



**ČESKÉ VYSOKÉ  
UČENÍ TECHNICKÉ  
V PRAZE**

**F3**

**Fakulta elektrotechnická  
Katedra počítačové grafiky a interakce**

**Bakalářská práce**

# **Prohlížečka 3D modelů ve virtuální realitě**

**Petr Šimek**

**Otevřená informatika - Počítačové hry a grafika**

**Květen 2022**

<https://gitlab.fel.cvut.cz/langweil-mmp/vrmodelviewer>

**Vedoucí práce: Ing. David Sedláček, Ph.D.**





## I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Šimek** Jméno: **Petr** Osobní číslo: **492200**  
Fakulta/ústav: **Fakulta elektrotechnická**  
Zadávající katedra/ústav: **Katedra počítačové grafiky a interakce**  
Studijní program: **Otevřená informatika**  
Specializace: **Počítačové hry a grafika**

## II. ÚDAJE K BAKALÁŘSKÉ PRÁCI

Název bakalářské práce:

**Prohlížečka 3D modelů ve virtuální realitě**

Název bakalářské práce anglicky:

**VR 3D model viewer**

Pokyny pro vypracování:

- 1) Seznamte se s 3D modelem Langweilova modelu Prahy, s použitými formáty a organizací dat.
- 2) Seznamte se s aplikacemi pro virtuální realitu (VR) určenými pro prohlížení 3D modelů (např. Sketchfab, VR model viewer, různá VR muzea a galerie).
- 3) Dle provedené rešerše (2) vyberte interakční techniky, nebo navrhnete nové, vhodné pro výběr a prohlížení jednotlivých dílů modelu a/nebo skupin dílů ve virtuální realitě. Předpokládejte VR náhlavní soupravu s ovladači.
- 4) Navržené techniky implementujte a otestujte s cílovou skupinou uživatelů.
- 5) Zjistěte výkonostní nároky s ohledem na velikost Langweilova modelu a navrhnete optimalizační techniky (např. LOD, využití postupného načítání, streamování dat, vhodná úprava dat).
- 6) Implementujte prohlížečku Langweilova modelu Prahy a otestujte ji s uživateli.

Seznam doporučené literatury:

- 1] Jason Jerald. 2015. The VR Book: Human-Centered Design for Virtual Reality. Association for Computing Machinery and Morgan & Claypool, New York, NY, USA.
- 2] Steven M. LaValle - Virtual Reality, Cambridge University Press 2016
- 3] Kateřina Bečková, Miroslav Fokt. Svědectví Langweilova modelu Prahy. Schola Ludus Pragensia 1996.

Jméno a pracoviště vedoucí(ho) bakalářské práce:

**Ing. David Sedláček, Ph.D. katedra počítačové grafiky a interakce FEL**

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) bakalářské práce:

Datum zadání bakalářské práce: **09.02.2022** Termín odevzdání bakalářské práce: \_\_\_\_\_

Platnost zadání bakalářské práce: **30.09.2023**

Ing. David Sedláček, Ph.D.  
podpis vedoucí(ho) práce

podpis vedoucí(ho) ústavu/katedry

prof. Mgr. Petr Páta, Ph.D.  
podpis děkana(ky)

## III. PŘEVZETÍ ZADÁNÍ

Student bere na vědomí, že je povinen vypracovat bakalářskou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v bakalářské práci.

\_\_\_\_\_  
Datum převzetí zadání

\_\_\_\_\_  
Podpis studenta



## Poděkování / Prohlášení

Chtěl bych poděkovat Ing. Davidu Sedláčkovi, Ph.D. za vedení práce, ale také za užitečné rady, připomínky a celkově za příjemný přístup. Dále bych chtěl poděkovat Ing. Ladislavu Čmolíkovi, Ph.D. za radu ohledně tvorby textur v programu Blender. A také bych chtěl poděkovat Ing. Tomáši Havlíkovi za osvětlení situace ohledně vykreslování paprsku.

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

.....  
V Praze dne 20. 5. 2022

## Abstrakt / Abstract

Práce pojednává o tvorbě aplikace na prohlížení Langwailova modelu Prahy ve virtuální realitě. V rámci práce byly navrženy, implementovány a testovány různé způsoby prohlížení 3D modelů ve virtuální realitě. Jedním z důležitých bodů práce je nízká výpočetní náročnost, která je pro virtuálně realitní aplikace klíčová.

**Klíčová slova:** virtuální realita; VR; unity; steamvr; 3D modely; Langweil; Langweilův model Prahy.

This bachelor thesis describes the creation of a virtual reality application designed for viewing the Langweil's model of Prague. Various viewing techniques were designed, implemented and tested during the process of making. One of the important aspects of this thesis is low computational demands which is a key feature of a virtual reality application.

**Keywords:** virtual reality; VR; unity; steamvr; 3D models; Langweil; Langweil's model of Prague.

**Title translation:** VR 3D model viewer

# Obsah /

|   |    |  |    |
|---|----|--|----|
| <b>1 Úvod</b> .....   | 1  | 5.1 Průběh testování .....                 | 32 |
| <b>2 Analýza</b> .....  | 2  | 5.2 Dotazník .....                         | 33 |
| 2.1 Antonín Langweil a jeho model ..                                  | 2  | 5.3 Výsledky testování .....               | 34 |
| 2.2 Virtuální realita .....   | 2  | <b>6 Závěr</b> .....                       | 36 |
| 2.3 Analýza problematiky .....  | 3  | 6.1 Další vývoj .....                      | 36 |
| 2.4 Unity .....   | 4  | <b>Literatura</b> .....                    | 37 |
| 2.5 Unreal Engine .....   | 5  | <b>A Struktura příložených souborů</b> ... | 39 |
| 2.6 Analýza výchozích dat .....                                       | 5  | <b>B Návod na spuštění</b> .....           | 40 |
| 2.7 Blender .....   | 6  | <b>C Konkrétní podoba dotazníku</b> .....  | 42 |
| 2.8 Rešerše .....   | 6  | <b>D Odkazy</b> .....                      | 43 |
| 2.8.1 Sketchfab .....   | 6  | D.1 Gitlab repozitář práce .....           | 43 |
| 2.8.2 Strata inStudio VR .....  | 8  | D.2 Ukázkové video .....                   | 43 |
| 2.8.3 The VR Museum of<br>Fine Art .....                              | 9  | D.3 SteamVR balíček pro Unity ...          | 43 |
| 2.8.4 Infinite Art Museum .....                                       | 10 |  |    |
| <b>3 Návrh řešení</b> .....   | 12 |  |    |
| 3.1 Požadavky .....   | 12 |  |    |
| 3.2 Platforma .....   | 12 |  |    |
| 3.3 Prohlížečské metody .....   | 13 |  |    |
| 3.3.1 Prvotní návrh prohlí-<br>žečských metod .....                   | 13 |  |    |
| 3.3.2 Ovládání prvotních<br>metod prohlížení .....                    | 14 |  |    |
| 3.3.3 Testování prvotních<br>prohlížečských metod .....               | 15 |  |    |
| 3.3.4 Výsledky testování pr-<br>votních prohlížečských<br>metod ..... | 15 |  |    |
| 3.3.5 Nové metody prohlížení .  | 16 |  |    |
| 3.4 Návrh scény .....   | 16 |  |    |
| 3.5 Úprava dat .....  | 17 |  |    |
| 3.6 Způsob načítání dat .....   | 17 |  |    |
| <b>4 Implementace</b> .....   | 19 |  |    |
| 4.1 Způsoby prohlížení .....  | 19 |  |    |
| 4.2 Příprava modelu .....   | 20 |  |    |
| 4.2.1 Rozdělování modelu .....  | 20 |  |    |
| 4.2.2 Retexturování .....   | 21 |  |    |
| 4.2.3 Příprava materiálů na<br>vypečení textury .....                 | 23 |  |    |
| 4.2.4 Asset balíčky .....   | 25 |  |    |
| 4.3 Materiály v Unity .....   | 27 |  |    |
| 4.4 Struktura aplikace .....  | 28 |  |    |
| 4.5 Vykreslování paprsku ve Ste-<br>amVR .....                        | 29 |  |    |
| 4.6 Ovládání .....  | 30 |  |    |
| <b>5 Testování</b> .....  | 32 |  |    |

## Tabulky / Obrázky

|  |    |  |    |
|--|----|--|----|
| <b>4.1.</b> Porovnání načítacích časů . . . . .              | 27 | <b>1.1.</b> Vystavený Langweilův model . . . . . | 1  |
| <b>4.2.</b> Porovnání načítacích časů po<br>úpravě . . . . . | 27 | <b>2.1.</b> Náhlavní displej pro VR . . . . .    | 3  |
|  |    | <b>2.2.</b> Sketchfab . . . . .                  | 7  |
|  |    | <b>2.3.</b> SketchfabVR . . . . .                | 8  |
|  |    | <b>2.4.</b> Strata inStudio VR . . . . .         | 9  |
|  |    | <b>2.5.</b> The VR Museum of Fine Art . . . . .  | 10 |
|  |    | <b>2.6.</b> Infinite Art Museum . . . . .        | 11 |
|  |    | <b>3.1.</b> Prvotní scéna . . . . .              | 14 |
|  |    | <b>3.2.</b> Popis tlačítek na ovladači . . . . . | 15 |
|  |    | <b>4.1.</b> Manipulační způsob . . . . .         | 19 |
|  |    | <b>4.2.</b> Procházezí způsob . . . . .          | 20 |
|  |    | <b>4.3.</b> Rozdělený model . . . . .            | 21 |
|  |    | <b>4.4.</b> Porovnání rozlišení . . . . .        | 22 |
|  |    | <b>4.5.</b> Uzlový graf . . . . .                | 23 |
|  |    | <b>4.6.</b> Proces použití textury . . . . .     | 24 |
|  |    | <b>4.7.</b> Díl LW modelu . . . . .              | 25 |
|  |    | <b>4.8.</b> Díl LW modelu . . . . .              | 26 |
|  |    | <b>4.9.</b> Díl LW modelu . . . . .              | 26 |
|  |    | <b>4.10.</b> Vychozí materiál . . . . .          | 27 |
|  |    | <b>4.11.</b> Nelesklý materiál . . . . .         | 27 |
|  |    | <b>4.12.</b> Neosvětlený materiál . . . . .      | 28 |
|  |    | <b>4.13.</b> Vybírací scéna . . . . .            | 29 |
|  |    | <b>4.14.</b> Single pass vykreslování . . . . .  | 30 |
|  |    | <b>4.15.</b> Multi pass vykreslování . . . . .   | 30 |
|  |    | <b>5.1.</b> Oculus Quest 2 . . . . .             | 32 |
|  |    | <b>5.2.</b> Graf známkovacích odpovědí . . . . . | 34 |
|  |    | <b>5.3.</b> Znemožněná manipulace . . . . .      | 35 |

# Kapitola 1

## Úvod

V roce 1826 začal Antonín Langweil, tehdejší zaměstnanec Univerzitní knihovny v Klementinu, pracovat na detailním papírovém modelu historického centra Prahy [1]. Na modelu pracoval až do své smrti (1837). Nyní je model vystaven v Muzeu hlavního města Prahy – viz obrázek 1.1 níže. Model si návštěvníci muzea mohou aktuálně prohlédnout pouze za vitrínou. Langweilův model je poměrně velký, měří 7 metrů do délky a 5 metrů do šířky. Z tohoto důvodu si ho návštěvníci nemohou prohlédnout natolik zblízka, aby mohli obdivovat detaily, nad kterými pan Langweil strávil mnoho času. A tím se dostáváme k cíli této práce – umožnit návštěvníkům muzea prohlédnout si model zblízka, ze všech úhlů a možnost obdivovat tak i ty nejmenší detaily modelu. Pro prohlížení modelu bude použita virtuální realita, která umožní uživatelům (potažmo návštěvníkům muzea) prohlédnout si model zblízka a ze všech možných úhlů.



**Obrázek 1.1.** Vystavený Langweilův model v Muzeu hlavního města Prahy. Zdroj: [2]

# Kapitola 2

## Analýza

### 2.1 Antonín Langweil a jeho model

Antonín Langweil se narodil jako deváté dítě do rodiny sládka v Postoloprtech v roce 1791. Po přestěhování do Českého Krumlova a studiu na gymnáziu v Českých Budějovicích nastoupil v roce 1814 jako úředník na magistrát v Krumlově. O pár let později se vydal studovat Akademii výtvarných umění do Vídně, kde se věnoval studiu litografie. Po návratu z Vídně a přestěhování do Prahy si na Staroměstském náměstí v roce 1819 otevřel svou vlastní litografickou dílnu. Pár let nato však musel svou dílnu zavřít a nastoupil jako knihovní sluha do Univerzitní knihovny v Klementinu. V roce 1826 byl v pražském Platýzu vystaven reliéf Paříže od francouzského autora. Návštěva této výstavy Antonína inspirovala natolik, že se rozhodl, že začne pracovat na svém vlastním papírovém modelu Prahy. Na modelu pracoval až do konce svého života (1837).

Během práce na modelu Prahy byl rozpracovaný model mnohokrát vystaven pro veřejnost. Po smrti autora byl model v roce 1891 vystaven na Zemské jubilejní výstavě a také byl od roku 1905 součástí expozice Lapidária Národního muzea. V polovině 20. století byl model převzat Muzeem hlavního města Prahy a mezi lety 1963 a 1969 proběhla restaurace modelu. Po restauraci model zavítal na krátkou chvíli do nizozemského muzea Madurodam. Aktuálně se model nachází v muzeu hlavního města Prahy, kde je vystaven již od roku 1970 [1].

Langweilův model zachycuje více než dva tisíce budov z tehdejších oblastí Starého Města, Josefova, Malé Strany a Hradčan. Model je tvořen v měřítku 1:480 a obsahuje detailně nakreslené fasády domů, zahrady, parky či kostely. Model je velice unikátní, jelikož z původních dvou tisíc budov byla téměř polovina zbourána a mnoho budov bylo přestavěno. Z tohoto důvodu tak nabízí Langweilův model jedinečnou příležitost pozorovat původní podobu budov.

V roce 2006 se začalo pracovat na velkém projektu, totiž na digitalizaci celého Langweilova modelu. Na přenosu historického modelu do počítačové podoby se podílela holandská firma Visual Connection, a.s.<sup>1</sup>, spolu s odborníky na robotiku, počítačové vidění a grafiku z ČVUT a mnoha dalšími lidmi [3]. V průběhu 3D rekonstrukce bylo nafoceno téměř 300 tisíc fotografií ve vysokém rozlišení. Projekt byl dokončen v roce 2010 a Praha se tak v tu dobu stala jediným městem na světě, které má digitalizovaný historický model města takové velikosti[4]. Více informací o průběhu 3D rekonstrukce lze najít v[5].

### 2.2 Virtuální realita

Virtuální realita (dále jen VR) je technologie, která má za cíl přenést uživatele do počítačem vygenerovaného prostředí, se kterým může uživatel interagovat jako v reálném světě[6].

<sup>1</sup> <https://thevisualconnection.nl/>



VR zážitek či aplikace se prožívá za použití tzv. náhlavního displeje, většinou v kombinaci s ovladači (viz obrázek 2.1 níže), které uživatel drží v ruce a které používá na interakci s VR prostředím. Náhlavní displej zprostředkovává uživateli pohled do VR světa. Na tomto typu displeji lze nalézt dvě speciální čočky (každá pro jedno oko), které uživateli zkreslují obraz na displeji (který se nachází za čočkami) takovým způsobem, že umožňují uživateli stereoskopické vidění, tedy že dokáže vnímat hloubku obrazu jako v reálném světě. Ovladače mají několik tlačítek, která může uživatel využívat na různé interakce. Například: pohyb s daným objektem, střelba ze zbraně či malování štětcem. Aby byl zážitek z VR autentický, ovladače mají tzv. haptickou odezvu. Jedná se o to, že když například uživatel ve VR drží v ruce zbraň a vystřelí z ní, vibrační motory v ovladači na krátkou chvíli zavibrují, aby simulovaly zpětný ráz zbraně.

Aplikace pro VR se běžně tvoří v herním engine. Herní engine je softwarový framework, který se v aplikaci stará mj. o vykreslování scény, fyziku, animace či zvuky. V dnešní době je na světě mnoho herních engine. Dokonce bývalo zvykem, že každé velké herní vývojářské studio mívalo svůj vlastní herní engine, o který se s nikým nedělilo. Dnes už tomu tak ale není a více herních studií využívá stejný engine. Z těch známějších můžeme jmenovat například Unreal Engine, CryEngine, Unity či Godot. Avšak ne všechny herní engine mají podporu pro vývoj VR her či aplikací. Ze čtyř již jmenovaných herních engine, se všechny řadí mezi ty, ve kterých lze vytvořit aplikaci ve VR, nicméně těmi nejpoblárnějšími pro tento účel jsou Unity a Unreal Engine.



**Obrázek 2.1.** Náhlavní displej Acer Windows Mixed Reality AH101 s ovladači. Zdroj: [7]

## 2.3 Analýza problematiky

Cílem této práce bude vytvoření VR aplikace umožňující prohlížení kompletního digitalizovaného Langweilova modelu (dále jen LW model), za využití náhlavního displeje a ovladačů. Vzhledem k faktu, že se bude jednat o aplikaci pro VR, bude nezbytně nutné zajistit nízkou výpočetní náročnost při práci s LW modelem. V opačném případě by se totiž mohlo uživatelům při neplynulém prohlížení dělat nevolno. Za tímto účelem budou v práci použity v podstatě dva LW modely – jeden v nízkém rozlišení a druhý v rozlišení vysokém.

V aplikaci bude umožněná uživatelům mj. manipulace s LW modelem za účelem prohlížení ze všech stran a úhlů. Tedy navržení efektivních prohlížečích způsobů, které budou mít intuitivní ovládání, bude dalším cílem.

Jelikož se v aplikaci nikdy nebude v jeden moment nacházet celý LW model ve vysokém rozlišení, bude potřeba zajistit rychlý a spolehlivý způsob načítání LW modelu za běhu aplikace. Tento způsob načítání bude použit pro menší části LW modelu. Také bude zapotřebí menší části LW modelu vytvořit a připravit na načítání.

## 2.4 Unity

Unity je multiplatformní herní engine, který podporuje tvorbu 2D, 3D, VR a AR<sup>1</sup> her či aplikací. Tento herní engine byl poprvé oznámen a vydán v červnu 2005 na Apple Worldwide Developers Konferenci<sup>2</sup> a od té doby nabral na popularitě, zejména díky poměrně jednoduchému používání a také díky možnostem vývoje mobilních her na platformy Android a iOS[8]. Jedna z největších výhod Unity je v jeho komunitě, která se každým rokem rozrůstá. Psaní skriptů pro tento herní engine je realizováno v programovacím jazyce C#. Unity má k dispozici svůj vlastní obchod - Unity Asset Store<sup>3</sup>, odkud si vývojáři mohou do svých projektů přidávat nejrůznější balíčky assetů<sup>4</sup>, které posouvají jejich možnosti implementace. Na Unity Asset Store lze najít například balíčky různých modelů a skriptů či celé knihovny.

Unity také umožňuje tzv. „zabalení“ libovolného assetu do balíčku (Asset Bundle) pro následné použití. Do balíčku mohou být velice prakticky zabaleny přímo celé objekty, jejichž součástí už jsou například skripty či je objekt již konkrétně nastavený. Vytvořený balíček může být velice jednoduše načten do aplikace i za jejího běhu.

Unity podporuje celkem čtyři vykreslovací řetězce (render pipeline). Vykreslovací řetězec je sekvence operací, která má za úkol vykreslit objekty na obrazovku monitoru (obecně cílového zařízení). Těmi čtyřmi jsou:

- Built-In Render Pipeline – základní vykreslovací řetězec, který podporuje všechny platformy, nicméně umožňuje jen omezené úpravy.
- Universal Render Pipeline – vykreslovací řetězec se zaměřením na vysoký výkon. Tento řetězec podporuje širokou škálu zařízení a umožňuje více úprav než základní vykreslovací řetězec. Nabízí však omezenou podporu balíčků a assetů z Unity Asset Store.
- High-Definition Render Pipeline – vykreslovací řetězec, který se zaměřuje na vysoké rozlišení a používá se u nejnáročnějších projektů. Také podporuje více úprav než základní řetězec a také má omezenou podporu ze strany Unity Asset Store.
- Custom Scriptable Render Pipeline – vykreslovací řetězec, který umožňuje kompletně upravit či předělat vykreslovací řetězec pro speciální potřeby vývojáře, nicméně opět s omezenou podporou rozšiřovacích balíčků či assetů z Unity Asset Store.

<sup>1</sup> AR - Augmented reality (= rozšířená realita)

<sup>2</sup> <https://developer.apple.com/wwdc22/> (ročník 2022)

<sup>3</sup> <https://assetstore.unity.com/>

<sup>4</sup> Za asset se považuje cokoliv vytvořeného vývojářem - např.: model, textura, celá scéna nebo také napsaný skript

## 2.5 Unreal Engine

Unreal Engine je herní engine od společnosti Epic Games. Poprvé byl použit již v roce 1998 při tvorbě hry Unreal. Přestože byl tento herní engine původně zamýšlen pouze na tvorbu her z žánru střílečky z první osoby, byl později použit i na hry jiných žánrů. Tento herní engine je psán v programovacím jazyce C++. Unreal Engine je oproti Unity open-source, tedy každý uživatel má přístup do zdrojových kódů a každý může navrhovat změny funkčnosti. V porovnání s Unity je považován za složitější herní engine a na Unity také ztrácí i v počtu stažitelných assetů – Unity má položek ke stažení zhruba třikrát více[9]. Naopak grafické rozlišení finálního produktu je lepší v Unreal Engine, proto je mnoho velkých her vytvářeno právě v tomto herním engine. Do nedávna byla nejnovější verze Unreal Engine 4, ale v dubnu 2022 byla vydána nová verze – Unreal Engine 5.

## 2.6 Analýza výchozích dat

Struktura dat, ve kterých mi byl zapůjčen LW model, byla následující. Data obsahovala šest 3D objektů (ve formátu `.obj`) reprezentujících celý LW model a šest souborů definujících materiály modelu (ve formátu `.mtl`). Dále byly součástí dat textury<sup>1</sup>, a to rovnou dvě sady – jedna sada ve vysokém rozlišení a druhá sada v nízkém rozlišení.

Obsah a struktura dat v `.obj` souborech byla ve výchozím stavu následující: `.obj` soubory obecně popisují geometrii modelu, tedy obsahují informace o jednotlivých vrcholech a plochách modelu. Vrcholy jsou definovány třemi čísly, které určují souřadnice vrcholu na osách X, Y a Z. V počítačové grafice se nejčastěji jako plochy používají trojúhelníky – mj. z důvodu zaručené konvexity plochy. I v tomto případě jsou plochy modelu trojúhelníky a jsou definovány pomocí indexů vrcholů. Kromě toho jsou součástí také názvy materiálů a texturovací souřadnice, pokud model využívá textury (v tomto případě je model využívá). Texturovací souřadnice slouží k identifikaci místa v textuře, které se má nanést na daný vrchol (více o texturovacích souřadnicích v kapitole 4.2.3).

Textury byly ve výchozí struktuře organizovány do mnoha složek. Každý `.obj` soubor měl ve výchozích datech svůj příslušný `.mtl` soubor, ve kterém byly definovány materiály. V definici materiálu se nacházela informace o tom, která textura, z jaké složky se má pro daný materiál použít.

Na ukázce níže lze vidět jak část `.obj` souboru, tak část příslušného `.mtl` souboru, spolu s komentáři jednotlivých řádků.

Ukázka z `.obj` souboru:

```
mtllib o00.mtl           definice příslušného .mtl souboru
g _01_105210_9_MG       definice skupiny vrcholů
v 1120.52399 -65.47468 -315.32889      pozice vrcholů
v 1113.9593 -65.9279 -329.74936
...
vt 0.535193 0.318952     texturovací souřadnice
vt 0.901399 0.929382
...
```

<sup>1</sup> Textura je většinou barevný 2D obrázek, který se nanese na povrch modelu a přidává tak modelu jemné detaily.

```

usemtl _01_105210_9_MG                volba materiálu
f 1/2 2/42 4/43                        definice trojúhelníků pomocí indexů
f 4/43 2/42 3/44                        index vrcholu/index texturovací souřadnice
...

```

Ukázka z .mtl souboru:

```

newmtl _01_105210_9_MG                definice materiálu
Ka 0.117647 0.117647 0.117647        definice vlastností materiálu
Kd 0.752941 0.752941 0.752941
Ks 0.752941 0.752941 0.752941
map_Kd 01_maps/_01_105210_9_MG.png    adresa textury
...

```

## 2.7 Blender

Blender je freeware a zároveň open-source program pro práci s 3D modely, také je považován za jeden z nejlepších zdarma dostupných programů pro práci s 3D modely. První verze programu byla vydána již v roce 1994. Od té doby se software změnil a nabral na popularitě. V Blenderu je možné modely jak editovat, tak i vytvářet. Kromě toho však nabízí i další funkce, které se při práci s 3D modely často využívají – například komplexní tvorbu materiálů či textur, UV mapování, animace, práci se světlem, tvorbu simulací, vytváření renderů, a další. Blender API<sup>1</sup>, které podporuje programovací jazyk Python, nabízí možnost tvorby vlastních funkcí, či schopnost ovládat program z prostředí kódu. Blender je také multiplatformní, tedy funguje na zařízeních s operačním systémem Windows, Linux, ale i macOS[10]. Komunita uživatelů tohoto programu je velká, tedy není problém najít nejrůznější návody.

## 2.8 Rešerše

V této kapitole bych se rád věnoval analýze VR aplikací na prohlížení 3D modelů a aplikací, které zprostředkovávají návštěvu VR muzea. Součástí bude také analýza způsobu samotného prohlížení a pohybu ve VR.

VR aplikací na prohlížení 3D modelů či VR muzeí lze v dnešní době nalézt poměrně mnoho, nicméně každá se trochu liší a každá většinou využívá lehce jiný způsob prohlížení či pohybu. Pro svou analýzu jsem si vybral aplikaci na prohlížení 3D modelů „Sketchfab“, aplikaci „Strata inStudio VR“, která se věnuje umístování 3D modelů do scény a dále bych se rád v této kapitole věnoval VR muzeím, konkrétně aplikaci „The VR Museum of Fine Art“ a „Infinite Art Museum“. Všechny čtyři aplikace lze stáhnout zdarma z obchodní platformy Steam<sup>2</sup>. Aplikace byly testovány na náhlavním displeji Acer Windows Mixed Reality AH101 s ovladači (viz obrázek 2.1).

### 2.8.1 Sketchfab

Aplikace Sketchfab<sup>3</sup> nabízí možnost prohlížení 3D modelu ať už ve webovém prohlížeči jako takovém, nebo ve VR či AR. Ve Sketchfabu je také možné si kromě samotného prohlížení 3D modelu, model koupit či nabídnout model k prodeji. Na webu aplikace

<sup>1</sup> API - Application Programming Interface (rozhraní pro programování)

<sup>2</sup> <https://store.steampowered.com/>

<sup>3</sup> <https://sketchfab.com/>

lze nalézt širokou škálu nejrůznějších modelů – například modely zvířat, strojů, zbraní, bot či dokonce historických artefaktů. Prohlížení modelů ve VR lze realizovat dvojím způsobem. Jednak je možné si spustit prohlížení ve VR přímo z webového prohlížeče bez nutnosti stahování aplikace nebo si lze aplikaci Sketchfab VR stáhnout z platformy Steam<sup>1</sup>. Obě možnosti nabízejí jiný způsob prohlížení. Nejprve bude popsáno prohlížení z prostředí webového prohlížeče a následně bude způsob prohlížení a celá aplikace porovnána s její stažitelnou verzí.

Po přepnutí do režimu pro VR ve webové verzi aplikace se uživatel objeví ve scéně vedle zvoleného modelu. Ve scéně se lze pohybovat pomocí teleportace, která je realizována namířením ovladače na místo na zemi a následným stisknutím tlačítka trigger na ovladači. V okamžiku teleportace uživateli lehce problikne obrazovka, což působí velmi příjemně a připomíná přirozené mrknutí člověka. Tímto způsobem si může uživatel model libovolně obcházet ze všech stran. Kromě pohybu ve scéně je možné si prohlížený model zvětšit či naopak zmenšit. Tato interakce je realizována pomocí natažení joysticku na ovladači doleva a doprava. Po změně velikosti si může uživatel lépe prohlédnout model. Nicméně změna velikosti modelu je prováděna příliš zběsile, model se zvětšuje/zmenšuje velice rychle. Tento fakt mi při testování aplikace způsoboval nemalou nevolnost a zhoršení celkového názoru na tuto aplikaci. Kromě změny velikosti a pohybu ve scéně webová verze aplikace Sketchfab ve VR nenabízí nic jiného. Prohlížení historických artefaktů probíhá totožně, jako prohlížení jiného libovolného modelu.



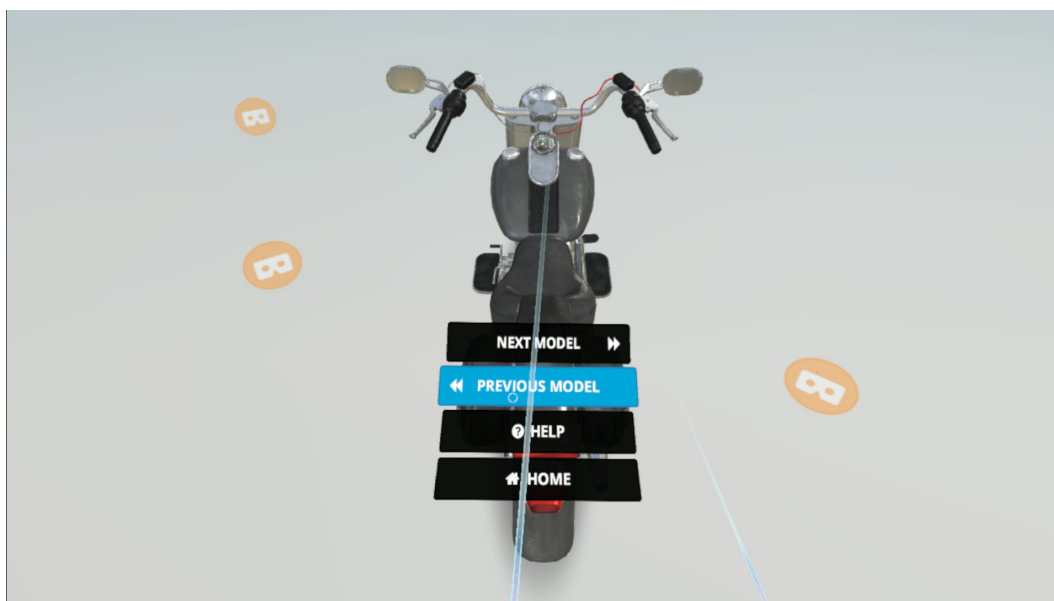
**Obrázek 2.2.** Scéna ve webové verzi Sketchfabu

Stažená verze aplikace Sktechfab, s názvem Sketchfab VR, funguje lehce odlišně. Jak již název napovídá, aplikace nabízí prohlížení pouze ve VR. Po spuštění aplikace se uživatel objeví v první scéně, která slouží jako menu. V menu si uživatel může vybrat jednu ze sedmi prohlídek. Prohlídka začíná přesunutím do scény s prvním modelem. Scéna je v tomto smyslu podobná jako u verze pro webový prohlížeč, avšak prohlížení je realizováno jiným způsobem. Ve scéně se není možné libovolně pohybovat, je pouze definováno pár míst, odkud si může uživatel model prohlížet. Přepínání mezi těmito místy je realizováno stisknutím tlačítka trigger na ovladači. Po vyčerpání všech pozic na prohlížení je uživatel přesunut do další scény z vybrané prohlídky. Po skončení prohlídky

<sup>1</sup> [https://store.steampowered.com/app/464940/Sketchfab\\_VR/](https://store.steampowered.com/app/464940/Sketchfab_VR/)

je uživatel navrácen do původní scény s menu. Jakákoliv manipulace s modelem není možná, stejně jako ve verzi ve webovém prohlížeči. V této verzi aplikace nebylo ani možné změnit velikost modelu. Nicméně pár míst, ze kterých bylo umožněno model pozorovat, se nacházelo ve vzduchu, tedy pohled na model seshora byl u některých modelů možný.

Nepříjemnost, na kterou jsem v průběhu prohlížení narazil, byla spojená s objevujícím se menu při prohlídce některých modelů. Když jste jako uživatel ve scéně s modelem a prohlížíte si jej, je možné si zobrazit menší menu, a to tak, že se jednoduše podíváte směrem k zemi. Nápad na zobrazování menu se mi ze začátku líbil, nicméně vícekrát se mi stalo, že se menu zobrazovalo, když jsem si jen chtěl prohlédnout část modelu, která byla blíže zemi, a zobrazené menu mi tak model zakrylo (viz obrázek 2.3).



**Obrázek 2.3.** Menu překrývající prohlížený model ve Sketchfab VR

### 2.8.2 Strata inStudio VR

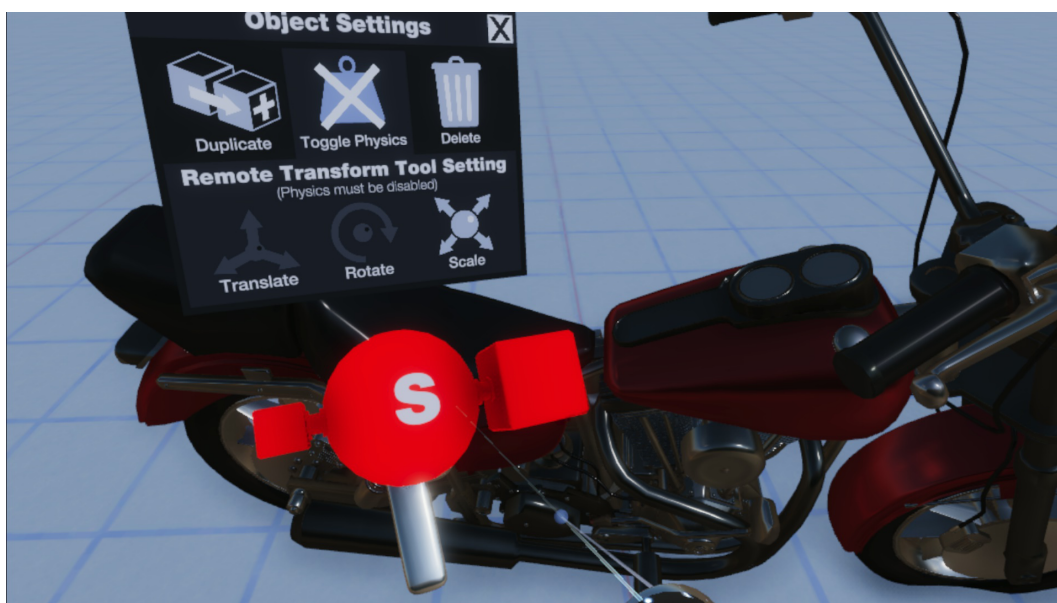
Další testovaná aplikace slouží jako „VR hřiště pro 3D modely“<sup>1</sup>. V této aplikaci si lze do scény vkládat různé 3D modely. Aplikace nabízí několik již připravených modelů, ale také má možnost pro vložení vlastních 3D modelů. Po spuštění aplikace se objeví uživatel v menu, kde si může nastavit prostředí scény a další parametry. Po přesunu do prázdné scény si lze otevřít malé menu, ve kterém je možné si vybrat modely pro vložení do scény. Přestože prohlížení 3D modelů není primárním cílem této aplikace, je to v aplikaci možné a prohlížeč způsob se nezdá vůbec špatným. Po vložení modelu do scény lze s modelem různě manipulovat. A to tak, že ve scéně lze vypnout vypočítávání fyziky (tedy že model například zůstane ve vzduchu a nepadne na zem) a poté je možná změna vlastností modelu. V aplikaci lze zmenšovat a zvětšovat daný model, ale také je možné model nejrůzněji otáčet a libovolně s ním ve scéně pohybovat. Pohyb uživatele ve scéně je zajištěn opět teleportací.

Změna velikosti modelu je provedena tak, že si uživatel nejdříve vybere, že chce upravovat velikost modelu, a poté se mu na levé ruce (ovladači) objeví interakční kulička (viz obrázek 2.4). Druhou rukou je možné kuličku uchopit a táhnout s ní doprava/doleva pro zvětšení/zmenšení. Změna rotace je implementována podobným způsobem jako

<sup>1</sup> [https://store.steampowered.com/app/627530/Strata\\_inStudio\\_VR/](https://store.steampowered.com/app/627530/Strata_inStudio_VR/)



změna velikosti. Uživatel nejdříve zvolí, že chce měnit orientaci objektu, a následně se mu opět na levé ruce (ovladači) objeví interakční kulička. Tentokrát pohyb s kuličkou nijak nemění orientaci modelu, nicméně interakční technika pro změnu orientace je podobná. Tedy uživatel pravou rukou uchopí kuličku a následným otáčením ruky udává modelu rotaci. Uchopení kuličky jak pro změnu velikosti, tak pro změnu orientace je provedeno stisknutím a podržením tlačítka trigger na ovladači. Tento způsob prohlížení funguje poměrně efektivně, nicméně vypnutí výpočtu fyziky a následná volba změny velikosti/orientace mi přišla příliš zdlouhavá a lehce nepohodlná. Změna orientace s technikou otáčení interakční kuličky se neovládala nejlépe. Občas bylo složité dosáhnout požadované orientace, nicméně i tak pohyb spolu s manipulací s modelem tvoří poměrně dobrý způsob prohlížení.



**Obrázek 2.4.** Interakční kulička při prohlížení ve Strata inStudio VR

### ■ 2.8.3 The VR Museum of Fine Art

Obecný koncept VR muzea je sám o sobě velice povedený. Uživatel si tímto způsobem může prohlížet nejrůznější exponáty a není přitom omezován vitrínami či pracovníky ochranné služby. Zároveň přesun z výstavy historických letadel do přehlídky egyptských pyramid udělá uživatel takřka lusknutím prstu. Aplikace The VR Museum of Fine Art<sup>1</sup> přesně toto nabízí. Po startu aplikace se uživatel objeví u recepce muzea, kde si může vzít do ruky mapu a přesunout se k prvním exponátům. VR muzeum nabízí například expozici sochy Davida od světoznámého Michelangela, sošky čínských vojáků z druhého století př. n. l., či expozici velkého Buddhy z Kamakury.

Ve scéně se opět může uživatel pohybovat teleportací, která mu dává možnost prohlédnout si exponáty ze všech stran. Zážitek v muzeu obohacují popisky, které se objeví ve chvíli, kdy se uživatel k danému předmětu přiblíží. Historické artefakty tu jsou prezentovány na různých podstavcích, které může uživatel obcházet. Prezentace všech exponátů je tu ovšem statická, tedy není možné s předměty nijak manipulovat.

<sup>1</sup> [https://store.steampowered.com/app/515020/The\\_VR\\_Museum\\_of\\_Fine\\_Art/](https://store.steampowered.com/app/515020/The_VR_Museum_of_Fine_Art/)



**Obrázek 2.5.** Ukázka scény z aplikace The VR Museum of Fine Art. Lze vidět mapu a v pozadí první exponáty.

#### ■ 2.8.4 Infinite Art Museum

Druhým navštíveným VR muzeem je Infinite Art Museum<sup>1</sup>, které nabízí prohlídku obrazů. Po spuštění aplikace se uživatel objeví ve vysoké budově s mnoha patry. V každém patře lze nalézt spoustu obrazů od různých autorů. Aktuálně se v aplikaci nachází osm pater a muzeum nabízí k prohlížení celkem přes osm set obrazů od více než 70 autorů. Aplikace také nabízí možnost přidat do výstavy vlastní obrazy. Pohyb ve scéně je realizován opět pomocí teleportace, nicméně v této aplikaci je pohyb doplněn o „pomalé chození“. Toho uživatel dosáhne stisknutím a podržením tlačítka grip na ovladači a následným pohybem ruky do směru, ve kterém se chce uživatel pohnout. Tento způsob pohybu se mi při testování zdál poněkud divný, nicméně pro nějaké menší pohyby by využití našel.

Ve výstavě jsou obrazy prezentovány přirozeně pověšené na zdech podél celé místnosti. Lze nalézt obrazy malé, ale i velké. Způsob prohlížení zde není nijak speciální, avšak pro prohlížení obrazů žádný speciální způsob prohlížení není třeba. Uživatel si může obrazy prohlížet pouze tak, že se přiblíží pohybem (teleportací), a poté může obdivovat detaily obrazu.

<sup>1</sup> [https://store.steampowered.com/app/1011000/Infinite\\_Art\\_Museum/](https://store.steampowered.com/app/1011000/Infinite_Art_Museum/)





**Obrázek 2.6.** Ukázka scény z aplikace Infinite Art Museum

# Kapitola 3

## Návrh řešení

V následujících kapitolách bude popsán design aplikace.

### 3.1 Požadavky

Aplikace, která bude vytvořena v rámci této práce, by mohla být použita v Muzeu hlavního města Prahy vedle exponátu fyzického LW modelu. Do muzea chodí široké spektrum návštěvníků a je tedy potřeba počítat s tím, že většina z nich nebude mít zkušenosti s VR. Z tohoto důvodu by aplikace měla mít intuitivní a jednoduché ovládání. Dále by aplikace měla být pro návštěvníky muzea přínosná, tedy aby neukazovala jen to, co si mohou prohlédnout na vystaveném fyzickém modelu.

Jelikož se jedná o aplikaci pro VR, je nezbytně nutné, aby aplikace dosahovala nízké výpočetní náročnosti. Tedy aby byl obraz v náhlavním displeji dostatečně plynulý. Také je důležité, aby prohlížecké způsoby byly pro člověka příjemné. V opačném případě by se člověku mohlo dělat nevolno z neplynulého obrazu, nebo by například mohl mít závratě z nepříjemného způsobu prohlížení (podobně jako u prohlížeckého způsobu u webové verze Sketchfabu - viz kapitola 2.8.1).

Data LW modelu ve vysokém rozlišení jsou paměťově náročná, nicméně musí být zajištěno, aby jejich prohlížení výpočetně náročné nebylo. Bude tedy zapotřebí zajistit, aby prohlížení probíhalo tak, že si uživatel bude vždy prohlížet jen malou část z celého LW modelu. Načtení této malé části proběhne až za běhu aplikace. Dalším stanoveným požadavkem bylo, aby se všechna data, která obsahují LW model a textury pro něj, nenacházela přímo v aplikaci, ale aby je aplikace byla schopná použít.

### 3.2 Platforma

Aplikace na prohlížení LW modelu Prahy bude vytvořena v herním enginu Unity pro Windows. Volba herního enginu pro vývoj aplikace byla jednoduchá. Důvodem byla dřívější zkušenost pouze s jedním herním enginem, totiž s Unity.

Jak již bylo zmíněno v kapitole 2.4, tak Unity nabízí na výběr celkem ze čtyř vykreslovacích řetězců. Používání základního vestavěného řetězce v dnešní době již není časté (dalo by se říct, že je již zastaralý), protože novější řetězce nabízejí lepší možnosti využití. V dnešní době se standardem pro běžné aplikace stává řetězec URP<sup>1</sup>, se kterým se jednoduše pracuje a není tak náročný jako řetězec HDRP<sup>2</sup>. HDRP se používá u aplikací a her, kde je cílem dosáhnout nejvyšší grafické kvality, čímž také dosahuje vysoké výpočetní náročnosti. V rámci této práce bude tedy použit vykreslovací řetězec URP.

Vývoj VR aplikace v samotném herním enginu Unity by byl složitý, proto bude při vývoji použit tzv. SDK<sup>3</sup> pro práci s VR. Použití SDK na vývoj VR aplikací pomáhá

<sup>1</sup> URP - Universal Render Pipeline

<sup>2</sup> HDRP - High-Definition Render Pipeline

<sup>3</sup> SDK - Software Development Kit

vývojáři například s tím, že se vývojář nemusí starat o základní věci jako je propojení kamery s náhlavním displejem či otáčení kamery poté, co uživatel otočí hlavou. I přesto, že je VR pro herní enginy stále poměrně nová, tak i v dnešní době lze nalézt poměrně mnoho SDK pro vývoj VR aplikace s podporou pro Unity. Z těch známějších můžeme jmenovat například Oculus Interaction SDK, SteamVR či OpenXR. Volba SDK pro vývoj aplikace částečně určí, na jakém zařízení (tedy náhlavním displeji) bude možné aplikaci používat. Protože některé SDK podporují jen a pouze jeden náhlavní displej. A tím, že aplikace by mohla být použita v Muzeu hlavního města Prahy na zatím neznámém náhlavním displeji, bude zapotřebí zvolit takový SDK, který podporuje širokou škálu náhlavních displejů.

Oculus Interaction SDK od společnosti Meta<sup>1</sup>, je balíček sloužící na vývoj VR aplikací pouze pro platformu Oculus. SteamVR, což je SDK vyvíjeno společností Valve<sup>2</sup>, podporuje velké množství náhlavních displejů – například Oculus, Windows Mixed Reality, HTC a další. Díky této podpoře bude možné používat aplikaci na různých zařízeních bez nutnosti provádění změn v kódu aplikace. Poslední zmíněný – OpenXR, je nový SDK od společnosti Khronos Group<sup>3</sup>, který byl vyvinut za účelem sjednocení VR/AR vývoje[11]. Tento balíček se v budoucnu pravděpodobně stane standardem pro vývoj v oblasti VR/AR, nicméně v této době OpenXR ještě nemá plnou podporu na některých zařízeních. Vzhledem k tomu, že SteamVR podporuje mnoho zařízení a také již funguje spolu s OpenXR, bude na vývoj aplikace pro prohlížení LW modelu použit právě SDK SteamVR.

SteamVR dokáže mimo jiné detekovat, jaké zařízení je právě používáno a podle toho správně nastavit například ovládání aplikace. Tedy aby tlačítka na ovladačích fungovala tak, jak bylo zamýšleno, i na jiném náhlavním displeji (respektive ovladači), než byla aplikace vyvíjena (za předpokladu předchozího nastavení daného displeje v rámci používaného Steam účtu).

Náhlavní displej, který bude používán při vývoji aplikace, je Acer Windows Mixed Reality AH101 (viz obrázek 2.1), který mi byl zapůjčen z katedry počítačové grafiky a interakce. Nicméně jak již bylo zmíněno, díky využití SteamVR, nebude výsledná aplikace na prohlížení LW modelu Prahy limitována pouze na tento náhlavní displej.

## 3.3 Prohlížečské metody

V této kapitole bude rozepsán průběh návrhu prohlížečských metod. Návrh probíhal ve dvou fázích. V první fázi byly navrženy určité metody prohlížení, které byly následně otestovány (tyto metody nejsou v práci implementovány). Poté byla zpracována zpětná vazba a prohlížečské způsoby byly upraveny.

### 3.3.1 Prvotní návrh prohlížečských metod

Prvotní návrh probíhal s omezenou sadou dat, konkrétně byly k dispozici pouze tři díly (z celkových 52) ve vysokém rozlišení. Byla vytvořena scéna, do které byly umístěny všechny tři díly – scénu lze vidět na obrázku 3.1. Pro každý díl ve scéně byl navržen a implementován jiný způsob prohlížení. Všechny tři typy prohlížení však byly založeny na pohybu (teleportaci) ve virtuálním světě.

U prvního dílu (na obrázku 3.1 vlevo) mohl uživatel s modelem pohybovat, a to nahoru a dolů, aby si jej mohl prohlédnout dle potřeby z různých úhlů a výšek. Kromě

<sup>1</sup> <https://about.facebook.com/>

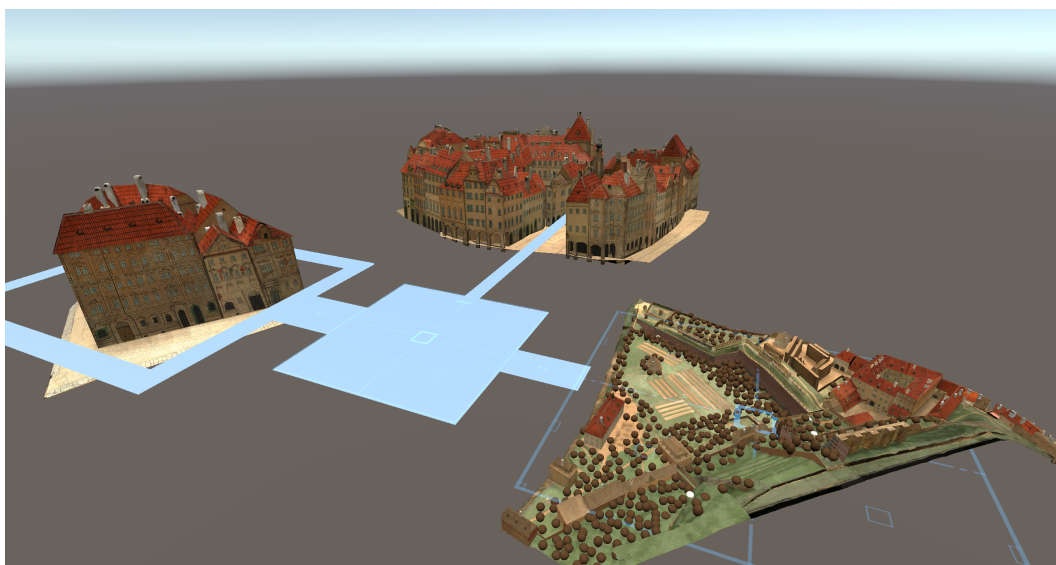
<sup>2</sup> <https://www.valvesoftware.com/>

<sup>3</sup> <https://www.khronos.org/>

pohybu s modelem bylo možné prohlédnout si první díl i ze všech stran a to tak, že se uživatel mohl „procházet“ (= teleportovat se) po pomyslné cestičce okolo modelu.

Druhý díl (na obrázku 3.1 uprostřed), ve kterém kromě domů nalezneme i ulici, nabídl uživateli prohlížení dvojitým způsobem. Uživatel se mohl procházet po virtuálním chodníku, který vede ulicí, a rozhlížet se jak na obě strany, tak bylo možné dívat se pod sebe, kde mohl uživatel vidět dlážděnou cestu. K procházce byl připojený ještě druhý způsob prohlížení, a to prohlídka zastavěné části modelu, která je ze všech stran obklopena domy. Na tuto prohlídku se uživatel dostal pomocí stisknutí daného tlačítka na ovladači, které ho teleportovalo přímo do modelu, kde si mohl prohlédnout menší prostranství s průchodem či balkóny bytů. Celkem bylo prohlídek uvnitř modelu pět a u každé si uživatel mohl povšimnout, že i v částech modelu, které jsou takto obklopené domy, autor nešetřil detaily, či barvou.

Při prohlížení třetího dílu ve scéně (na obrázku 3.1 vpravo) se uživatel vžil do role obra, který mohl kráčet po modelu Prahy. Prohlížení tohoto dílu bylo velice jednoduché – uživateli bylo dovoleno teleportovat se po virtuální ploše, která se nacházela přímo nad modelem, a tedy si mohl vše prohlížet z větší výšky.



**Obrázek 3.1.** Scéna prvotního návrhu

### 3.3.2 Ovládání prvotních metod prohlížení

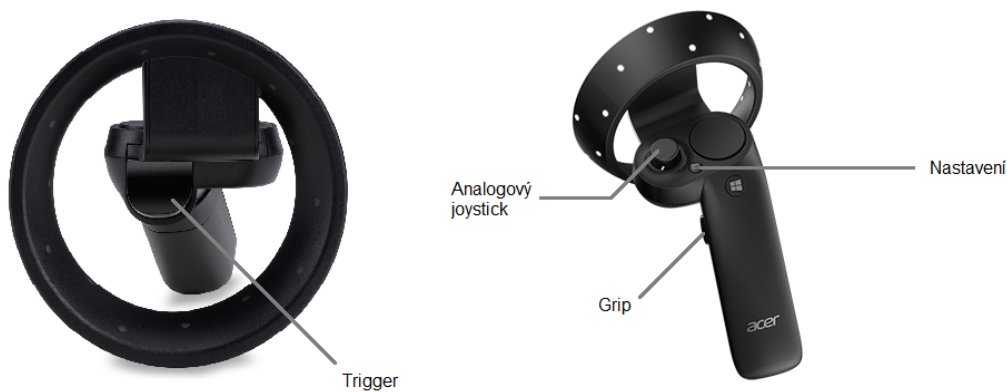
Jak již bylo zmíněno, hlavní pohyb ve VR scéně byl zajištěn pomocí teleportace, která byla implementována pomocí analogového joysticku – jakmile uživatel naklonil páčku dopředu (resp. od sebe), tak se objevila křivka a zelený kruh znázorňující místo, kam se uživatel teleportuje po puštění páčky.

S joystickem je také spojené otáčení kolem vlastní osy. Když uživatel natáhne páčku doleva/doprava a ihned pustí, tak se kamera otočí o 30° doleva/doprava. Uživatel se kolem své osy také otáčí přirozeným otáčením hlavy v reálném světě.

Pohyb s prvním dílem byl realizován dvěma tlačítky – „triggerem“ a „gripem“. Když uživatel stiskne trigger, model se začne pohybovat nahoru, podobně stisknutí gripu pohybovalo s modelem dolů.

U druhého dílu byla kromě teleportace po ploše také možnost teleportace přímo do modelu. Tohoto přesunu do modelu uživatel dosáhl stisknutím tlačítka pro nastavení.

Tlačítka na ovladači jsou popsána na obrázku 3.2.



**Obrázek 3.2.** Popsaná tlačítka ovladače od náhlavního displeje Acer Windows Mixed Reality AH101. Zdroj: [7] (upraveno)

### ■ 3.3.3 Testování prvotních prohlížečských metod

Jelikož je zamýšleno, že konečná aplikace by mohla být použita v Muzeu hlavního města Prahy, bylo za účelem smysluplného otestování zapotřebí vybrat pro testování zástupce cílové skupiny uživatelů. Takovými uživateli, tedy potenciálními návštěvníky muzea, jsou například žáci základní či střední školy anebo rodiny s dětmi v tomto věku.

Prvotního testování se zúčastnila rodina se třemi dětmi – ve věku 9, 13 a 14 let. Testování se zúčastnily pouze děti, rodičům se do testování moc nechtělo, raději jen vše pozorovali z dálky.

Pro testování byl použit počítač s těmito specifikacemi:

- CPU: AMD Ryzen 5 3600
- GPU: NVIDIA GeForce GTX 1660 SUPER
- RAM: 16 GB, 3600 Hz
- Náhlavní displej: Acer Windows Mixed Reality AH101

Testování probíhalo následovně. Uživatelům bylo nejdříve v krátkosti sděleno, o jakou aplikaci se jedná a co se od nich čeká. Následně si nasadili náhlavní displej a začali s prohlížením. Po nasazení displeje bylo uživatelům vysvětleno, jak se mohou ve scéně pohybovat. Po krátkém osvojení pohybu byli instruováni k prohlížení jednotlivých dílů. U každého dílu bylo vysvětleno, jakým způsobem si lze díl prohlížet. Po prohlédnutí všech dílů byli uživatelé dotázáni, který způsob prohlížení se líbil nejvíce a také který se líbil nejméně.

### ■ 3.3.4 Výsledky testování prvotních prohlížečských metod

Testování probíhalo ve scéně, kterou lze vidět na obrázku 3.1. Ve scéně byly implementovány prohlížečské způsoby popsané v kapitole 3.3.1. Na začátek je důležité zmínit, že žádné z dětí, které testovaly aplikaci, nemělo dřívější zkušenosti s VR.

Testování ukázalo, že ne všechny prohlížečské způsoby jsou dobré, respektive ukázalo, které jsou lepší než jiné. Nejoblíbenějším způsobem se stal třetí prohlížečský způsob, tedy pohyb přímo po modelu (resp. nad modelem), nicméně se dva ze tří uživatelů shodli, že tento prohlížečský způsob nelze použít na části modelu s budovami. Třetí uživatel, nejmladší ze všech tří, byl názoru jiného (nicméně vzhledem k průběhu testování s nejmladším členem rodiny si nejsem jistý, zda byla zpětná vazba z jeho strany relevantní – nejmladšího uživatele nejvíce oslovila celá VR jako taková, spíše než prohlížení

modelů). Prohlížečím způsobem s nejnižší oblíbeností se stal způsob první, přestože se uživatelům pohybování s modelem líbilo, ostatní prohlížečící způsoby je oslovily více. U druhého dílu LW modelu se uživatelům líbilo procházení ulic, ale postrádali pohled na model z vnějšku. U procházení ulic se objevily dva názory týkající se virtuálního chodníku. První názor tvrdil, že by bylo dobré, kdyby byl chodník průhledný, aby uživatel viděl i přímo pod sebe, například na dlážděnou cestu. Druhý názor byl opačný a neprůhledný chodník zastával z pocitu jistoty pevné země pod nohama. Tvrdil, že z chození po průhledném chodníku by se uživateli mohlo dělat nevolno.

Problém, který se poměrně často vyskytoval, byl s otáčením. Otáčení uživatele ve scéně je ovládáno natažením páčky doleva či doprava a následným puštěním. Problém spočíval v tom, že když se uživatelé chtěli otočit kolem své osy, tak kromě toho, že páčku natáhli do strany, tak ji natáhli i částečně nahoru, což aktivovalo teleportaci. Výsledkem bylo, že kromě otočení okolo své osy se uživatel také nechtěně teleportoval na místo, kam zrovna mířil jeho ovladač.

### ■ 3.3.5 Nové metody prohlížení

Na základě zpětné vazby z prvotního testování a provedené rešerše bylo zapotřebí prohlížečící metody upravit. Zvolený díl LW modelu si bude možné prohlížet dvěma způsoby. Jeden ze způsobů bude založen na pohybu ve scéně, podobně, jak tomu bylo v prvotním návrhu. Druhý prohlížečící způsob bude založen na prohlížení zmenšeného dílu a manipulaci s ním.

U procházečícího způsobu prohlížení budou použity virtuální chodníky, nicméně dojde k menším změnám. Bude změněn materiál chodníků, a to tak, aby byl částečně průhledný. Tímto způsobem uživatel uvidí přímo pod sebe, ale stále bude mít jistotu, že se nevznáší ve vzduchu. Tento prohlížečící způsob bude použit dvěma způsoby. Tam, kde to bude možné a vhodné, se bude moci uživatel procházet po celém prohlíženém dílu (podobně jako tomu bylo u třetího způsobu v prvotním návrhu, viz kapitola 3.3.1). Pokud si však uživatel vybere na prohlížení díl s domy a ulicemi (většina LW modelu je pokryta právě domy a ulicemi), bude se uživatel procházet po virtuálních chodnících, které budou umístěny do ulic dílu (podobně jako tomu bylo u druhého prohlížečícího způsobu v prvotním návrhu, viz kapitola 3.3.1).

Druhý prohlížečící způsob, který bude použit, bude založen na přímé manipulaci s prohlíženým dílem. Tuto možnost jsem při prohlížení v aplikaci Sketchfab (viz kapitola 2.8.1) poměrně postrádal a jsem toho názoru, že uživateli dokáže nabídnout další a jiné pohledy na LW model. Nicméně vzhledem k tomu, že provedení nepřímé manipulace s interaktivní kuličkou v aplikaci Strata inStudio VR (viz kapitola 2.8.2) pro mě také nebylo dostatečně uspokojivé, bude moci uživatel s modelem manipulovat přímo. Uživateli bude dovoleno s dílem různě otáčet, ale také si díl bude moci zmenšovat či zvětšovat.

## ■ 3.4 Návrh scény

Aplikace na prohlížení LW modelu bude sloužit na detailní prohlížení menšího dílu z celého modelu, který si uživatel sám libovolně zvolí. Bude tedy potřeba aplikaci rozčlenit minimálně na dvě scény – jednu, ve které se uživateli zobrazí kompletní LW model a ve které si také vybere jednu část modelu k prohlížení, a druhou, ve které se načte zvolený díl modelu v plném rozlišení. Jak již bylo zmíněno výše, od prvotního testování se bude měnit způsob prohlížení, bude se přidávat nový. Tato změna bude aplikaci rozšiřovat o další scénu. Tedy uživatel si nejprve v první scéně zvolí díl k prohlížení,



poté bude přesunut do scény, kde si bude díl v plném rozlišení prohlížet jedním ze zmíněných způsobů a současně bude v aktuální scéně nějaký prvek, který aktivuje přesun do scény třetí, kde bude použit druhý prohlížeč způsob. Samozřejmě bude jak libovolné přepínání mezi těmito způsoby prohlížení, tak návrat do původní scény s kompletním LW modelem, kde si bude moci uživatel vybrat na prohlížení jinou část modelu.

### 3.5 Úprava dat

Po prvotním testování<sup>1</sup> výchozích dat vyšlo najevo, že bude nutné data upravit, a to z více důvodů. Tím hlavním však byla nízká hodnota FPS<sup>2</sup> v náhlavním displeji ve chvíli, kdy byl do scény načten celý LW model v nízkém rozlišení. Snímková frekvence se po načtení pohybovala mezi 1-2 FPS. Druhý důvod, který však nebyl ani zdaleka tak závažný, byl poměrně dlouhý čas načítání – celý model v nízkém rozlišení se do scény načtl za 7,6 s. Bylo by vhodné, kdyby čas načítání byl kratší, protože celý model se při prohlížení bude načítat vícekrát – vždy když si uživatel bude chtít vybrat jiný díl.

Příčina problémů nebyla na první pohled zcela jasná, jelikož digitalizovaný model neměl tolik plošek, aby to takto razantně snížilo snímkovou frekvenci a textury, které model využíval, také neměly příliš vysoké rozlišení, které by mohlo způsobovat tento problém. Jádro problému bylo v materiálech modelu. Data modelu využívala nespočet materiálů (řádově nízké desítky tisíc), kde každý materiál používal jednu malou texturu – tedy textur bylo obrovské množství, ale byly malé.

Prvotní testování také ukázalo, že načtení tří dílů LW modelu v plném rozlišení trvá 7,4 s. Ve výsledné aplikaci se v plném rozlišení bude načítat podobné množství dílů a ani tento čas není moc uspokojivý. Příčina dlouhého načítání modelu v plném rozlišení byla stejná jako u modelu v nízkém rozlišení, tedy velké množství materiálů využívajících mnoho malých textur.

Řešením problému se jeví přepečení<sup>3</sup> textur, a to tak, že textur bude méně, ale budou větší. Tímto způsobem bude model využívat menší počet materiálů a obraz v náhlavním displeji bude plynulejší a načítací časy kratší. Tvorba nových textur bude potřeba jak pro sadu dat v nízkém rozlišení, tak pro sadu dat v plném rozlišení.

Také bude zapotřebí celý LW model rozdělit na menší části, za účelem snížení výpočetní náročnosti spojené s prohlížením modelu ve vysokém rozlišení. Jak bylo popsáno v předchozí kapitole, uživatel si nejprve vybere z celého LW modelu menší část k prohlížení, která se mu následně načte ve vysokém rozlišení. Avšak kompletní LW model se ve výchozím stavu skládá pouze ze šesti částí, bude tedy nutné tyto části dále rozčlenit.

### 3.6 Způsob načítání dat

Vzhledem k velikosti a paměťové náročnosti celého modelu ve vysokém rozlišení, bude v jeden moment ve scéně načten právě jeden díl k prohlížení, tedy ten díl, který si uživatel zvolí. Z tohoto důvodu bude zapotřebí zajistit načtení správného dílu až po spuštění aplikace. Stejným způsobem bude provedeno načítání LW modelu do scény, kde bude zobrazeno celé město, tedy model v nižším rozlišení.

Jak již bylo zmíněno v kapitole Analýza výchozích dat (viz kapitola 2.6), data modelu mají strukturu .obj souborů spolu s obrázkovými texturami. Tento typ dat lze v Unity

<sup>1</sup> Testování, včetně měření načítacích časů, probíhalo na počítači se specifikacemi uvedenými v 3.3.3

<sup>2</sup> FPS - Frames Per Second (= počet snímků za sekundu)

<sup>3</sup> Pečení (baking) - Proces zapékání materiálů do textury

zpracovat několika způsoby. Jedním z nich je použití načítacího programu (potažmo skriptu), který dokáže načítat data tohoto typu za chodu programu. Takovým načítacím programem je například asset Runtime OBJ Importer, dostupný z Unity Asset Storu<sup>1</sup>. Postup použití by spočíval v načtení modelu spolu s texturami a následném upravování vlastností modelu (velikost, rotace, pozice). V aplikaci na prohlížení LW modelu bude použit zvýrazňovací asset Quick Outline (rovněž dostupný na Unity Asset Storu<sup>2</sup>), který bude ve scéně s celým LW modelem zvýrazňovat díl, na který uživatel aktuálně míří ovladačem. Tento zvýrazňovací asset se používá přidáním skriptu ke konkrétnímu objektu, který bude následně zvýrazněn. Při načítání do scény s výběrem dílu by tedy po načtení a úpravě vlastností následovalo ještě přidání a nastavení zvýrazňovacího assetu, případně přidání dalších funkčních prvků a jejich nastavení.

Pro tuto práci jsem se však rozhodl použít jiný způsob načítání. Jak již bylo zmíněno v kapitole 2.4, tak Unity nabízí svůj vlastní způsob načtení assetů za běhu programu – použitím tzv. asset balíčků. Kromě toho, že je tato funkcionality již přímo implementována v Unity, mají asset balíčky i další výhodu. Do asset balíčků lze v Unity zabalit pouze prefab. Prefab je v Unity objekt, který již byl do scény v herním enginu umístěn a má nastavené své vlastnosti a komponenty<sup>3</sup>. Nejčastější použití prefabu spočívá ve vícenásobném použití stejného objektu, který již byl definován. Tím, že je použití asset balíčků v Unity podmíněno použitím prefabu, nabízí tento způsob velkou výhodu oproti použití načítacího programu pro `.obj` soubory. Výhoda spočívá v nastavení všech potřebných vlastností a komponent ještě před vytvořením asset balíčku. Oproti použití načítacího programu pro `.obj` soubory bude tedy ušetřena výpočetní náročnost, která by vznikla po načtení modelu do scény. Z tohoto důvodu budou použity asset balíčky.

<sup>1</sup> <https://assetstore.unity.com/packages/tools/modeling/runtime-obj-importer-49547>

<sup>2</sup> <https://assetstore.unity.com/packages/tools/particles-effects/quick-outline-115488>

<sup>3</sup> Objektům v Unity se přidávají tzv. komponenty, které objektu poskytují nové vlastnosti či chování. Komponentou může být např.: Collider (komponenta detekující kolize objektu) nebo také skript.



# Kapitola 4

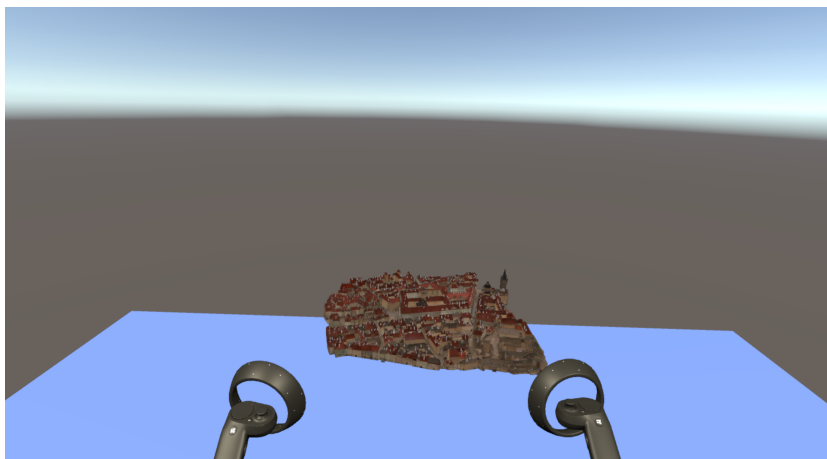
## Implementace

V této kapitole bude popsáno, jak konkrétně probíhala implementace aplikace na prohlížení LW modelu Prahy.

### 4.1 Způsoby prohlížení

Na základě prvotního návrhu prohlížečích metod a jejich otestování došlo k úpravě prohlížečích způsobů. V aplikaci byly implementovány dva prohlížečí způsoby – „manipulační“ způsob a procházečí způsob.

Manipulační prohlížečí způsob spočívá, jak již název napovídá, v přímé manipulaci s modelem. Tento prohlížečí způsob byl koncipován pro sedícího či stojícího uživatele, tedy uživatele, který se nebude ve scéně pohybovat. Prohlížení začíná tak, že se uživatel ocitne ve scéně. Před ním se nachází zmenšený díl LW modelu, který si uživatel vybral. Kromě toho, že se uživatel může nad model naklánět a různě si ho prohlížet, je také možné s modelem různě otáčet. S modelem lze otáčet do stran, a to za pomoci analogového joysticku. Když uživatel natáhne joystick na pravém ovladači doleva/doprava, model se začne pomalu otáčet na danou stranu. Stejným způsobem se bude model otáčet, pokud uživatel stiskne a podrží tlačítko trigger na pravém ovladači, a přitom bude pravou rukou pohybovat doleva či doprava. Dalo by se říct, že je toto otáčení rychlejší, nicméně je plně pod kontrolou uživatele. Pokud tedy uživatel chce, lze tímto způsobem otáčet s modelem i pomaleji, a to pomalejším pohybem ruky. Pro případ, že by si například sedící uživatel chtěl model prohlédnout seshora, je možné model naklopit směrem k uživateli. Naklopení se ovládá joystickem na levém ovladači. Natažení joysticku směrem dolů začne naklápět model k uživateli a natažení joysticku směrem nahoru vrací model do původního stavu naklopení. Uživatel může také libovolně měnit velikost modelu. A to tak, že nejdříve na obou ovladačích zároveň stiskne a podrží tlačítko grip. Poté může vzdalováním rukou od sebe model zvětšovat, a naopak přibližováním rukou k sobě model zmenšovat. Po uvolnění tlačítka grip si model zachová danou velikost.



**Obrázek 4.1.** Ukázka ze scény s manipulačním způsobem prohlížení

Procházezí prohlížečí způsob je implementačně o poznání jednodušší, nicméně i tak dokáže nabídnout zajímavé pohledy na LW model. Procházezí způsob je založen na pohybu, respektive teleportaci, ve scéně. Uživatel se na začátku objeví ve scéně a přímo před ním se vykreslí zvětšený díl LW modelu, který si uživatel vybral. Díl je zvětšen do takové míry, aby měl uživatel alespoň trochu pocit, že se prochází opravdovým městem. Kromě samotného dílu se ve scéně nachází cestičky, po kterých se může uživatel procházet. Cestičky byly vytvořeny z materiálu modré barvy. Od prvotního návrhu došlo ke změně typu materiálu – materiál je nyní poloprůhledný.

Pokud si uživatel vybere na prohlížení díl s domy a ulicemi (většina dílů), může se procházet ulicemi města. Cestičky jsou do dílu umístěny tak, aby se uživatel mohl projít po větších ulicích daného dílu, případně po náměstí. Cestičky tedy nejsou rozmístěny do všech ulic. Pokud si však uživatel vybere díl modelu, ve kterém se nachází pole a statky (takový díl je pouze jeden), je možné se procházet přímo nad modelem v pomyslném těle obra. K ovládání pohybu, respektive teleportace, se používá analogový joystick na ovladači. Pokud uživatel natáhne joystick (na libovolném ovladači) směrem nahoru a zamíří ovladačem k zemi, tak se na cestičce objeví zelený kruh. Tento kruh znázorňuje místo, na které se uživatel teleportuje po uvolnění páčky. Tímto způsobem se může uživatel vydat na prohlídku přímo do modelu. Ve scéně je také zajištěný způsob návratu na výchozí pozici. Uživatel toho může využít například za účelem zvolení jiné trasy prohlídky. Návrat na začátek je zajištěn stisknutím tlačítka pro nastavení (na libovolném ovladači).



**Obrázek 4.2.** Ukázka ze scény s procházečím způsobem prohlížení

## 4.2 Příprava modelu

Časově velice náročnou částí byla příprava samotného modelu k prohlížení. Výchozí data byla pro VR aplikaci téměř nepoužitelná a bylo tedy nutné model upravit. Všechny úpravy však byly pouze optimalizační, tedy nikde nedošlo ke změně samotného 3D modelu. Veškeré úpravy modelu byly prováděny v programu Blender.

### 4.2.1 Rozdělování modelu

Ve výchozím stavu se celé město (celý digitalizovaný LW model) skládalo ze šesti dílů. Bylo tedy jasné, že se model bude muset rozdělit na menší části, neboť prohlížení jedné šestiny města v plném rozlišení by bylo jednak výpočetně náročné a také by pro uživatele nebylo tolik přínosné. Také by bylo vhodné, aby dělení modelu bylo logické,

tedy aby se například půlka Staroměstského náměstí nenacházela v jednom díle a druhá půlka v díle jiném. Originální fyzický LW model se skládá z 52 dílů. Původní záměr byl takový, že si uživatel bude prohlížet díl, který by velikostně odpovídal zhruba třem až čtyřem dílům původního modelu, tedy výsledných dílů by bylo 13 až 18. V průběhu rozdělování modelu se však ukázalo, že výsledných dílů bude o něco více ale budou menší. Po finálních úpravách byl model rozdělen na 24 dílů (viz obrázek 4.3).



**Obrázek 4.3.** LW model rozdělený na 24 dílů

#### 4.2.2 Retexturování

Druhým krokem v úpravě výchozích dat bylo retexturování, tedy přepečení výchozích textur do textur nových. Jak již bylo zmíněno, výchozí model využíval obrovské množství malých textur, které zbytečně zvyšovalo výpočetní náročnost používání těchto dat. Plynulost obrazu je u aplikace na prohlížení 3D modelu důležitá, o to důležitější je, pokud se jedná o aplikaci pro virtuální realitu, kde by nízké hodnoty FPS mohly uživateli způsobovat například nevolnost. Tvorba nových textur tedy byla nutná. Výchozí data obsahovala 3D objekty LW modelu, jednu sadu textur v nízkém rozlišení a jednu sadu ve vysokém rozlišení. Bylo potřeba vytvořit nové textury pro obě sady. Volba vhodného rozlišení textur, ať pro sadu ve vysokém rozlišení nebo pro sadu v nízkém rozlišení, byla klíčová.

Pro textury v nízkém rozlišení bylo omezení jasné, načítání daného dílu muselo být rychlé, jelikož se do scény pro výběr dílu k prohlížení bude načítat všech 24 dílů. Z tohoto důvodu probíhalo testování za účelem najít vhodné rozlišení textury, jak z hlediska rychlosti načítání, tak z hlediska kvality textury, tedy aby se díl s novou texturou nenačetl takřka okamžitě na úroveň kvality textury. Pro pár dílů, z celkových 24, byla použita textura v rozlišení 4K (tedy 4096 x 4096 pixelů), tyto díly se načítaly v průměru za 0,6 s. Avšak pro většinu dílů bylo nutné použít texturu v nižším rozlišení, a to ve 2K (tedy 2048 x 2048 pixelů). Důvodem byl až příliš dlouhý čas načítání, kdy časy načítání s 4K texturou bez problému přesahovaly 30 s na jeden díl. Ale stejný díl se s texturou v 2K rozlišení načítal za pouhých 0,21 s. Textury ve 2K ovšem samozřejmě nedosahují takových detailů jako textury v rozlišeních vyšších. Nicméně tento fakt nebyl takovým problémem, jelikož když byl ve scéně načten díl s texturou ve 2K rozlišení, tak se nacházel v takové vzdálenosti od kamery, resp. uživatele, že rozdíl mezi 2K a 4K texturou byl na náhlavním displeji neznatelný.

Druhým omezením pro textury v nízkém rozlišení, které nebylo na první pohled vůbec zřejmé, je právě jedna textura pro jeden díl. Důvodem pro toto omezení je zvýrazňování dílu při výběru. Zvýrazňovací asset, který je v aplikaci použit, funguje tak, že barevně obtáhne okraje geometrie, nicméně pouze pro první materiál (tedy texturu). Když bylo textur použito více, tak se například barevně obtáhla pouze jedna stěna jednoho domu, místo celého dílu. Vypékání textur v nízkém rozlišení v programu Blender bylo poměrně rychlé, textura byla většinou hotová do pár vteřin.

Textury pro druhou sadu dílů, tedy pro ty ve vysokém rozlišení, žádná speciální omezení nevyžadovaly. Pouze bylo potřeba zajistit, aby se výsledný díl načetl za rozumný čas, nicméně to nebyl problém, jelikož ve scéně, ve které by se tento díl načítal, by byl pouze tento jediný díl. Jediným požadavkem byla kvalita výsledné textury – ta by se od kvality modelu ve výchozím stavu neměla moc lišit, respektive vůbec. Tím, že tyto díly nebyly zvýrazňovány, tak nebyly omezené na jednu texturu na díl. To přineslo možnost využít na jeden díl textur více a dosáhnout tak ještě vyšší kvality. Pro většinu dílů tak byly vytvořeny tři textury, u jednoho dílu byly vytvořeny čtyři. Každá z textur byla použita pro jinou část modelu. Až na výjimky bylo rozdělení následující – jedna textura byla použita pro všechny střechy daného dílu, druhá textura zachycovala všechny stěny domů a třetí textura byla použita pro zbylé objekty (komíny, podlahy, ploty a další). Co se rozlišení textur týče, tak pro textury ve vyšším rozlišení bylo zvoleno rozlišení 8K (tedy 8192 x 8192 pixelů). Bylo zvažováno i rozlišení vyšší (16K), nicméně takto velké textury už dosahovaly vysoké paměťové náročnosti a zlepšení kvality u nich nebylo pozorováno (v porovnání s 8K texturami). Na obrázku 4.4 níže lze vidět porovnání různých rozlišení textur, v porovnání s originální texturou LW modelu.



**Obrázek 4.4.** Porovnání textur různých rozlišení. Rozlišení textur zleva: 1K, 2K, 4K, 8K a originální textura.

Vypékání textur ve vysokém rozlišení trvalo o poznání déle. Většina textur byla vypečena využitím grafické karty, která je na takové operace optimalizována. V tomto případě trval proces v průměru zhruba tři minuty. Avšak některé textury byly zřejmě tak obsáhlé, že je nebylo možné vypécti grafickou kartou a bylo potřeba tento proces provést za využití procesoru. Procesor počítače takovou operaci zvládne také, nicméně výrazně pomaleji – takto tvorba jedné textury trvala v průměru 15 minut.

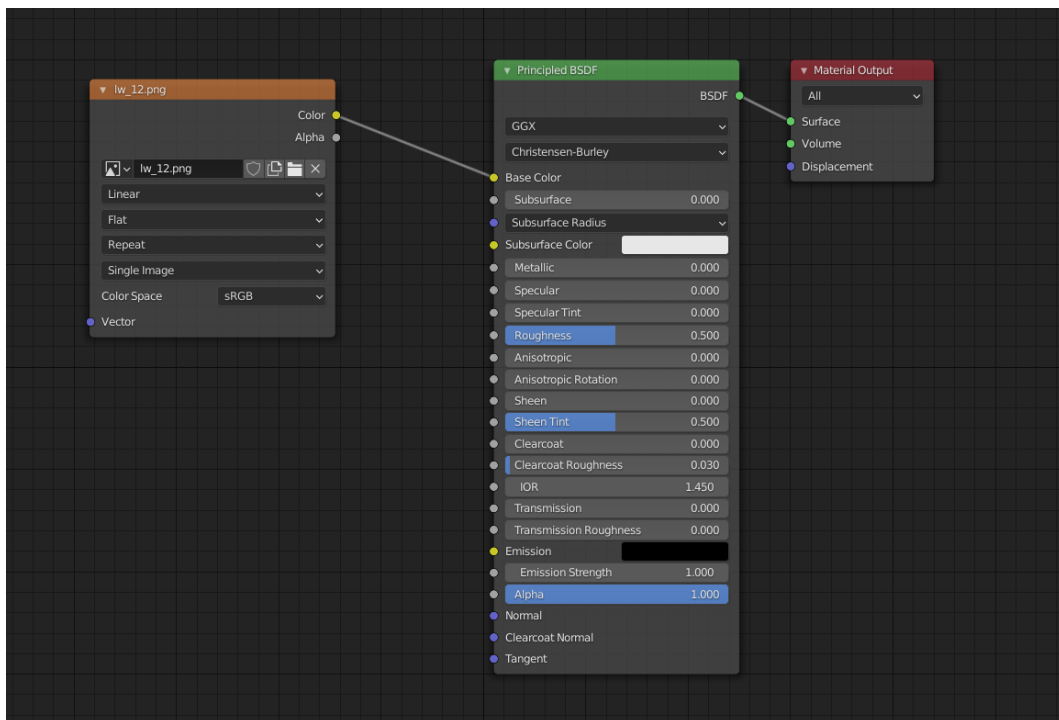
Struktura materiálu je běžně definována více parametry než jen barvou. Bývá dokonce samozřejmostí, že model používá více textur. Vždy bývá součástí modelu textura tzv. base color, neboli základ pro barvu. Dále může model používat tzv. specular



mapu(lesklost), roughness mapu (hrubost) či jiné mapy. Všechny takovéto mapy či textury dodávají plochám modelu další informace o materiálu. Zmíněná specular mapa nese informaci o tom, jak je které místo na modelu lesklé, naopak roughness mapa říká, jak je které místo hrubé. Tyto textury však nebyly při úpravě textur LW modelu vytvořeny, vypékala se pouze základní barva.

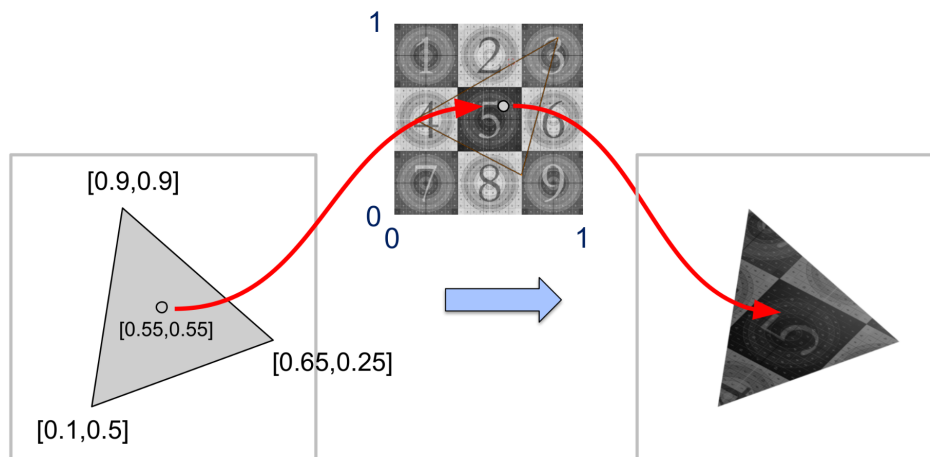
### 4.2.3 Příprava materiálů na vypečení textury

Při tvorbě nových textur bylo ještě před samotným zahájením procesu vypékání zapotřebí materiály správně nastavit tak, aby se vypekly do požadované textury a aby se vypekly správně. Materiály v programu Blender jsou reprezentovány uzlovým grafem (viz obrázek 4.5), ve kterém může uživatel přidávat a měnit uzly různých funkcí. Uzly fungují tak, že jim je přidělena hodnota na vstup, uzel hodnotu zpracuje a výsledek pošle do výstupu. Uzly se mezi sebou dají různě spojovat a kombinovat, tedy výstup jednoho uzlu může být připojen na vstup uzlu jiného. Jedním z uzlů je například Image Texture (tedy textura z obrázku), který má na vstupu obrázkovou texturu a výstupem je barva. Výstupní barva je následně aplikována na 3D objekt podle příslušných texturovacích souřadnic (dále jen UV souřadnice).



**Obrázek 4.5.** Uzlový graf v programu Blender

UV souřadnice říkají informaci o tom, kde v textuře nalezneme barvu příslušného vrcholu. Tedy každý vrchol 3D modelu, který využívá texturu, má mj. uložené své UV souřadnice. UV souřadnice jsou dvě čísla –  $u$  a  $v$ , kde  $u$  reprezentuje souřadnici příslušného pixelu v textuře na ose X, podobně  $v$  reprezentuje souřadnici na ose Y. Tato dvě čísla jsou normalizována do intervalu  $\langle 0; 1 \rangle$ , tedy bod, který je UV souřadnicemi reprezentován jako  $[0, 0]$  odpovídá levému dolnímu rohu textury.



**Obrázek 4.6.** Proces nanesení textury na trojúhelník. Zdroj: [12].

Jak již bylo zmíněno, ve výchozím stavu byla data LW modelu rozdělená do šesti dílů a také bylo zmíněno, že model ve výchozím stavu používá velké množství malých textur. Například se pro každou střechu každého domu využívala jedna textura, tedy každá střecha měla své vlastní UV souřadnice (totéž platí pro všechny ostatní části modelu – stěny, komíny atd.). Tento fakt měl za důsledek následující – UV souřadnice jedné střechy se překrývaly s UV souřadnicemi střechy druhé (platí obecně, nejen pro střechy). Nicméně ve výsledku chceme dosáhnout toho, že celý jeden díl z celkových 24, bude používat právě jednu texturu (pro díly využívající textury v nízkém rozlišení), není tedy přípustné, aby se UV souřadnice v rámci daného dílu překrývaly. Z tohoto důvodu musela být vytvořena nová tzv. UV mapa (soubor UV souřadnic pro celý objekt). Avšak nová UV mapa byla nejdříve pouze přidána k té původní (tedy v této fázi měl objekt UV mapy dvě). Objekty v Blenderu mohou mít UV map přiřazených více, nicméně pouze jedna může být nastavena jako aktivní – tedy ta, podle které se bude objekt obarvovat.

Vypékání textur v Blenderu funguje na jednoduchém principu – zvolí se, do jaké textury má program materiály vypékat a spustí se proces. Kromě cílové textury je také možné si nastavit, podle které UV mapy se má do textury zapékat. Volba cílové textury je v Blenderu provedena přidáním Image Texture uzlu do grafu daného materiálu, vybráním cílové textury v nově přidaném uzlu a následným zaktivněním uzlu. Aby vypékání proběhlo správně, je potřeba mít v grafu materiálu právě jeden aktivní uzel typu Image Texture. Uzel Image Texture má kromě možnosti výběru textury, jakožto vstupní položky, také možnost zvolit vstupní UV mapu. A to přidáním nového uzlu typu UV map a připojením do vstupu uzlu Image Texture. Blender standardně do textury vypéká to, co se aktuálně nachází na objektu (tedy podle aktivní UV mapy). Takže postup byl takový, že aktivní UV mapou na objektu byla původní UV mapa a pro aktivní uzel Image Texture s cílovou texturou byl vytvořen a připojen nový uzel UV map s nově vytvořenou UV mapou.

Nicméně tento proces (přidání dvou nových uzlů do grafu materiálu a jejich následné nastavení) bylo zapotřebí provést na všech materiálech daného dílu, kterých bylo několik tisíc. Pro tento účel byl vytvořen skript v jazyce Python, který vše zajistí. Po doběhnutí skriptu už jen stačilo zahájit proces vypékání. Po vypečení všech textur proběhl v programu Blender export do .obj souboru. V rámci exportu byl automaticky vytvořen i .mtl soubor. Struktura dat tedy zůstala stejná jako ve výchozím stavu, jen došlo k úpravám textur a rozdělení modelu na více dílů. Modely v tomto formátu nejsou v rámci práce odevzdány, nicméně jsou k dispozici u vedoucího práce.

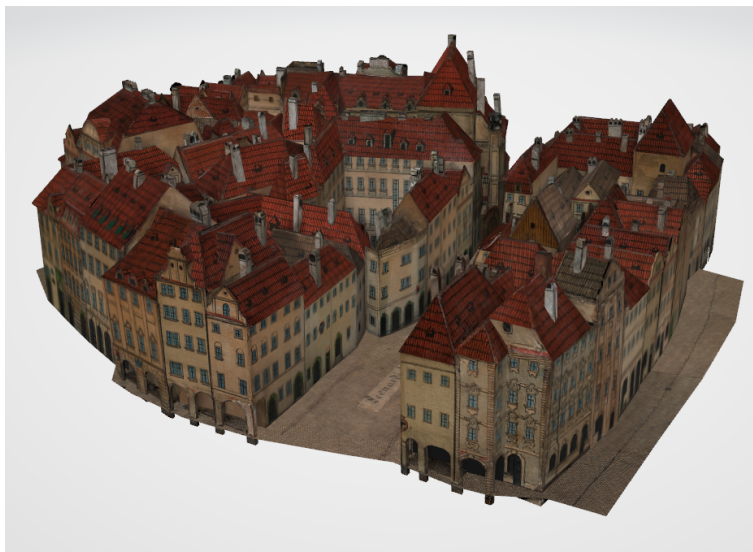
#### 4.2.4 Asset balíčky

Načítání dílů LW modelu do Unity scény je realizováno vyžitím asset balíčků. Jak již bylo zmíněno v kapitole 3.6, do asset balíčků v Unity lze pouze zabalit prefaby. Poté, co byly nově vytvořené díly vyexportovány, bylo zapotřebí si je vložit do Unity scény a všechno nastavit.

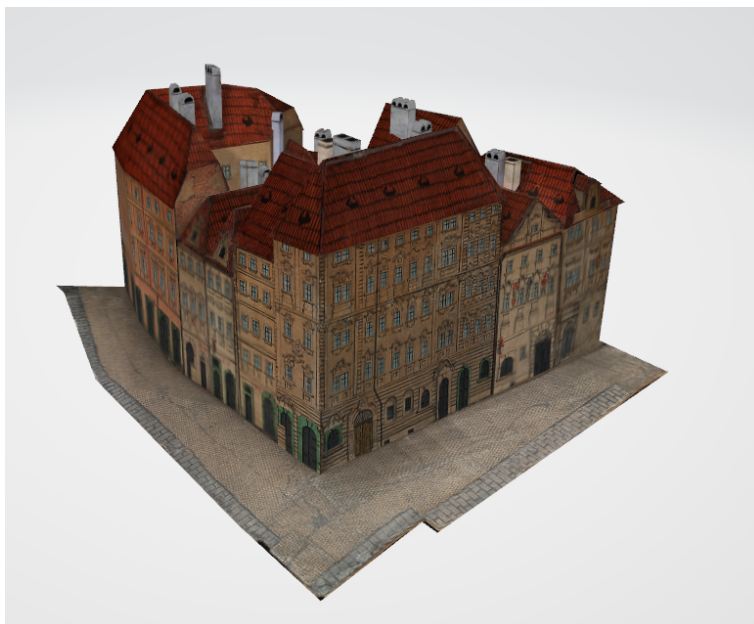
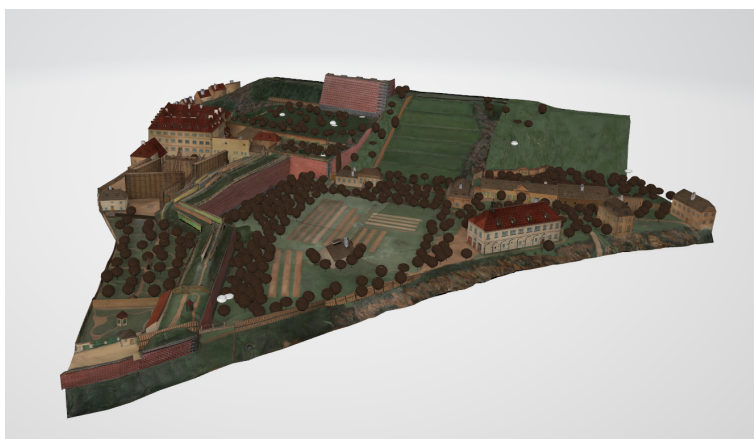
U dílů v nízkém rozlišení bylo zapotřebí k objektu daného dílu přidat a nastavit komponenty typu Collider a zvýrazňovací skript. Využití komponenty Collider bylo nutné z důvodu identifikace dílu, na který uživatel aktuálně míří. Zvýrazňovací skript slouží k obtažení dílu, na který uživatel aktuálně míří. Poté proběhlo nastavení dalších vlastností jako je Tag a Vrstva, které se také využívají k identifikaci.

U dílu ve vysokém rozlišení nebylo potřeba přidávat žádné komponenty, nicméně bylo zapotřebí vytvořit jistou hierarchii daného objektu. Konkrétně bylo potřeba zajistit, aby v manipulačním prohlížečím způsobu nedošlo ke změně os otáčení. Jak již bylo zmíněno v předchozí kapitole, model je možné rotovat kolem své osy doleva a doprava, ale také je možné model naklápět směrem k uživateli. Bylo potřeba, aby bylo možné s modelem rotovat doleva a doprava i ve chvíli, kdy je model nakloněný k uživateli. Byla vytvořena následující hierarchie objektu. V Unity byl vytvořen nový prázdný objekt, který byl nastaven jako rodičovský objekt danému dílu LW objektu. Tedy struktura byla rodičovský objekt, který měl jako svého potomka díl LW modelu ve vysokém rozlišení. Využití při rotaci a naklonění bylo následující. Pokud uživatel naklopil díl směrem k sobě, změnil tak rotaci rodičovskému objektu. Avšak díky použití zmíněné hierarchie se v moment naklonění naklonil i díl LW modelu. S nakloněním prohlíženého dílu se naklonila i osa rotace, okolo které je možné model otáčet doleva a doprava. Pokud by tato hierarchie nebyla použita, nebylo by možné tyto dvě rotace kombinovat.

Po nastavení objektů v Unity a vytvoření příslušných prefabů bylo možné asset balíčky vytvořit. Při tvorbě balíčků je možné do jednoho balíčku vložit libovolné množství prefabů. S asset balíčky tedy proběhlo testování, za účelem zjistit, jaký způsob zabalení je nejefektivnější, tedy jaký počet prefabů v balíčku je ideální. Testování probíhalo se sadou dílů, která byla k dispozici při prvotním testování – tedy se šesti díly v nízkém rozlišení (reprezentující celý LW model) a se třemi díly ve vysokém rozlišení (tyto tři díly lze vidět na následujících obrázcích).



Obrázek 4.7. Díl č. 8

**Obrázek 4.8.** Díl č. 9**Obrázek 4.9.** Díl č. 27

Výsledkem testování vyšlo najevo, že nejrychlejší způsob načítání LW modelu je takový, kde každý díl má svůj vlastní balíček. Tímto způsobem se všech šest dílů v nízkém rozlišení načítalo za 7,6 s (1,0 s až 1,5 s na jeden díl). Druhý, také rychlý způsob spočíval v tom, že všechny díly byly v jednom balíčku dohromady – tímto způsobem trvalo načítání 8,2 s.

Časy načítání byly také změřeny pro tři díly ve vysokém rozlišení. Pro tyto tři díly byly časy změřeny před úpravou textur (tedy ve výchozím stavu – mnoho textur) i po úpravě (pouze jedna textura). Právě úprava textur byla nejdříve testována pouze na těchto třech dílech, a teprve poté byly vytvořeny nové textury pro celý LW model.

V tabulce 4.1 lze vidět časy načítání jednotlivých dílů ve výchozím stavu a v tabulce 4.2 lze vidět časy načítání stejných dílů, avšak po úpravě textur. Jednalo se o díly s číslem 8, 9 a 27 (viz obrázky 4.7, 4.8 a 4.9). Na základě předchozího testování byl do asset balíčku vždy umístěn pouze jeden daný díl. Kromě času načtení jsou v tabulce uvedeny detaily jako počet trojúhelníků daného dílu, paměťová náročnost textur a jejich počet.



| číslo dílu | počet trojúhelníků | počet textur | velikost textur | čas načtení |
|------------|--------------------|--------------|-----------------|-------------|
| 8          | 7 693              | 172          | 238 MB          | 2.2 s       |
| 9          | 800                | 67           | 53 MB           | 0.5 s       |
| 27         | 103 931            | 243          | 376 MB          | 4.7 s       |

**Tabulka 4.1.** Tabulka porovnání časů načítání dílů ve výchozím stavu

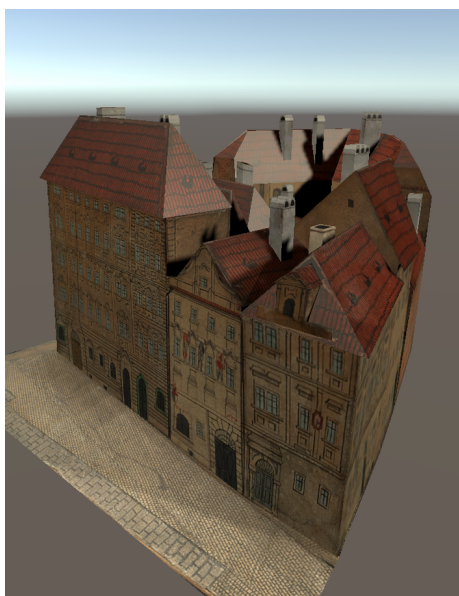
| číslo dílu | počet trojúhelníků | počet textur | velikost textur | čas načtení |
|------------|--------------------|--------------|-----------------|-------------|
| 8          | 7 693              | 1            | 102 MB          | 1.1 s       |
| 9          | 800                | 1            | 24 MB           | 0.3 s       |
| 27         | 103 931            | 1            | 346 MB          | 1.2 s       |

**Tabulka 4.2.** Tabulka porovnání časů načítání dílů po úpravě textur.

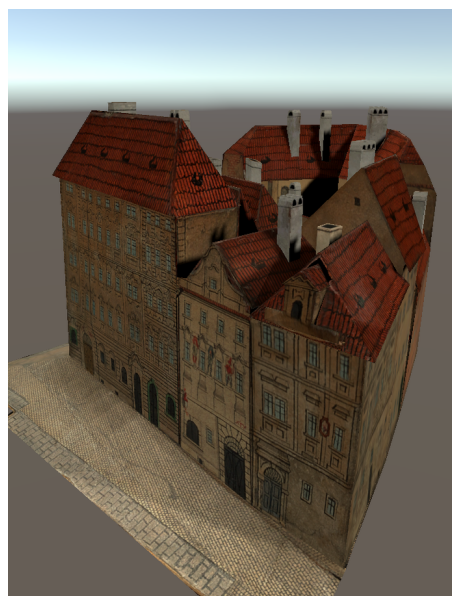
Jak lze vidět v tabulce 4.2, tak se načítací časy výrazně snížily. Podle velikosti textur lze říct, že došlo ke snížení rozlišení. To je pravda, nicméně rozdíl v náhlavním displeji byl téměř neznatelný. Závěrem tedy byly vytvořeny asset balíčky jak pro sadu modelů v nízkém rozlišení, tak pro sadu ve vysokém rozlišení. Díly LW modelu byly do balíčku zabaleny tak, že každý díl měl svůj vlastní asset balíček. Tímto způsobem se celý LW model v nízkém rozlišení načel za 4,5 s.

### 4.3 Materiály v Unity

Materiály digitalizovaného modelu byly lehce upraveny. Při prvních experimentech s prohlížením modelu se ihned ukázalo, že je model příliš lesklý. Materiály samotného modelu byly tedy pozmeněny. A to konkrétně jejich „smoothness“ (jemnost), která byla nastavena na 0 (původní hodnota byla 1) a byly odstraněny „specular highlights“ (zrcadlové zvýraznění). Na obrázcích 4.10 a 4.11 níže lze vidět znatelný rozdíl – nejlépe je možné rozdíl vidět na střechách domů. Po této změně se model při prohlížení jevil lépe.

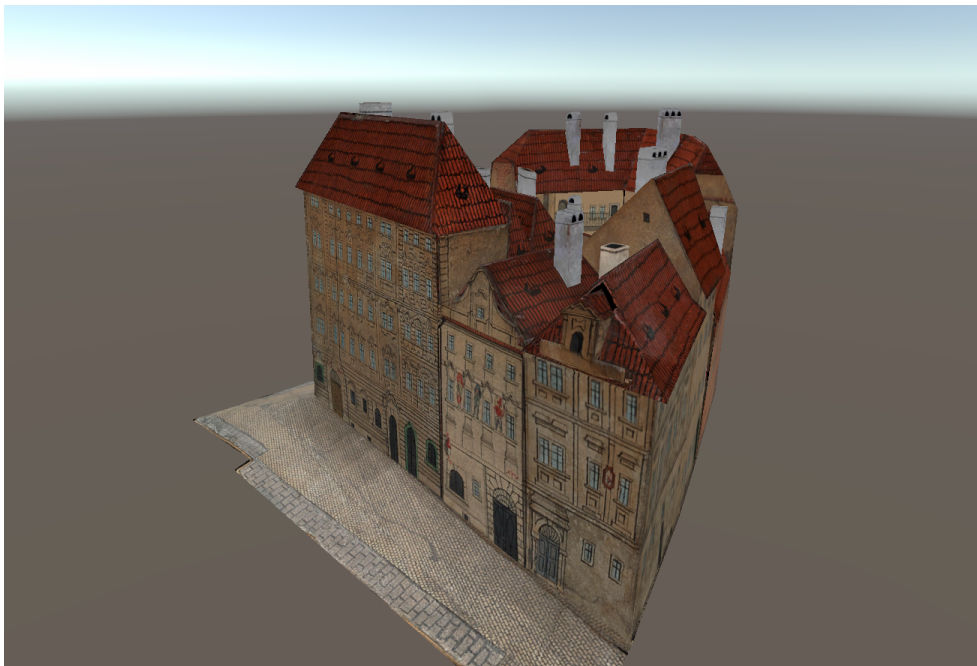


**Obrázek 4.10.** Výchozí materiál



**Obrázek 4.11.** Materiál po odstranění odlesků

Nicméně po důkladnějším testování osvětlení modelu vyšlo najevo, že je zapotřebí do scény přidat další zdroje světla, aby se netvořily tmavé stíny na stěnách domů modelu. Přidání světel tento problém vyřešilo, ale také zvýšilo výpočetní náročnost. Byla tedy provedena poslední úprava materiálu, a to přímo změna z materiálu typu „lit“ (osvětlený) na materiál typu „unlit“ (neosvětlený). Po této změně vypadal model nejlépe ze všech předchozích možností a také byla snížena výpočetní náročnost, protože se osvětlení na dílech LW modelu nevypočítávalo. Na obrázku 4.12 níže lze vidět, jak vypadá materiál po finální změně. Takto byl materiál v aplikaci použit.

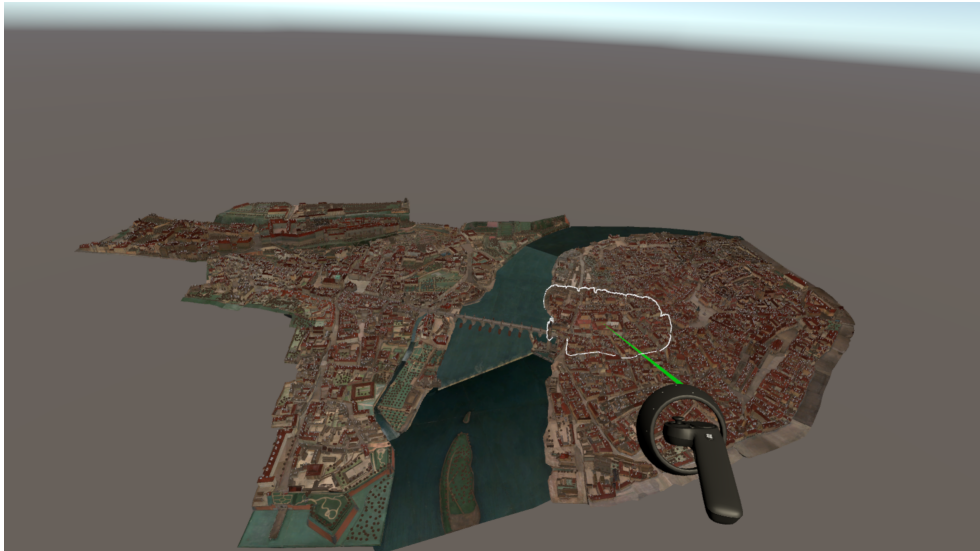


**Obrázek 4.12.** Materiál modelu po změně na typ unlit

## 4.4 Struktura aplikace

V této kapitole je popsána struktura aplikace, jednotlivé scény, přechody mezi nimi a jak celá aplikace funguje.

Po spuštění aplikace, se uživatel objeví v první scéně, ve které se téměř nic nenachází. Je zde k vidění pouze plocha, na které uživatel stojí a krychle s nápisem „Start“. Přechody mezi scénami jsou v aplikaci realizovány interakcemi právě s těmito krychlemi. V této první scéně si může uživatel vyzkoušet, kam ovladačem míří – tato technika je později od uživatele vyžadována. Pokud uživatel zamíří pravým ovladačem směrem k zemi, objeví se zelený paprsek, který znázorňuje místo, kam ovladač aktuálně míří. Když uživatel tímto způsobem zamíří na krychli a stiskne na pravém ovladači tlačítko trigger, zahájí celé prohlížení. V následujícím momentu je uživatel přesunut do scény, ve které se po načtení objeví celý LW model (v nízkém rozlišení). Tato scéna slouží právě k výběru dílu k detailnímu prohlížení. Výběr provede uživatel podobně jako interakci s krychlí v předchozí scéně. Stačí namířit pravým ovladačem směrem k zobrazenému modelu a objeví se zelený paprsek a také se zvýrazní díl, na který aktuálně uživatel ovladačem míří. Poté, co se uživatel rozhodne, jaký díl si chce prohlédnout, stačí při míření na zvolený díl stisknout tlačítko trigger na pravém ovladači.



**Obrázek 4.13.** Scéna pro výběr dílu k prohlížení

Následuje další přesun, tentokrát do prohlížecké scény, kde je použit „manipulační“ způsob prohlížení. Kromě zmenšeného dílu LW modelu se ve scéně nacházejí dvě interakční krychle. Interakce s krychlí, nacházející se po levici uživatele, jej přenesou zpět do scény pro výběr dílu. Na této krychli se nachází nápis „Back to choosing“ (tedy „zpět k výběru“). Druhá krychle, která se nachází po pravici uživatele, nese nápis „Walking mode“ (tedy „procházecký způsob“). Interakcí s touto krychlí bude uživatel přesunut do další scény, ve které se k prohlížení používá procházecký způsob. V této poslední scéně se znovu, kromě zvětšeného dílu LW modelu, nachází dvě interakční krychle. Krychle na levé straně od uživatele má stejnou funkci jako krychle v minulé scéně, tedy navrátí uživatele do scény s výběrem dílu. Na druhé straně, tedy napravo od uživatele se nachází krychle se slovy „Back to viewing mode“ (tedy „zpět k prohlížeckému způsobu“). Interakce s touto krychlí přesune uživatele zpět do scény se zmenšeným dílem LW modelu, kde je použit manipulační způsob prohlížení.

Z knihovny SteamVR byly použity dva prefaby – Player a Teleporting. Prefab Player ve scéně reprezentuje uživatele. Jeho součástí je například kamera či ruce hráče, které se pohybují stejně jako ovladače, které uživatel drží v rukou. Prefab Teleporting ve scéně zajišťuje teleportaci – tento prefab byl použit pouze ve scéně s procházeckým způsobem prohlížení.

Směr paprsku z ovladače je v Unity získán použitím proměnné `Transform.Forward` z objektu `RightHand` (součást prefabu Player). Tato proměnná si uchovává vektor, který je určený směrem, kam aktuálně objekt míří. Stejná proměnná byla využita i při vrhání paprsku do scény, které slouží k detekci dílu, na který uživatel míří. Díl byl detekován díky komponentě `Collider`, která se nachází na všech dílech LW modelu v nízkém rozlišení.

## 4.5 Vykreslování paprsku ve SteamVR

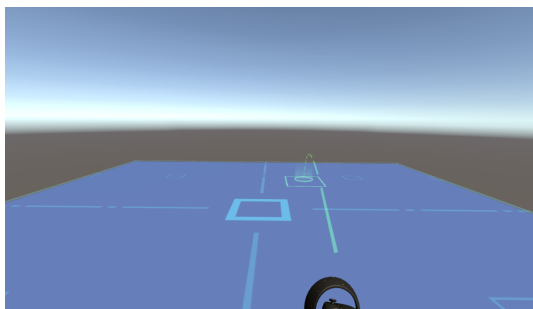
V průběhu práce se pochopitelně vyskytlo několik problémů. Jeden z nich bych tu ale rád zmínil. Jedná se o vykreslování objektu při použití knihovny SteamVR pro Unity ve vykreslovacím režimu `Single Pass Instanced`<sup>1</sup>. Tento vykreslovací režim funguje tak,

<sup>1</sup> <https://docs.unity3d.com/Manual/SinglePassInstancing.html>

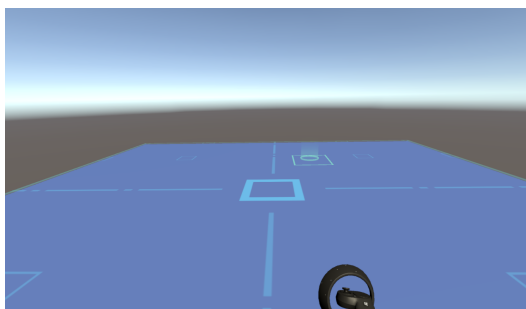
že obraz vykreslí pouze pro jedno oko. Vykreslený obraz se v náhlavní displeji zobrazí pro jedno oko a pro druhé oko se zobrazí stejný obraz, ale trochu upravený.

Aplikace byla původně vyvíjena právě ve zmíněném vykreslovacím režimu, nicméně se objevil zásadní problém, který se nepodařilo vyřešit. Jedná se o problém, kdy po načtení objektu z asset balíčku do scény byl objekt v náhlavní displeji vykreslen pouze na jedno oko. Objekty, které se ve scéně nacházely ještě před spuštěním, se vykreslovaly správně, avšak ty načtené z balíčků ne. Tento problém vyřešilo přepnutí aplikace do jiného vykreslovacího režimu, totiž do Multi Pass. Multi pass je vykreslování, které vykreslí obraz pro každé oko zvlášť. Je tedy pomalejší, ale v náhlavní displeji nebyl pozorován velký rozdíl.

Po změně režimu se objekty již správně vykreslovaly na obě oči. Netrvalo ale dlouho a objevil se problém s vykreslováním jiných objektů – paprsků (na vykreslování paprsků v Unity se používá komponenta Line renderer). Paprsky se vykreslovaly přesně opačně než načítané objekty – v režimu Single Pass Instanced fungovalo vykreslování správně, ale v režimu Multi Pass se paprsky nevykreslovaly vůbec. Po dalším pozorování se ukázalo, že v režimu Multi Pass se dokonce nevykreslují ani paprsky, které jsou součástí knihovny SteamVR (jedná se například o paprsek, který se vykreslí, když se chce uživatel ve scéně teleportovat, viz obrázky 4.14 a 4.15 níže). Vykreslování načtených objektů na obě oči bylo jasnou prioritou, aplikace tedy aktuálně pracuje v režimu Multi Pass.



**Obrázek 4.14.** Vykreslovací režim Single pass. Lze vidět paprsek teleportace.



**Obrázek 4.15.** Vykreslovací režim Multi pass. Paprsek se nevykresluje.

Po krátké konverzaci s Ing. Tomášem Havlíkem, na kterého jsem se s problémem ohledně paprsku obrátil, mi bylo sděleno, že se jedná o známý problém<sup>1</sup>, který nemá přímé řešení. Problém zřejmě spočívá někde v kombinaci použití knihovny SteamVR, použití vykreslovacího řetězce URP a vykreslovacího režimu Multi pass. Na doporučení pana Havlíka byl problém s nevykreslujícím se paprskem vyřešen vykreslením jednoho úzkého trojúhelníku namísto paprsku.

## 4.6 Ovládání

Tlačítko, které se na ovladači používá nejvíce, je tlačítko trigger na pravém ovladači. Slouží k interakci s krychlemi, které uživatele přesouvají mezi scénami, a ke zvolení dílu k prohlížení ve scéně s celým LW modelem. Ve scéně s manipulačním prohlížením toto tlačítko slouží na otáčení s daným dílem (v kombinaci s pohybem pravé ruky).

Analogové joysticky mají dvojí využití, a to v závislosti na tom, v jaké scéně se uživatel aktuálně nachází. Ve scéně s manipulačním způsobem prohlížení slouží k otáčení

<sup>1</sup> [https://github.com/ValveSoftware/steamvr\\_unity\\_plugin/issues/845](https://github.com/ValveSoftware/steamvr_unity_plugin/issues/845)

a naklápění s modelem – joystick na pravém ovladači slouží k otáčení a joystick na levém ovladači slouží k naklápění. Ve scéně s procházečím způsobem prohlížení slouží joystick na pravém ovladači k ovládní teleportace. Po natažení joysticku (na libovolném ovladači) směrem nahoru a namířením ruky směrem k zemi se na cestičce zobrazí zelený kruh. Po uvolnění joysticku je uživatel teleportován na místo, kde se nacházel kruh.

Na pravém ovladači je využito tlačítko pro nastavení, které se také používá ve scéně s procházečím způsobem prohlížení – používá se na přesun uživatele zpět na začátek.

Posledním využitým tlačítkem na ovladačích je tlačítko grip. Využívá se ve scéně s manipulačním prohlížecím způsobem, a to ke změně velikosti prohlíženého dílu. Pokud uživatel stiskne a podrží toto tlačítko na obou ovladačích zároveň a bude vzdalovat ruce od sebe navzájem, bude se model zvětšovat. Naopak přibližováním rukou k sobě se bude model zmenšovat. Po uvolnění kteréhokoliv tlačítka grip si model zachová velikost.

Tlačítka ovladače jsou popsána na obrázku 3.2.

# Kapitola 5

## Testování

V této kapitole je popsáno, jak byla aplikace na prohlížení LW modelu Prahy testována, jakým způsobem testování probíhalo a jaké jsou výsledky testování.

Cílovou skupinou, na kterou je tato aplikace zaměřená, jsou návštěvníci muzea. Takovými návštěvníky mohou být například žáci základní či střední školy. Testování proběhlo se dvěma skupinami žáků. V první skupině byli žáci paté až šesté třídy základní školy. Ve druhé skupině byli žáci tercie až kvarty z víceletého gymnázia (odpovídá osmé až deváté třídě ZŠ). Testování proběhlo v budově fakulty elektrotechnické ČVUT v Praze na Karlově náměstí. Žáci do školy přijeli za účelem testování výukového programu o tvorbě Langweilova modelu Prahy a v rámci tohoto programu si vyzkoušeli i VR aplikace, které jsou spojené s LW modelem, včetně této.

### 5.1 Průběh testování

Testování probíhalo v učebně školy, tedy na školním počítači. Aplikace byla testována s jiným náhlavním displejem, než na kterém byla vyvíjena. Nicméně díky využití knihovny SteamVR to nebyl problém - stačilo přemapovat jednotlivé akce na tlačítka použitého náhlavního displeje. Testovalo se na počítači s těmito specifikacemi:

- CPU: Intel Core i9-10900x
- GPU: NVIDIA GeForce GTX 1070 Ti
- RAM: 32 GB, 2933 MHz
- Náhlavní displej: Oculus Quest 2 (viz obrázek 5.1)



**Obrázek 5.1.** Náhlavní displej Oculus Quest 2 s ovladači. Zdroj: [13]



Použitý náhlavní displej Oculus Quest 2 nabízí způsob bezdrátového používání. Nicméně za účelem nejvyššího výkonu byl náhlavní displej používán v propojení s počítačem, a to kabelem USB-C -> USB-C.

Testování se zúčastnilo 18 uživatelů ve věku 10 až 15 let, kteří se postupně střídali. Před samotným testováním byl uživatel dotázán, jaké jsou jeho zkušenosti s VR. Pokud neměl žádné nebo malé zkušenosti, byl mu představen náhlavní displej, ovladače a jak VR funguje. Poté si uživatel nasadil náhlavní displej a chopil se ovladačů. V aplikaci nebyl implementován způsob, kterým by byl uživatel informován o ovládání aplikace. Z tohoto důvodu bylo ovládání vysvětlováno v průběhu testování. Pokud by ovládání bylo vysvětleno před samotným testováním, uživatel by byl zbytečně zahlcen mnoha informacemi a ovládání by si nezapamatoval.

Po spuštění aplikace byl uživatel postupně instruován k přechodům mezi scénami, k výběru a prohlížení zvoleného dílu LW modelu. Pokud bylo očividné, že se již uživatel v aplikaci trochu zorientoval a osvojil si její ovládání, byl mu uložen úkol. Úkol spočíval v nalezení mlýnského kola v modelu. To se v modelu nachází pouze jedno. Uživatel si tedy musel postupně k prohlížení vybírat díly, kde by se mohlo kolo nacházet. Po zvolení dílu se musel také zorientovat, jak je díl otočený a na které straně se nachází řeka. Pokud se uživateli povedlo nalézt mlýnské kolo rychle, tak mu byl uložen další hledací úkol. Tentokrát měl za úkol nalézt strážní budky vojáků – ty se v modelu nachází například před Pražským hradem nebo na Staroměstském náměstí. Budky jsou malé a někde jsou ukryté například pod stromem, tedy tento úkol byl těžší. Tyto úkoly byly spíše pro mladší skupinu žáků, kterým se líbila VR jako taková, ale samotný LW model je už tolik nenadchnul. U starších žáků to bylo obráceně – úkoly splnily rychle a poté obdivovaly LW model, jeho detaily a kvalitu digitalizovaného modelu.

Mladších žáků bylo poměrně mnoho a v testování se střídali zhruba po osmi minutách. Žáků starších však bylo méně a mohli si tak v klidu prohlédnout celý LW model. Ti se střídali po zhruba 13 minutách. Po uplynutí časového limitu byl uživatel požádán o zpětnou vazbu ve formě vyplnění připraveného dotazníku.

## 5.2 Dotazník

Vzhledem k věku žáků jsem v dotazníku respondentům tykal. Dotazník obsahoval následující otázky:

- Už si někdy zkoušel/a virtuální realitu?

Následovalo pět otázek, ve kterých mohli respondenti známkovat (1 až 5) jako ve škole:

- Jak se ti to celé líbilo?
- Jak se ti líbilo prohlížení modelu – otáčení a naklápění? (manipulační způsob)
- Jak se ti ovládalo otáčení, naklápění a změna velikosti? (manipulační způsob)
- Jak se ti líbilo procházení zvětšeným městem? (procházeční způsob)
- Jak se ti ovládalo procházení zvětšeným městem? (procházeční způsob)

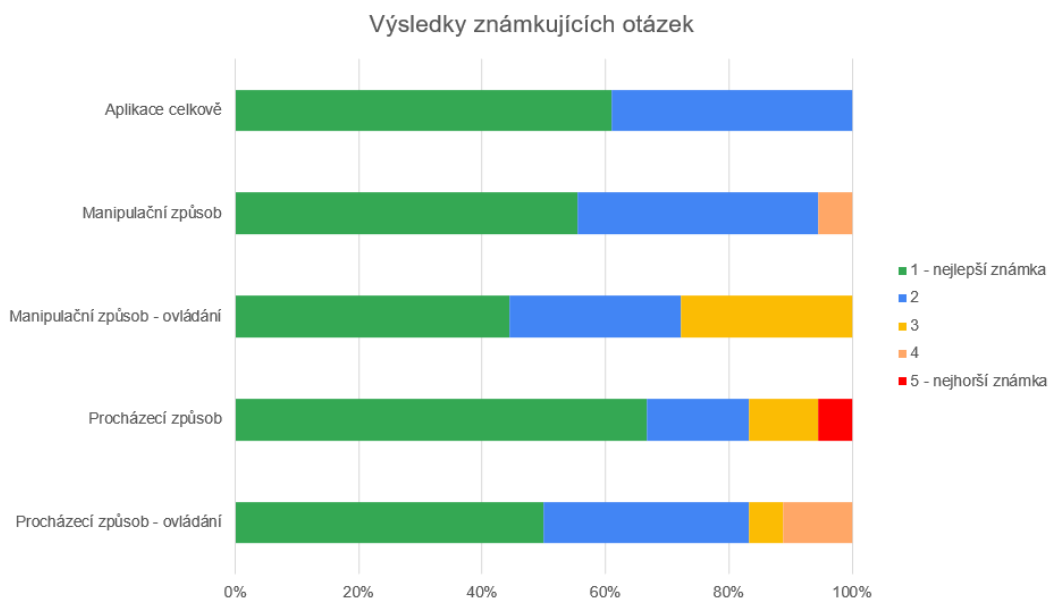
Následovaly další otázky:

- Dělal se ti při prohlížení špatně?
- Napadá tě, co by se dalo zlepšit? Nebo co by se dalo udělat jinak?
- Bylo něco, co se ti nelíbilo? Proč?
- Co se ti líbilo nejvíce?
- Chtěl/a bys mi něco vzkázat?

Konkrétní podobu dotazníku lze vidět v příloze C.

### 5.3 Výsledky testování

Z celkových 18 respondentů bylo 10 uživatelů, kteří již dříve zkoušeli VR. Celkový dojem z prohlížení byl u respondentů velmi pozitivní, průměrná známka prohlížení jako celku byla 1,4. Podle známek u oblíbenosti dvou způsobů prohlížení lze říct, že oba se uživatelům líbily stejně. Manipulační způsob prohlížení dostal celkovou známku 1,56 a procházeční způsob obdržel známku 1,61. Co se ovládání týče, tak i tam byly průměrné známky u obou prohlížečích způsobů téměř totožné (1,83 pro manipulační způsob a 1,78 pro procházeční způsob). Na obrázku 5.2 níže lze vidět odpovědi na jednotlivé známkovací otázky.



**Obrázek 5.2.** Graf známek pro jednotlivé otázky

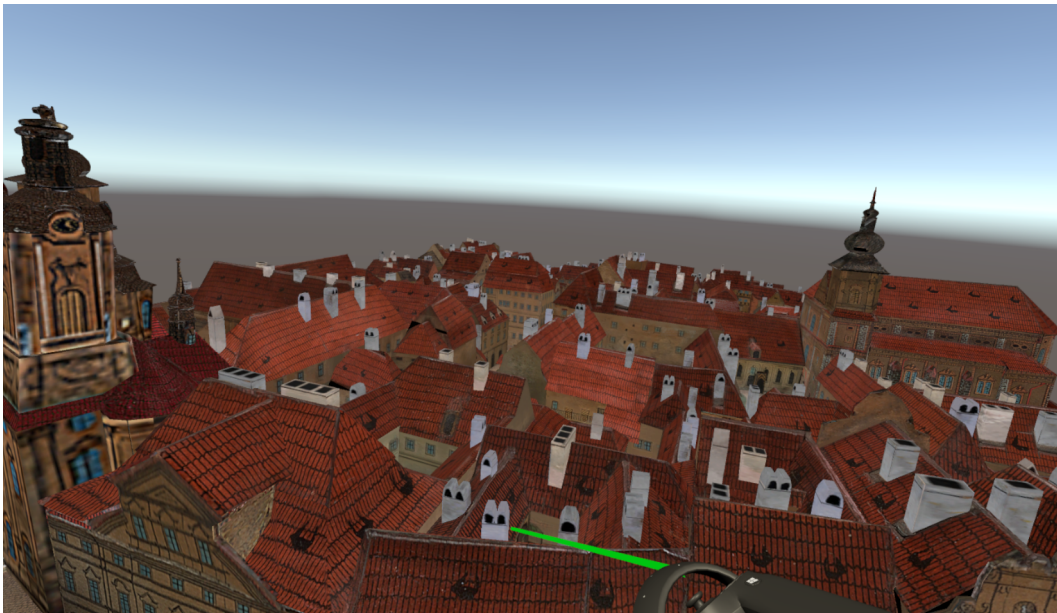
V průběhu prohlížení si dokonce jeden z uživatelů, který nikdy předtím nezkoušel VR, pochvaloval jednoduchost ovládání. I přesto, že oba prohlížeční způsoby dosáhly podobných známek, tak v otázce, co se uživatelům líbilo nejvíce jeden způsob dominoval. Více oblíbený byl procházeční způsob. Z 18 respondentů jen dva odpověděli, že se jim dělalo trochu nevolno, ostatním se nevolno nedělalo.

V dotazníku se nevyskytlo mnoho zpětné vazby k tomu, co je v aplikaci uděláno špatně. I přesto ale bylo několik problémů v průběhu testování vyzorováno. První z nich byl problém se směrem, kam ve scéně míří ovladač. Tento problém se vyskytoval opravdu často. Docházelo k tomu, že když chtěl uživatel namířit na interaktivní krychli, tak nevěděl, kam aktuálně míří. Pohyboval tedy rukou kolem krychle do té doby, než na ní zamířil a zobrazil se zelený paprsek. Problém byl v tom, že se paprsek zobrazoval pouze ve chvíli, kdy uživatelův ovladač mířil na nějaký objekt, se kterým může interagovat, nebo na zem.

Druhý problém, který také občas nastal, spočíval ve znemožnění manipulace s dílem LW modelu. Toto se objevovalo pouze ve scéně s manipulačním způsobem. Jednalo se o to, že když uživatelův ovladač mířil na zem či interaktivní krychli, tak se zobrazoval zelený paprsek. V kódu aplikace to znamenalo, že se kontroluje, zda uživatel nestiskne tlačítko trigger pro spuštění interakce. Tím pádem se nekontrolovalo, zda uživatel používá tlačítka, která jsou určena k manipulaci s LW modelem. Pokud však uživatel



zamířil ovladačem mimo krychli a zem, tak zelený paprsek zmizel a manipulace s modelem opět fungovala. Případ, kdy nebylo možné s modelem manipulovat lze vidět na obrázku 5.3.



**Obrázek 5.3.** Situace, kdy nebylo možné s modelem manipulovat. Ovladač míří skrz model na zem a zobrazuje se tak zelený paprsek.

Další problém, který se také objevoval, byl spojený s nastavením použitého náhlavního displeje. Na prohlížení byl použit náhlavní displej Oculus Quest 2. Při používání tohoto náhlavního displeje je možné si nastavit, v jak velké oblasti se bude člověk pohybovat. Nastavení je to bezpečnostní, aby uživatel nenarazil například do zdi. Právě ono nastavení oblasti pohybu bylo jádro problému. Problém spočíval v tom, že při spuštění aplikace nebyl do počátku scény umístěn uživatel, ale právě tato oblast. Až v rámci této oblasti byl do scény umístěn uživatel. Tedy uživatel se po spuštění aplikace nenacházel přímo v počátku a ani se nekoukal směrem, kterým by se koukat měl. Znamenalo to, že když se například uživatel nacházel ve scéně s výběrem, tak se celý LW model vykreslil za uživatelem a uživatel se musel nejdříve otočit. Větší problém se však děl ve scéně s manipulačním způsobem prohlížení. V této scéně se kvůli nastavené oblasti pohybu uživatel nacházel například přímo v interakční krychli. Aplikace byla vyvíjena na jiném náhlavním displeji, nicméně bez nastavení oblasti pohybu. Náhlavní displej Oculus nabízí i možnost stacionárního používání (tedy bez oblasti pohybu). Při tomto nastavení by aplikace fungovala správně. Bohužel se na příčinu problému přišlo až po testování.

# Kapitola 6

## Závěr

Cílem práce bylo seznámit se s digitalizovaným Langwailovým modelem a implementovat virtuálně realitní aplikaci na jeho prohlížení. V práci byla popsána analýza již existující aplikací na prohlížení 3D modelů. V rámci prvotního návrhu byly navrženy a implementovány prohlížecké způsoby, které byly následně otestovány s cílovou skupinou uživatelů. Vzhledem k formátu výchozích dat bylo nutné data upravit. Ve výchozím stavu byla data v podstatě nepoužitelná pro aplikaci pro virtuální realitu. Byly proto navrženy a následně provedeny úpravy dat, které výrazně snížily výpočetní náročnost aplikace. S ohledem na zpětnou vazbu z prvotního testování byly navrženy a implementovány nové způsoby prohlížení modelu. Tyto způsoby byly následně otestovány s cílovou skupinou uživatelů. V rámci testování byly vyzorovány problémy aplikace, nicméně vyskytlé problémy nebyly nijak rozsáhlé. Celkový ohlas od uživatelů z testování byl pozitivní. Považuji tak všechny požadavky aplikace za splněné. Rovněž byly splněny všechny body zadání práce. I přes vyskytlé problémy jsem se svou prací spokojen.

### 6.1 Další vývoj

Co se dalšího vývoje týče, tak určitě bude zapotřebí v aplikaci vyřešit problémy, které se vyskytly v testování. Jejich řešení by nemělo být složité.

V průběhu práce bylo plánováno asynchronní načítání modelů do scény. Toto však z časových důvodů nebylo v aplikaci implementováno. Záměr asynchronního načítání byl takový, že ve scéně pro výběr modelu by se uživateli postupně zobrazovaly jednotlivé načtené díly. V aktuálním stavu aplikace se všechny díly zobrazí najednou, avšak pro kratší prodlevě. Použitím asynchronního načítání by tak k této prodlevě nedocházelo.

V procházečím způsobu by bylo možné implementovat popisky. Fungovalo by to tak, že pokud by se uživatel přiblížil k významné budově, tak by se v blízkosti dané budovy objevilo malé okénko. V okénku by bylo v krátkosti napsáno pár informací o dané budově. Jednalo by se o podobné popisky, které byly použity ve VR muzeu The VR Museum of Fine Art (popsáno v kapitole 2.8.3). Popisky by tak uživatele obohatily o další informace. Implementace popisků by v Unity spočívala ve využití komponent Collider. Tyto komponenty by registrovaly přiblížení a vzdálení uživatele od dané budovy a podle toho by zobrazovaly příslušné okno s popisky.

Další věc, kterou bude pravděpodobně nutné implementovat, je způsob vysvětlení ovládání uživateli. Jednalo by se nejspíše o nějakou formu tutoriálu v samotné aplikaci, kde by bylo uživateli jednoduše a srozumitelně vysvětleno, jak se aplikace ovládá. V současném stavu žádný takový způsob v aplikaci implementován není, a tedy je v podstatě nemožné aplikaci používat bez popisu ovládání či osoby, která zná konkrétní ovládání aplikace.

## Literatura

- [1] Kateřina Bečková a Miroslav Fok. *Svědectví Langweilova modelu Prahy*. Schola Ludus Pragensia, 1996. ISBN 80-900668-8-7.
- [2] Bavor V. *Poznámky pana Bavora*. [online]. 2019.  
<https://poznamkypanabavora.wordpress.com/2019/06/09/langweiluv-model-prahy-1/>. Navštíveno 17.5.2022.
- [3] ČVUT. *Langweilův model Prahy bude digitální*. [online]. 2008.  
<https://fel.cvut.cz/cz/vz/media/0801.html>. Navštíveno 18.5.2022.
- [4] Zuzana Strnadová, Kateřina Bečková a Jan Buriánek. *Digitalizace Langweilova modelu Prahy*. [online].  
[https://www.cz-museums.cz/UserFiles/File/muchang%20III/langweil\\_muchang.doc](https://www.cz-museums.cz/UserFiles/File/muchang%20III/langweil_muchang.doc). Navštíveno 17.5.2022.
- [5] David Sedláček, Jan Buriánek a Jiří Žára. *3D Reconstruction Data Set - The Langweil Model of Prague*. International Journal of Heritage in the Digital Era. 2013. 2(2), 195-220. ISSN 2047-4970.
- [6] Jason Jerald. *The VR Book: Human-Centered Design for Virtual Reality*. New York, NY, USA: Association for Computing Machinery and Morgan Claypool, 2015. ISBN 1970001127.
- [7] Acer Inc. *Windows Mixed Reality Headset*. [online]. 2017.  
<https://www.acer.com/ac/en/US/content/model/VD.R05AP.002>. Navštíveno 17.5.2022.
- [8] Wikipedia. *Unity (game engine)*. [online].  
<https://w.wiki/3k2S>. Navštíveno 17.5.2022.
- [9] Simran Kaur Arora. *Unity vs Unreal Engine: Which Game Engine Should You Choose?* [online].  
<https://hackr.io/blog/unity-vs-unreal-engine>. Navštíveno 17.5.2022.
- [10] Blender. *Blender*. [online].  
<https://www.blender.org/about/>. Navštíveno 17.5.2022.
- [11] Khronos. *OpenXR*. [online].  
<https://www.khronos.org/openxr/>. Navštíveno 17.5.2022.
- [12] Ladislav Čmolík. *Representing materials with textures*. [přednáška]. Praha: ČVUT. 25. 11. 2021.
- [13] Meta. *META QUEST 2*. [online].  
<https://store.facebook.com/quest/products/quest-2/>. Navštíveno 18.5.2022.



# Příloha A

## Struktura přiložených souborů

Projekt v herním engine Unity ve verzi 2020.1.6f1. Postup spuštění a nastavení naleznete v příloze B.

- Unity.zip

Ilustrační materiály

- Images.zip

Základní informace

- README.md

Python skript použitý pro přidávání uzlu do materiálu v programu Blender

- insertTextureNodes.py

Data modelů - nutné pro fungování aplikace

- StreamingAssets (dostupné pouze vedoucímu a oponentovi práce na nasdíleném gitlabu)

# Příloha B

## Návod na spuštění

Ke spuštění aplikace budete potřebovat:

- Soubor `Unity.zip`
- Složku `StreamingAssets`
- UnityHub s nainstalovanou verzí Unity 2020.1.6f1
- Steam s nainstalovanou knihovnou SteamVR
- Náhlavní displej s ovladači, který je podporován knihovnou SteamVR (například Acer Windows Mixed Reality AH101 nebo Oculus Quest 2)

Stáhněte si Unity projekt (`Unity.zip`) a složku obsahující asset balíčky modelů (`StreamingAssets`). Soubor `Unity.zip` rozbalte a složku `langweil_vr_viewer` umístěte do libovolného adresáře. Do stejného adresáře umístěte i složku `StreamingAssets`. Ujistěte se, že ve složce `StreamingAssets` jsou asset balíčky modelů (soubory `lw_1_high`, `lw_2_high` atd.). Je nutné, aby se složka `StreamingAssets` nacházela vedle složky `langweil_vr_viewer`.

Otevřete si Unity projekt v Unity editoru ve verzi Unity 2020.1.6f1. Spusťte program Steam a připojte Váš náhlavní displej. Ve SteamVR je nyní potřeba namapovat jednotlivé akce na tlačítka Vašeho ovladače. Toto je nutné, protože při vývoji byl používán jiný Steam účet, než který používáte vy. Do okna pro nastavení akcí se dostanete následovně - v otevřeném projektu v Unity editoru se navigujte takto: Window -> SteamVR Input. V zobrazeném okně sjedte dolů a klikněte na tlačítko „Open binding UI“. V zobrazeném okně klikněte na tlačítko „Edit“ pod nápisem „Current Binding“. Nyní se ujistěte, že máte v horní části okna zvolený set akcí „default“.

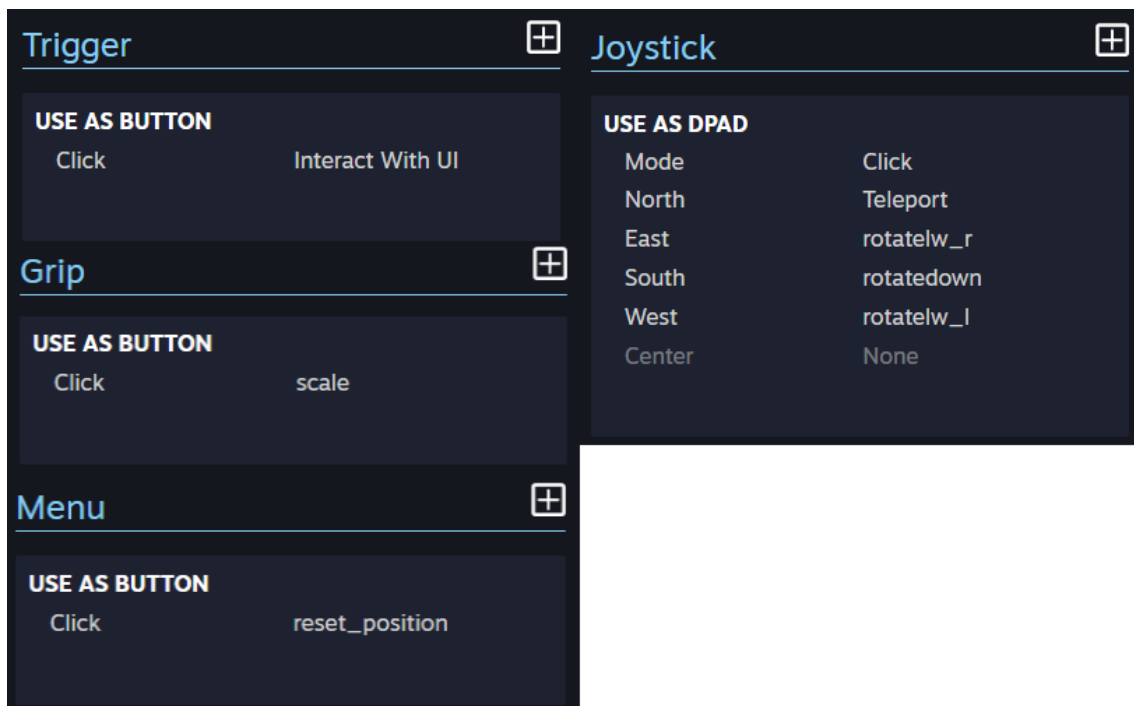
Nyní je zapotřebí nastavit následujících 7 akcí: `Interact With UI`, `scale`, `reset_position`, `Teleport`, `rotatelw_r`, `rotatelw_l`, `rotatedown`. Akce nastavujte se zapnutým režimem „Mirror mode“ (uprostřed obrazovky).

Je zapotřebí provést změny, aby byly akce nastaveny následovně:

- Trigger -> Use as Button -> akce „Interact With UI“
- Grip -> Use as Button -> akce „scale“
- Menu -> Use as Button -> akce „reset\_position“
- Joystick -> Use as DPAD -> pozice North -> akce „Teleport“
- Joystick -> Use as DPAD -> pozice South -> akce „rotatedown“
- Joystick -> Use as DPAD -> pozice West -> akce „rotatelw\_l“
- Joystick -> Use as DPAD -> pozice East -> akce „rotatelw\_r“

Akce „Interact With UI“ a „Teleport“ nejspíše již budou namapované správně. U ostatních akcí budete muset buď změnit akci na daném bindingu, nebo přidat nový binding. Při přidávání volte typ „Button“ (pro tlačítka Trigger, Grip a Menu), případně typ „DPAD“ (pro joystick).

Po namapování jednotlivých akcí by mělo nastavení vypadat tak, jak je ukázáno na následující stránce. Při používání displeje Oculus Quest 2 je možné a pohodlné si akci „reset\_position“ namapovat na jedno z tlačítek A/B/X/Y.



Po namapování akcí vše uložte stisknutím tlačítka „Replace default binding“ v pravé dolní části obrazovky. Poté se můžete vrátit zpět do Unity editoru. Otevřete si scénu „Start“ (scéna se nachází ve složce Scenes) a nyní můžete aplikaci spustit tlačítkem Play v horní části obrazovky.

# Příloha C

## Konkrétní podoba dotazníku

### Prohlížení Langweilova modelu Prahy

1. Už si někdy zkoušel/a virtuální realitu?

Označte jen jednu elipsu.

- Ano  
 Ne

2. Jak se ti to celé líbilo?

Označte jen jednu elipsu.

- 1 2 3 4 5  
Nejlepší      Nejhorší

3. Jak se ti líbilo prohlížení modelu - otáčení a naklápění?

Označte jen jednu elipsu.

- 1 2 3 4 5  
Nejlepší      Nejhorší

4. Jak se ti ovládalo otáčení, naklápění a změna velikosti?

Označte jen jednu elipsu.

- 1 2 3 4 5  
Nejlepší      Nejhorší

5. Jak se ti líbilo procházení zvětšeným městem?

Označte jen jednu elipsu.

- 1 2 3 4 5  
Nejlepší      Nejhorší

6. Jak se ti ovládalo procházení zvětšeným městem?

Označte jen jednu elipsu.

- 1 2 3 4 5  
Nejlepší      Nejhorší

7. Dělal/a se ti při prohlížení špatně?

Označte jen jednu elipsu.

- Ano, hodně  
 Jen trochu  
 Vůbec

8. Napadá tě, co by se dalo zlepšit? Nebo co by se dalo udělat jinak?

\_\_\_\_\_

9. Bylo něco, co se ti nelíbilo? Proč?

\_\_\_\_\_

10. Co se ti líbilo nejvíce?

\_\_\_\_\_

11. Chtěl/a bys mi něco vzkázat?

\_\_\_\_\_



## **Příloha D**

### **Odkazy**

#### **D.1 Gitlab repozitář práce**

- <https://gitlab.fel.cvut.cz/langweil-mmp/vrmodelviewer>

#### **D.2 Ukázkové video**

- <https://youtu.be/ojr6-Z6eU5s>

#### **D.3 SteamVR balíček pro Unity**

- <https://assetstore.unity.com/packages/tools/integration/steamvr-plugin-32647>