všechny úpravy domu. Proto by bylo vhodné uvažovat o půjče. Toto již není předmětem této bakalářské práce.

6.2 Důležité nástroje

Nejčastěji používanou a nejdůležitější technikou pro modelování v Blenderu je Extrude Region neboli extrudování (vytažení). Platí zde jedno "zlaté pravidlo": Dát jednu ruku na myš a druhou na klávesnici. Toto pravidlo ušetří spoustu času, jelikož většina funkcí je dostupná pomocí klávesových zkratek, které jsou zmíněny již výše.

Pro pochopení je zde uveden jednoduchý příklad: Nejdříve si vložíme do scény plochu (klávesová zkratka Shift + A), následně se přepneme do editačního módu (tlačítko Tab), ve kterém je možné editovat jednotlivé vertexy, hrany a plochy objektu. Klávesou A (vybrání všech bodů) si označíme všechny hrany a stiskneme klávesu E (klávesa pro extrudování) a pohybujeme myší nahoru, až je vytvořen obdélník a potvrdíme levým tlačítkem myši. Opět stiskneme klávesu E, ale hned vzápětí klávesu S a při tažení ke středu objektu "vytažené" hrany zmenšíme. Poté opět opakujeme první krok s vytažením hran nahoru. Pro dokončení nám chybí už jen poslední krok, znovu stiskneme klávesu S a vytáhneme hrany nahoru a potvrdíme levým tlačítkem myši, zmáčkneme klávesu S a zmenšíme hrany. Vytvořili jsme jednoduchý patník. Při metodě extrudování často vznikají nežádoucí plochy navíc, pro odstranění použijeme funkci Remove Doubles, která je v postranním panelu 3D okna.

Dalším důležitým nástrojem je Make Edge/Face (Vytvoření hrany/Plochy), který je dostupný pod klávesovou zkratkou F. Při označení 2 vertexů se vytvoří jedna hrana, která je mezi těmito vertexy. Jestliže chceme vytvořit plochu musí být označeny alespoň 3 vertexy anebo alespoň 2 hrany. Často využívané nástroje v Blenderu jsou modifikátory, které jsou k nalezení v editore Properties. Tyto modifikátory umožňují automatické operace s objekty, které by bylo neefektivní anebo příliš časově náročné modelovat. Mezi nejčastěji používané modifikátory patří Subdivision Surface (vyhlazování objektu), Mirroring (zrcadlení), Array (pole kopií objektu), boolean (booleovské operace jako je rozdíl, sloučení nebo průnik) a Bevel (zkosení hran).

V neposlední řadě je důležitý nástroj zvaný UV mapování. Jedná se o nejlepší způsob, jak nanést v Blenderu na objekt texturu. K tomu slouží UV/Image editor. S objektem, na který se nanáší textura pracujeme v editačním módu. Další možností pro nanášení textur a materiálů je node editor, ve kterém pracujeme s uzly.

6.3 Modelování

Ještě, než začneme modelovat je vhodné si projekt rozdělit do několika scén, jelikož v jedné scéně by byl projekt nepřehledný a zbytečně složitý. Hlavní důvod rozdělení do scén je složitost modelu. Mohlo by se stát, že při spouštění aplikace by projekt "zamrznul" a nešel by spustit. Výsledná aplikace v našem případě obsahuje šest scén. První scéna zobrazuje bytový dům zvenčí, kde je možné si prohlédnout dům s venčí. Druhá scéna se odehrává v jedné z chodeb. V té jsou schody do jednotlivých pater a výtah. Třetí scéna umožňuje nahlédnout do jednoho z bytů, který bude vybaven nábytkem. Čtvrtá scéna zobrazuje GUI, které se objeví při spuštění aplikace a slouží ke spuštění interaktivního průchodu. Poslední dvě scény jsou pomocné. Použivájí se jako překrývací vrstvy pro první tři scén. V prvních třech scénách je kamera, která bude sloužit k interaktivnímu průchodu. Ve čtvrté scéně je kamera pouze statická.

Objekty v jednotlivých scénách byly vytvářené v 3D view editoru pomocí různých technik. Pro různé vymodelované objekty byly využívané různé Mesh objekty, jako jsou například Plane (plocha) Circle (kruh), Cylinder (válec), Circle (kruh), UV Sphere (Koule) a Cube (krychle), které byly přidávány v objektovém módu pomocí klávesové zkratky (Shift + A). V tomto módu se objekty upravovaly pomocí funkcí Rotation (rotace) a Scale (změna velikosti). Při přepnutí do editačního módu (klávesa Tab) se pro úpravu objektů zpřístupní další funkce Extrude, která slouží k "vytahování bodů".

Při modelování byly často využívané modifikátory. Nejčastěji používanými byly Array, Subdivision surface, mirroring, boolean a bevel. Složitější modely byly spojovány do skupin (klávesová zkratka Ctrl + G), což značně zpřehlední práci. Pro většinu objektů byly použity výše zmíněné funkce, ale v některých případech bylo potřeba použít i další funkce, které Blender nabízí, aby byl dosažen výsledek efektivně.

6.3.1 Venkovní scéna



Obrázek 2: Venkovní scéna bytového domu

1. Bytový dům

V první řadě byla vytvořena přední stěna bytového domu. To bylo provedeno přidáním objektu plocha, a to pomocí klávesové zkratky Shift + A. Následně byla do 3D view přidána fotografie části bytového domu, přesněji jednoho vchodu. Podle této fotografie byla v editačním okně upravena velikost plochy. Velikost plochy bylo možné upravit i pomocí posouvání bodů, označením hrany, která byla posunuta na určené místo. Pro vytvoření "díry" na dveře a okna bylo nutné nejprve rozdělit plochu více hranami, k tomuto kroku slouží klávesová zkratka Ctrl + R (přidání hrany na určité místo dovnitř objektu). Jakmile bylo vytvořeno potřebné množství hran, označila se plocha a pomocí funkce extrude byla plocha přemístěna dozadu na požadované místo a vznikla tak "díra" na dveře. Obdobně se takto vytvořila místa na okna.

Stejným způsobem byla vytvořena i zadní stěna bytového dumu, jen s jiným rozmístěním oken a dveří. Následně byly obě části spojeny bočními stěnami a střechou.

Tímto způsobem probíhalo vytvoření jedné části bytového domu. Druhou část by bylo zbytečně zdlouhavé modelovat stejným způsobem. Jelikož jsou obě části totožné, byl použit modifikátor zrcadlení. U modifikátoru zrcadlení se musí nastavit, podle které osy chceme objekt zrcadlit, jestli chceme výsledek sloučit do jednoho objektu, popřípadě zanechat textury původního objektu. Volitelným

parametrem je pak zrcadlový objekt, což je objekt, podle kterého chceme zrcadlit. V našem případě jsme zrcadlili pomocí osy X, zrcadlené objekty jsme sloučili a textury zanechaly. Jako zrcadlový objekt byl použit prázdný objekt, jenž není vidět.

2. Schody a zábradlí

Dále muselo být vytvořeno schodiště se zábradlím. Schodiště můžeme vymodelovat pomocí objektu Cube, který lze vložit do scény klávesovou zkratkou Shift + A. Krychle byla upravena na požadovanou velikost jednoho schodu pomocí funkce Scale (klávesová zkratka S) a posunu (klávesová zkratka G) na správné místo. Poté byl na objekt použit modifikátor array, který má v nastavení počet kopírování v poli, relativní cestu (nastavení, jak bude 2 a další objekt posunut), počátek pole a poslední složku pole. Následně bylo třeba přidat schodům zábradlí vymodelovaného pomocí válce (vložení pomocí Shift + A), který byl upraven na požadovanou velikost. Dále se zábradlí prodloužilo pomocí funkce extrudování, a společně s funkcí rotace bylo vytvořeno zaoblení, které muselo být vytvořeno z přechodu rovné plochy na schody. Tímto byla vytvořena vrchní část zábradlí. Prostřední část zábradlí jsme vytvořili pomocí zkopírování (klávesová zkratka Shift + D) vrchní části zábradlí a upravili její tloušťku a posunuli dolů. Jako poslední část byla vytvořena podstava pro zábradlí. To je obyčejný válec upraven na požadovanou tloušťku a výšku.

3. Dveře

Dalším krokem je vymodelování dveří. Ve scéně jsou různé typy dveří do bytu, vchodu, průchod na chodbě a výtahové dveře.

Budeme popisovat modelování jednoho typu dveří, a to sice dveří, které jsou umístěny na chodbě. Vložíme si pomocí klávesové zkratky Shift + A krychli. Tu si zvětšíme či zmenšíme podle toho jakou mají mít dveře velikost. Pomocí klávesové zkratky Ctrl + R přidáme hrany do krychle. Ty jsou umístěny tak, abychom mohli prostřední plochu vymazat a mohli místo ní umístit sklo. Mezi hranami, které byly u smazaných ploch, vznikly díry. Ty zaděláme pomocí klávesové zkratky F (vytvořit hranu/plochu. Nerovnosti na dveřích vytvoříme opět pomocí přidání hran a ty pak vtáhneme dovnitř dveří. Nakonec můžeme přidat sklo (jako 2 plochy), které je dvojité a má materiál, který má nastavenou průhlednost a odlesk na polovinu. U ostatních dveří, které vedou ven ze scény musíme nastavit materiál neprůhledný, jelikož za nimi ve scéně nic neexistuje. Dveřím je přiřazen materiál světlé dřevo, který jim byl přidán pomocí UV mapování. Ke dveřím bylo nutné vytvořit futra, ty byly vytvořeny stejným způsobem jako dveře, kterým byl přidán černý materiál.

Nakonec byly vytvořeny ke dveřím kliky a "kule". Klika byla vytvořena pomocí extrudování a rotace podle potřeb. Podstava "kule" byla vytvořena pomocí extrudování a samotná "kula" byla vytvořena pomocí krychle a přidáním modifikátoru subdivision surface. Materiál byl nastaven na "kov". Nastavení zrcadlení není bohužel v enginu Blender Game, proto je kov matný.

4. Okna

Vytvoření oken je v mnoha ohledech podobné dveřím. Také je zde potřeba nastavit neprůhlednost oken, které vedou pryč ze scény. Je to z toho důvodu, že venkovní a vnitřní prostředí jsou různé scény, tak by nic nebylo vidět. Oknům, která jsou zevnitř musíme vytvořit kliku pro otevírání, naopak pro venkovní pohled na okna nemusíme. Všechny okna jsou jen jeden typ, který se liší jen svými rozměry.

5. Chodník

Chodník byl vytvořen vložením krychle, která slouží jako patník. Krychle byla upravena na požadovanou velikost a požadovaný tvar pomocí funkce extrude a zvětšování a zmenšování. Další možností, jak jej upravit na požadovaný tvar je zaoblení (bevel). Následně byl na patník použit modifikátor pole (array). Poté bylo pole patníků pomocí klávesové zkratky Shift + D duplikováno a pomocí klávesy G posunuto, aby tvořilo patníky chodníku. Mezi patníky byla vložena plocha, na kterou byla aplikována textura dlažebních kostek.

6. Objekt Empty

Empty objekty se musí vyskytovat ve všech scénách, kde bude programován interaktivní průchod. Slouží jako rodičovský objekt pro kameru, je to pro to, aby kamera nemohla procházet objekty. Umožňují kameře pohyb po scéně a do schodů.

7. Stromy

Stromy jsou vytvořeny pomocí generátoru stromů (Sapling Tree Generator), který obsahuje Blender již v instalaci on verze 2.7. Tento generátor slouží, k už zmíněnému generování stromů, ale také ke generování rostlin. Generátor stromů je nutné povolit v nastavení User Preferences.



Obrázek 3: nastavení nového stromu

Po přidání stromu do scény se výsledný strom nastavuje v levém panelu v 3D

view, kde je možné každý strom nastavit (viz. obrázek 3), od vyhlazení až po nastavení listů. Stromy v našem projektu nelze nastavovat příliš detailně, jelikož čím detailnější strom, tím více ploch v modelu a mohlo by to zapříčinit sekání výsledné aplikace.

8. Terén

Do scény byl vložen objekt plocha, který byl přizpůsoben scéně. Na takto vzniklou plochu jsme použili několikrát funkci Subdivide. Tato funkce všechny plochy rozdělí na 4 menší plochy. Podle počtu dělení se dělí následné menší plochy na další 4 části, čím více použijeme rozdělení tím detailnější bude výsledný terén. Ale musíme myslet na to, že čím více rozdělení tím větší bude náročnost výsledné aplikace. Kopcovité terény lze v Blendru modelovat pomocí módu zvaným Sculpt. V tomto módu je několik nastavení, například typ zvedání.



Obrázek 4: Nastavení terénu

9. Další objekty tvořící výslednou scénu

Mezi další objekty patří lavička, auta, vrata od garáže, hromosvod na bytovém domu a parapety. Všechny tyto objekty jsou tvořeny pomocí již více zmíněných funkcí extrude, posunu a rotace. Byly použity různé typy modifikátorů.

6.3.2 Scéna chodba

Scéna chodba je vymodelována podobným způsobem jako předchozí scéna. Stěny jsou vloženy jako obyčejné plochy, boční stěny se pomocí klávesy S zvětší na požadovanou velikost. Zadní a přední stěna musí být pomocí extrude protažena, jelikož obsahuje dveře a okna. Dále se vytvoří prostor na okna a dveře jako to bylo u venkovní scény. Dalším prvkem této scény je výtah, který je tvořen stěnami. Mezi těmito stěnami je výtah, který má podlahu, strop a boční stěny. Následně jsou tady umístěna tlačítka na zvolení jednotlivých pater.



Obrázek 5: Schody scény chodba

Největším prvkem jsou v této scéně schody, jejichž vytvoření je popsáno v předchozí scéně. Jediný rozdíl je ve vytváření zábradlí, v nemž je sklo, které je poloprůhledné a je připevněno ke schodům pomocí kovových úchytů a má dřevěné držadlo. Podlaha je zde vytvořena pomocí krychle, která je modifikována podle potřeb a je na ni nasazena textura pomocí UV mapování. Okna a dveře nemusíme znovu vytvářet, jelikož jsou vytvořeny již v předchozí scéně, a tak je zbytečné je vytvářet znovu. Okna mají navíc kliku na otevírání. Jako poslední objekt v této scéně jsou dveře od výtahu, které byly vytvořeny pomocí funkce extrude a modifikace jako je rotování a změna velikosti.

S postupným vytvářením objektů se panel Outliner (panel, kde se objevují všechny objekty) stával nepřehledným, a tak bylo nutné objekty rozdělit do skupin. Tímto krokem se stalo modelování zase přehlednějším. Výhoda rozdělení objektů do skupin je, že se nemusí některá nastavení provádět na každý objekt zvlášť, ale může se nastavení aplikovat na celou skupinu.

6.3.3 Scéna byt

Tato scéna je modelována stejným způsobem jako je chodba. Venkovní stěny jsou vloženy jako obyčejné plochy, které se upravily na požadovanou velikost. Ve stěnách směřujících ven byly vytvořeny "díry", které slouží pro vložení oken. Dále byly vlo-

ženy stěny na boky. Na levém boku se prováděla jen změna velikosti. Avšak to na druhé straně musela být do stěny "vyříznuta díra", která bude sloužit pro vložení dveří. Následně byla stěna protažena pomocí funkce extrude. Okna a dveře byly stejně jako v předchozí scéně zkopírovány, jelikož by bylo zbytečné je modelovat znovu.



Obrázek 6: Drátěný model křesla

Pro tuto scénu byl vytvořen jednoduchý nábytek jako jsou skříně, postele, křeslo, židle a stůl. Pro příklad si uvedeme modelování skříně. Vložíme si krychli pomocí klávesové zkratky Shift + A, poté se přepneme do editačního módu, upravíme velikost skříně podle potřeby. Následně pomocí klávesové zkratky Ctrl + R přidáme dvě hrany doprostřed krychle na stranu, která bude zepředu. Vybereme plochu, která vznikne mezi těmito hranami a pomocí klávesy E vtáhneme plochu kousek dovnitř, tím vytvoříme efekt dvířek. To stejné uděláme zboku, kde umístíme nové hrany blízko kraje. V tomto případe, ale použijeme klávesu E a následně po ní klávesu S, která nám zajistí zmenšení plochy. Základ skříně máme, teď už jen vymodelovat otevírání. To vytvoříme pomocí vložení tří válců, které upravíme na požadovanou velikost a dva z těchto válců umístíme na přední stranu skříně vodorovně a třetí svisle. Poslední krok je přidání textury na skříň, tou bude dřevo a na úchyt použijeme kovový materiál.

6.3.4 Materiály

Ke každému objektu je vytvořený vhodný materiál. Materiály byly vytvořeny v editoru Properties v záložce Material. V této záložce se může nastavit barva a další parametry, kterými jsou průhlednost, odrazivost a stínování. Práce je vytvářena v Game Enginu, ve kterém není možné vytvořit tak kvalitní materiály, jako tomu je v enginu Cycles, ve kterém objekty vypadání o mnoho reálněji.

6.3.5 Osvětlení

Pro osvětlení scény nestačí pouze jeden zdroj světla. Je třeba použít několik světelných zdrojů. Blender nám nabízí několik variant světelných zdrojů, přesněji pět různých druhů: Sun, Spot, Point, Area a Hemi. Světelné zdroje jsou jedny z nejdůležitějších částí výsledné aplikace. (Pokorný, 2009)

Vložení světel se provádí stejně jako vkládání Mesh objektů. Příkazem Shift + A. Také posunování (klávesa G) a u některých typů se dokonce využívá rotace (klávesa R). Nastavení světelných zdrojů vyvoláme klávesou F5. Game Engine nenabízí příliš mnoho možností stínování pomocí světel. Jediné světelné zdroje, které vrhají stíny jsou Sun a Spot.

6.4 Logika aplikace

Logika aplikace se skládá z několika částí, které se provádí v Logic editoru. Popis tohoto editoru nalezneme výše. Nejprve bylo nutné vytvořit průchod scénou a nasazení do naší aplikace v již zmíněném Logic editoru. Jako další krok bylo vhodné vytvořit otáčení kamery pomoci myši, jelikož jen strohé chození sem a tam a nemožnost podívat se nahoru a dolů by konečným uživatelům neumožnilo se pořádně rozhlédnout. Jako další část bylo nutné vytvořit animaci dveří u kterých bude možnost otevírání. Tato animace byla vytvořena pomocí 3D View, Timeline, Graph Editor a v neposlední řadě Dope Sheet editorů.

6.4.1 Možnosti interaktivního průchodu

Pro tvorbu interaktivního průchodu existují dvě možnosti, které lze využít. Ještě před samotnými možnostmi musíme provést tyto kroky, které jsou společné pro obě možnosti. Ke kameře musíme přiřadit rodiče pomocí klávesové zkratky Ctrl + P. Tento rodič je objekt Empty. Důvod je jednoduchý, kamera musí mít nastavenou fyziku na static, což nám zabrání, aby po zapnutí aplikace kamera spadla na zem. Ale pro pohyb po scéně potřebujeme, aby se hýbala. A to nám umožní objekt Empty, který má nastavenou fyziku na dynamic. A jelikož je Empty nastaven jako rodič kamery tak se kamera bude pohybovat stejně. Objekt Empty nám umožní nejen pohyb po scéně, ale zabrání průchodu zdmi.

• Nastavení v Logic editoru

Nastavení pomocí tohoto editoru se provádí pomocí senzorů, regulátorů a akčních členů. Pro vytvoření interaktivního průchodu je nutné přidat 4 senzory pro klávesy, kterými bude prováděn průchod a dvou senzorů myši. Následuje stejný počet regulátorů a akčních členů. V akčních členech se nastaví vzdálenost, kterou chceme, aby po stisknutí klávesy kamera udělala. Toto řešení je neefektivní

Sensors 🕴 🗹 Sel 🗹 Act	🗹 Link 🗹 State	Controllers	🗹 Act 🛛 🗹 Link	Actuators 🕴 🗹 Sel 🛛 🗹 Act	🗹 Link 🗹 State
Empty	Add Sensor 🕴	0 Empty	Add Controller	Empty	Add Actuator
🗘 Keyboard Keyboard 🔬		And Nand 1		Motion Motion	x av 🛛 🕅
🗘 Keyboard Keyboard 📌		And Nand 1		Motion Motion.001	* 🗢 🛯 🕅
🗘 Keyboard Keyboard 📝		And Nand 1		Motion Motion.002	* 🗖 🛛 🕅
👂 Keyboard Keyboard 🖉		And Nand 1		Motion Motion.003	* 🗢 🛚 🛞
View Add					

při použití více scén a tím pádem více kamer. Museli bychom nastavovat každou kameru stejným způsobem, a to je příliš zdlouhavé.

Obrázek 7: Chůze implementována v Logic Editoru

• Vytvoření skriptu

Druhým způsobem vytvoření interaktivního průchodu je pomocí skriptů. Efektivita tohoto řešení je napsání jednoho skriptu a možnost použití na několik kamer. Pootáčení kamery, která má při předchozím způsobu dva senzory, dva regulátory a dva akční členy. Při použití skriptu se nám počet senzorů a regulátorů sníží na jeden. Bohužel akční členy budou i nadále dva, jelikož potřebujeme otáčet kamerou na všechny strany. Pro psaní skriptů lze využít Text editor, který usnadňuje práci pomocí zvýraznění syntaxe anebo číslování řádků. Pro zobrazení chybového hlášení slouží systémová konzole, která je k zobrazení v Info editoru (v záložce okno otevřeme Toggle Systém Console). Skripty se spouští pomocí Logic editoru, který se vloží jako regulátor typu Python, do kterého vstupují senzory a vystupují akční členy.



Obrázek 8: Chůže implementována pomocí Python scriptu

6.4.2 Animace dveří

Jak již bylo zmíněno animace dveří se provádí pomocí 4 editorů, které jsou zmíněny výše. Nejdříve bylo nutné všem částem dveří (klika a zámky) přiřadit rodičovský objekt dveře. Tím zajistíme animaci všech částí dveří. Nejdříve bylo nutné přenést ohnisko objektu na kraj dveří pro snazší rotaci. Toho se dalo dosáhnout i přidáním rodičovského objektu dveřím, který byl zmenšen a umístěn na kraj dveří.

🗢 🖉 Dope Sheet Summary			0	•			•	
▼ ▼ dvere.001			\diamond	\diamond			Output	
¬ → dvere.001Action			\diamond	\diamond			0	
Rotation	🖉 📢 🔓		>	\diamond				
X Euler Rotation	۵ 🕩 🎤		\diamond	\diamond				
Y Euler Rotation	الا 🖉 🖉		>	\diamond				
Z Euler Rotation	الا 🖉 🖉		>	\diamond				
			\diamond	~				
¬ → → → → → → → → → → → → → → →	n		\diamond	•				
	De 📣 🕞		\diamond	O				
X Euler Rotation	🖉 🕩 🦢		\	\diamond				
Y Euler Rotation	🖉 🕩 🦢		\	\diamond				
Z Euler Rotation	۵ 🕩 🎤		\diamond	~				
		20	0	20	40	60	80	100

Obrázek 9: Animace dveří v Dope Sheet editoru

V editoru Dope Sheet nastavíme kurzor na pozici 1. snímku. V okně 3D View použijeme klávesovou zkratku I (vkládání klíčového snímku) a zvolíme rotace. Tímto krokem se nám nastaví dveře do výchozí pozice. Následně nastavíme kurzor na pozici 30 snímků (Dope Sheet editor) a kolem osy Z rotujeme dveře o 90° (3D View editor) a vložíme klíčový snímek. Po dokončení rotace posuneme kurzor na pozici o 70 snímků dále, kde vložíme klíčový snímek. Totéž jako u otevírání dveří uděláme u zavírání dveří. Mezi snímky 1 až 30 probíhá animace otevírání dveří, mezi 30 a 90 snímkem zůstávají dveře otevřené a od 90 snímku do konce probíhá animace zavírání dveří. Výsledná délka animace má 120 snímků.

K otevření dveří jsme udělali první krok. Potom máme 2 možnosti. První je nastavení jen v Logic Editoru pomocí senzorů, regulátorů a akčních členů. Senzory jsou přesněji Keyboard a Near. Jako akční člen použijeme F-Curve, kde nastavíme naši vytvořenou animaci a propojíme ho se senzory pomocí regulátoru and. Druhou možností je napsat skript a pomocí logic editoru ho napojit na dveře.

6.4.3 Skripty v Pythonu

Pro interaktivní průchod a otevírání dveří byly použity skripty v programovacím jazyce Python. Pro které lze využít Text editor, jenž umožňuje popis řádků a zvý-

raznění syntaxe. Jsou to výhody, které usnadňují práci. Vytváření průchodu a animací všech dveří by bylo zbytečně zdlouhavé a neefektivní. K propojení skriptů a modelu v Logic editoru slouží regulátor typu Python, do kterého vstupují senzory a vystupují akční členy. Systémová konzole je prospěšná při debuggování.

• Skript 1

```
import bge
from bge import render
from mathutils import Vector
render.showMouse(1)
def main():
    cont = bge.logic.getCurrentController()
    own = cont.owner
    scene = bge.logic.getCurrentScene()
    mys = cont.sensors ["mys"]
    rotacex = cont.actuators ["rotacex"]
    rotacez = cont.actuators ["rotacez"]
    cont.activate(rotacex)
    cont.activate(rotacez)
    x = render.getWindowWidth()//2
    y = render.getWindowHeight()//2
    stred_obraz = (x, y)
    stred = Vector(stred_obraz)
    mys pozice = Vector(mys.position)
    hodnota = (mys_pozice - stred) * -0.002
    rotx.dRot = [hodnota.y, 0, 0]
    rotz.dRot = [0, 0, hodnota.x]
    render.setMousePosition(x, y)
main()
```

Výše zmíněný skript funguje na základě pohybu myši. Jestliže se myš pohybuje do stran, kamera kopíruje stejný pohyb. Totožné je to při pohybu nahoru a dolů.

• Skript 2

```
import bge
def main():
    cont = bge.logic.getCurrentController()
    own = cont.owner
    scene = bge.logic.getCurrentScene()
    vpred = cont.sensors["vpred"]
    vzad = cont.sensors["vzad"]
    vlevo = cont.sensors["vlevo"]
    vpravo = cont.sensors["vpravo"]
    pohyb = cont.actuators["pohyb"]
    kam = [0, 0, 0]
    kolik = 0.05
    if (vpred.positive):
        \operatorname{kam}[2] = -\operatorname{kolik}
    if (vzad.positive):
        kam[2] = kolik
    if (vlevo.positive):
    kam[1] = kolik
    if (vpravo.positive):
        kam[1] = -kolik
    move.dLoc = loc
    cont.activate(move)
main()
```

Tento skript slouží za pomocí tlačítek k pohybu scénou do všech směrů. V Logic editoru se vloží senzory všech kláves a propojí se s akčním členem pohyb pomocí regulátoru typu Python.

```
• Skript 3
```

```
import bge
def main():
    cont = bge.logic.getCurrentController()
    own = cont.owner
    scene = bge.logic.getCurrentScene()
    klavesnice = bge.logic.keyboard
    vpred = bge.logic.KX_SENSOR_ACTIVE ==
        keyboard.events[bge.events.UPARROWKEY]
    vzad = bge.logic.KX_SENSOR_ACTIVE ==
        keyboard.events [bge.events.DOWNARROWKEY]
    vlevo = bge.logic.KX SENSOR ACTIVE ==
        keyboard.events [bge.events.LEFTARROWKEY]
    vpravo = bge.logic.KX_INPUT.ACTIV ==
        keyboard.events[bge.events.RIGHTARROWKEY]
    kolik = 0.05
    if vpred:
        own.applyMovement((0, kolik, 0), True)
    if vzad:
        own.applyMovement((0, -kolik, 0), True)
    if vlevo:
        own.applyMovement((kolik, 0, 0), True)
    if vpravo:
        own.applyMovement((-kolik, 0, 0), True)
main()
```

Skript určený taktéž pro pohyb pomocí tlačítek, ale už nemusíme vkládat tolik senzorů. V pořádku si vystačíme s jedním senzorem typu Always, který bude propojen do regulátoru typu Python.

```
• Skript 4
```

```
import bge
def main():
    cont = bge.logic.getCurrentController()
    own = cont.owner
    scene = bge.logic.getCurrentScene()
    otevrit = cont.actuators ["otevrit"]
    zavrit = cont.actuators ["zavrit"]
    kliknuti = cont.sensors ["klavesa_otevreni"]
    vzdalenost = own.getDistanceTo(
        scene.objects ["hrac"])
    if vzdalenost > 5 and own ["otevrene"] == True:
        cont.activate(zavrit)
        own ["otevrene"] = False
    if vzdalenost < 5 and own ["otevrene"] == False:
        if kliknuti.positive:
            cont.activate(otevrit)
            own ["otevrene"] = True
main()
```

Tento skript je určený pro otevírání a zavírání dveří. Slouží, aby v určené vzdálenosti bylo možné otevřít dveře.

```
• Skript 5
```

```
import bge
def main():
        cont = bge.logic.getCurrentController()
        own = cont.owner
        scene = bge.logic.getCurrentScene()
        klavesnice = bge.logic.keyboard
        scena prepnout = cont.actuators ["scene"]
        scena_venek = bge.logic.KX_SENSOR_ACTIVE ==
            keyboard.events [bge.events.VKEY]
        scena_chodba = bge.logic.KX_SENSOR_ACTIVE ==
            keyboard.events [bge.events.BKEY]
        scena_byt = bge.logic.KX_SENSOR_ACTIVE ==
            keyboard.events[bge.events.NKEY]
        if scena_venek and own ["aktualni"] == 1:
                scena_prepnout.scene = "venkovni_scena"
        if scena_chodba and own ["aktualni"] == 2:
                scena_prepnout.scene = "scena_chodba"
        if scena_byt and own ["aktualni"] == 3:
                scena prepnout.scene = "scena byt"
        cont.activate(scena_prepnout)
main()
```

Skript, který umožňuje načítání různých scén pomocí tlačítek.

6.4.4 Export

Posledním krokem je export výsledného projektu do spustitelného souboru. Aby bylo možné toto provést musí se nejdříve povolit volby "Save As Game Engine Runtime" a "Game Engine Publishing" v nastavení User Preferencies. Export provedeme výběrem možnosti File -> Export -> Save As Game Engine Runtime. Následně vybereme lokaci, kam chceme soubor uložit. Export nepodporuje českou diakritiku, musíme proto zajisti, aby cesta k souboru neobsahovala takové znaky. Výsledný čas nebyl nikterak dlouhý a velikost souboru se vyšplhala na stovky Mega Bajtů.

7 Diskuse

7.1 Praktické využití

Jako výstup této práce je aplikace pro platformu Microsoft Windows a slouží obyvatelům bytového domu (dále jen domu) jako možný návrh rekonstrukce a úpravy domu a okolí. Každému obyvateli domu bude výsledná aplikace poskytnuta prostřednictvím e-mailu. V pohodlí domova si budou moci projít dům a jeho chodbu. Na další schůzi, která se koná na začátku prázdnin budou jejich prosby vyslyšeny a případně do modelu dopracovány. Dále může sloužit pro případné zájemce o koupi bytu.

Další praktický přínos vidím ve využití práce developerskými firmami anebo realitními kancelářemi. Stačilo by jen podle této práce vymodelovat dům, chatu, kancelářskou budovu a další. Použít skripty z této práce a výslednou aplikaci umístit na svoje webové stránky a budoucí zákazníci by se mohli procházet. Dále by bylo možné rozšíření práce pro hudební festivaly jako propagaci. Také by podle návodu vymodelovaly pódium, stánky a atrakce. Následně použít skripty a návštěvníci by se mohli procházet.

7.2 Co se povedlo

Jako první, co se povedlo bylo vytvoření modelů. Popis, jak se modelovaly je v sekci modelování. Druhou částí bylo vytvoření materiálů a textur, které byly nasazeny na vymodelované modely. Naposledy byla vytvořena interakce s uživatelem, toho bylo docíleno pomocí Logic Editoru a skriptů v jazyce Python.

7.3 Co se nepovedlo

Největšími nedostatky jsou hlavně méně kvalitní materiály. Pro vytváření materiálů slouží Node Editor, ve kterém je práce podstatně složitější. Dále Node Editor v Game Enginu neobsahuje všechny možnosti, které obsahuje v ostatních enginech (Cycles Engine a Blender Render Engine). Nepodporovaná nabídka v BGE je hlavně "mirror" efekt, což je efekt pro odraz světla a využívá se hlavně pro kovové povrchy a je náročný na výpočet. Je to spojené především s velkou výpočetní náročností BGE (Blender Game Engine) při real-time renderování. Ve výsledné aplikaci museli být stromy odstraněny, již 7 stromů způsobovalo trojnásobný nárůst aplikace. Aplikace se stromy se spouštěla dlouho a objevovalo se občasné sekání. Bylo to hlavně z důvodu velkého množství ploch které stromy obsahovali.

7.4 Problémy při vytváření aplikace

Největším problémem byla při vytváření inteaktivního průchodu "rozházenost" modelů při spuštění aplikace. Například u dveří, zmizela "kula" a zámek. To bylo vyřešeno spojením objektů pomocí klávesové zkratky Ctrl + J (Join). Dalším nemalým problémem bylo nevykreslování některých ploch, například celá zadní stěna bytového domu zmizela a byly k vidění jen okna. Zde už šlo o větší problém, jelikož na něj muselo být aplikováno větší množství oprav. Prvvním krokem bylo nastavení kamery, konkrétně parametru Clipping, jehož koncovou hodnotu bylo nutné zvýšit. Další krok spočíval v nastavení parametru Backface Culling v 3D View editoru. Tento parametr zajistí aby se plochy vyktereslovyli nejen ze předu, ale i ze zadu. Jako poslední krok jsme si nastavili v editačním okně viditelné normály. Poté byla použita klávesová zkratka Ctrl + N (Make Normal Consistent). Tři výše zmíněné kroky jak zpravit objekty, aby se vykreslovaly, pomohli u většiny objektů. Bohužel některé objekty nebyly opraveny ani těmito kroky.

7.5 Možné rozšíření

Možným rozšířením se nabízí vytvoření propracovanější funkcionality. Dále by bylo vhodné vymodelovat všechny byty, které jsou v bytovém domě. Jako zpestření aplikace by mohla být vytvořena postavička, která by se pohybovala jako reálny člověk. Mohli by být vytvořeny kvalitnější textury a materiály. Tohoto kroku by bylo možné dosáhnout pomocí skriptů a modulu Video Texture. Tento modul nabízí více možností. Dále by mohla být aplikace exportována jako webová aplikace. K tomu slouží engine s názvem Blend4Web. Tento engine využívá WebGL a další technologie prohlížeče. (Blend4Web, 2017)

8 Závěr

Cílem této bakalářské práce bylo vytvořit 3D model bytového domu s následným přidáním interakce pro uživatele. Teoretická část této práce obsahuje popis Blenderu, jeho editorů a jeho konkurence. Popisuje se zde uživatelské rozhraní, které je specifické oproti konkurenci a klávesové zkratky, které výrazně usnadňují a zrychlují práci. Praktická část práce popisuje, jak probíhalo samotné modelování 3D objektů a jejich rozdělení do scén. Následně bylo popsáno, jakým způsobem lze vytvořit interakci uživatele s výsledným modelem. Výsledný stav aplikace uživatelům umožňuje interaktivní průchod scénami. Umožňuje podívat se na výsledný model bytového domu zvenčí a zevnitř. Jak bylo zmíněno v diskusi museli být odstraněny z výledné aplikace stromy. Aplikace byla zabalena do archivu a je dostupná na webové adrese: https://akela.mendelu.cz/ xbocek/

8.1 Použité nástroje

Pro modelování byl vybrán program Blender, který obsahuje svůj Game Engine a byl vybrán hlavně díky již zmíněnému enginu a jeho dostupnosti. Dokáže v reálném čase renderovat objekty a díky tomu i vytvářet interaktivní aplikace. Výčet nejpoužívanějších nástrojů:

- 3D View Editor
- Logic Editor
- Timeline
- UV/Image Editor
- Node Editor

9 Reference

- BLENDER ABOUT Blender. [online]. 2017 [cit. 2017-05-11]. Dostupné z: https://www.lightwave3d.com/overview/.
- ABOUT LIGHTWAVE *LightWave.* [online]. 2015 [cit. 2017-05-11]. Dostupné z: https://www.lightwave3d.com/overview/.
- AUTODESK INC AutoDesk Stingray. [online]. 2017 [cit. 2017-05-11]. Dostupné z: http://www.autodesk.com/products/stingray/overview.
- AUTODESK INC 3ds Max. AutoDesk. [online]. 2017 [cit. 2017-05-11]. Dostupné z: http://www.autodesk.com/products/3ds-max/overview.
- BLENDER Blender dokumentace import export. Blender. [online]. 2017 [cit. 2017-05-11]. Dostupné z: https://docs.blender.org/manual/ko/dev/data_system/files/import_export.html.
- BLENDER Blender dokumentace obrazovky/uživatelské rozhraní. Blender. [online]. 2017 [cit. 2017-05-11]. Dostupné z: https://docs.blender.org/manual/es/dev/interface/window_system/screens.html.
- BLENDER Blender dokumentace 3DView. Blender. [online]. 2017 [cit. 2017-05-11]. Dostupné z: https://www.blender.org/manual/editors/3dview/index.html.
- BLENDER Blender dokumentace Timeline. Blender. [online]. 2017 [cit. 2017-05-11]. Dostupné z: https://www.blender.org/manual/editors/timeline.html.
- BLENDER Blender dokumentace Dope Sheet. Blender. [online]. 2017 [cit. 2017-05-11]. Dostupné z: https://www.blender.org/manual/editors/dope_sheet/introduction.html.
- BLENDERBlenderdokumentace-Historie.Blen-der.[online].2017[cit.2017-05-11].Dostupnéz:https:https://docs.blender.org/manual/en/dev/getting_started/about/history.html.
- BLENDER Blender dokumentace Logic Editor. Blender. [online]. 2017 [cit. 2017-05-11]. Dostupné z: https://www.blender.org/manual/editors/logic_editor.html.
- BLENDER Blender dokumentace Text Editor. Blender. [online]. 2017 [cit. 2017-05-11]. Dostupné z: https://www.blender.org/manual/editors/text_editor.html.
- BLENDERBlenderdokumentace-GraphEditor.Blen-der.[online].2017[cit.2017-05-11].Dostupnéz:https://www.blender.org/manual/editors/graph_editor/introduction.html.

- BLENDER Blender dokumentace Info Editor. Blender. [online]. 2017 [cit. 2017-05-11]. Dostupné z: https://docs.blender.org/manual/de/dev/editors/info/index.html.
- BLENDER Blender dokumentace modifikátory. Blender. [online]. 2017 [cit. 2017-05-11]. Dostupné z: https://wiki.blender.org/index.php/Doc:CZ/2.6/Manual/Modifiers.
- BLENDER Blender dokumentace Blender Render Engine. Blender. [online]. 2017 [cit. 2017-05-11]. Dostupné z: https://www.blender.org/manual/render/blender_render/index.html.
- BLENDER Blender dokumentace Cycles Render Engine. Blender. [online]. 2017 [cit. 2017-05-11]. Dostupné z: https://www.blender.org/manual/fr/render/cycles/index.html.
- BLENDER Blender dokumentace Blender Game Engine. Blender. [online]. 2017 [cit. 2017-05-11]. Dostupné z: https://www.blender.org/manual/game_engine/introduction.html.
- BLENDER Blender dokumentace UV/Image Editor. Blender. [online]. 2017 [cit. 2017-05-11]. Dostupné z: https://www.blender.org/manual/editors/uv_image/uv_editing.
- BLENDER Blender dokumentace Node Editor. Blender. [online]. 2017 [cit. 2017-05-11]. Dostupné z: https://www.blender.org/manual/editors/node_editor/introduction.html.
- BLENDER Blender dokumentace Python console. Blender. [online]. 2017 [cit. 2017-05-11]. Dostupné z: https://www.blender.org/manual/editors/python_console.html.
- BLENDER Blender dokumentace Properties editor. Blender. [online]. 2017 [cit. 2017-05-11]. Dostupné z: https://www.blender.org/manual/de/editors/properties/introduction.html.
- BLENDER3D.CZ Blender3D. [online]. [cit. 2016-05-02]. Dostupné z: http://www.blender3d.cz/drupal/?q=charakteristika.
- DIGITAL MEDIA S.R.O. Cinema4D. Cinema4D. [online]. 2017 [cit. 2017-05-11]. Dostupné z: http://www.cinema4d.cz/.
- Ewox Co je to open source. [online]. 2017 [cit. 2017-05-11]. Dostupné z: http://www.ewox.sk/co-je-to-opensource.
- KŘÍŽ, JAN Mistrovství v 3ds Max: [kompletní průvodce profesionálního grafika]. Brno: Computer Press, 2010. Mistrovství. ISBN 978-80-251-2464-2.

- MATOUŠEK, Ivo Počítačová grafika a CAD: základy geometrického modelování I. V Liberci: Technická univerzita, 2015. ISBN 978-80-7494-251-8.
- POKORNÝ, PAVEL Blender: naučte se 3D grafiku. 2., aktualiz. a rozš. vyd. Praha: BEN - technická literatura, 2009. ISBN 80-7300-244-2.
- RHINOCAD.CZ Rhinoceros. RhinoCAD.cz [online]. [cit. 2016-05-02]. Dostupné z: http://www.rhinocad.cz/produkty/mcneel/rhino/.
- ROBERT MCNEEL ASSOCIATES Rhinoceros 3D. Rhinoceros 3D. [online]. 2017 [cit. 2017-05-11]. Dostupné z: http://www.rhino3d.com/.
- SLABÝ, ANTONÍN Počítačová grafika. Hradec Králové: Gaudeamus, 2012. ISBN 978-80-7435-208-9.
- SMUTNÝ, L Interaktivita Co je interaktivita. In: Dmarketing: Magazín digitálního marketingu [online]. 2010 [cit. 2017-05-11]. Dostupné z: http://www.dmarketing.cz/2010/01/i-interaktivita-co-je-interaktivita/.
- TRIMBLE NAVIGATION SketchUp. SketchUp. [online]. 2017 [cit. 2017-05-11]. Dostupné z: http://www.sketchup.com/.
- UNITYTECHNOLOGIESUnitydokumentace.Unity.[online].2017[cit.2017-05-11].Dostupnéz:http://docs.unity3d.com/Manual/HOWTOImportObjectBlender.html.
- WIKIPEDIA Počítačová 3D grafika. Wikipedia. [online]. 2017 [cit. 2017-05-11]. Dostupné z: https://goo.gl/RBbijs.

Přílohy

A Ukázky modelů



Obrázek 10: Úvodní obrazovka aplikace



Obrázek 11: Sedací nábytek v obývacím pokoji



Obrázek 12: Ukázka z aplikace 1



Obrázek 13: Ukázka z aplikace2

B Přílohy na CD

Přiložený kompaktní disk obsahuje:

- archivovaný soubor s aplikací
- Práci v elektronické podobě