



Zadání bakalářské práce

Název:	Prohlížeč 3D modelů ve virtuální nebo smíšené realitě pomocí knihovny A-Frame
Student:	Daria Fedosova
Vedoucí:	Ing. Petr Pauš, Ph.D.
Studijní program:	Informatika
Obor / specializace:	Webové a softwarové inženýrství, zaměření Počítačová grafika
Katedra:	Katedra softwarového inženýrství
Platnost zadání:	do konce letního semestru 2023/2024

Pokyny pro vypracování

Cílem práce je vytvořit prohlížeč 3D modelů pro webové prohlížeče podporující WebXR v knihovně A-Frame. Aplikace by měla být schopna modely ve vhodném formátu zobrazit ve smíšené nebo virtuální realitě na podporovaných zařízeních a doplnit základní uživatelské rozhraní pro ovládání prohlížení.

1. Analyzujte možnosti zobrazování 3D modelů ve webovém prohlížeči.
2. Analyzujte možnosti knihovny A-Frame pro tento účel.
3. Pomocí metod softwarového inženýrství navrhnete prototyp aplikace včetně uživatelského rozhraní.
4. Prototyp implementujte.
5. Provedte vhodná testování.

Bakalářská práce

**PROHLÍŽEČ 3D MODELŮ
VE VIRTUÁLNÍ NEBO
SMÍŠENÉ REALITĚ
POMOCÍ KNIHOVNY
A-FRAME**

Daria Fedosova

Fakulta informačních technologií
Katedra softwarového inženýrství
Vedoucí: Ing. Petr Pauš, Ph.D.
16. května 2024

České vysoké učení technické v Praze
Fakulta informačních technologií

© 2024 Daria Fedosova. Všechna práva vyhrazena.

Tato práce vznikla jako školní dílo na Českém vysokém učení technickém v Praze, Fakultě informačních technologií. Práce je chráněna právními předpisy a mezinárodními úmluvami o právu autorském a právech souvisejících s právem autorským. K jejímu užití, s výjimkou bezúplatných zákonných licencí a nad rámec oprávnění uvedených v Prohlášení, je nezbytný souhlas autora.

Odkaz na tuto práci: Fedosova Daria. *Prohlížeč 3D modelů ve virtuální nebo smíšené realitě pomocí knihovny A-Frame*. Bakalářská práce. České vysoké učení technické v Praze, Fakulta informačních technologií, 2024.

Obsah

Poděkování	vii
Prohlášení	viii
Abstrakt	ix
Seznam zkratk	x
Úvod	1
1 Cíle práce	2
2 3D grafika ve webových prohlížečích	3
2.1 Adobe Flash	3
2.2 WebGL	4
2.3 WebGPU	4
2.4 WebXR	4
2.4.1 Virtuální realita	5
2.4.2 Smíšená realita	5
3 A-Frame	7
3.1 Úvod	7
3.2 Klíčové vlastnosti	7
3.3 Architektura	7
3.4 Podporovaná zařízení	8
3.5 Použití	9
3.6 Výhody a nevýhody	9
4 Návrh prototypu	10
4.1 Architektura	10
4.2 Procesy	10
4.2.1 Otevření modelu	11
4.2.2 Přidání a mazání modelu	13
4.2.2.1 Přidání modelu	13
4.2.2.2 Mazání modelu	14
4.2.3 Základní manipulace	15
4.3 Doménový model	15

4.3.1	Model	16
4.3.2	Autor	16
4.4	Relační datový model	17
4.5	Případy užití	17
4.5.1	Účastníci	18
4.5.2	Funkční požadavky	18
4.5.2.1	F1 – evidence modelů	18
4.5.2.1.1	UC1 – Přidání modelů pomocí URL	19
4.5.2.1.2	UC2 – Mazání modelů ze systému	19
4.5.2.2	F2 – zobrazení modelu	20
4.5.2.2.1	UC3 – Zobrazit model ze systému	20
4.5.2.2.2	UC4 – Zobrazit model pomocí URL	21
4.5.2.3	F3 – manipulací s modelem	21
4.5.2.3.1	UC5 – Rotace modelu	22
4.5.2.3.2	UC6 – Posun modelu	23
4.5.2.3.3	UC7 – Zvětšení nebo zmenšení modelu	23
4.5.3	Nefunkční požadavky	23
4.5.3.1	N1 – responzivní design	23
4.5.3.2	N2 – rozumná doba renderu modelu	24
4.5.3.3	N3 – podpora různých druhů zařízení	24
4.6	Návrh uživatelského rozhraní	24
4.6.1	Analýza uživatelského rozhraní	24
4.6.2	Výběr modelu	25
4.6.3	Prohlížení modelů	25
4.6.3.1	Prohlížení pomocí gest	25
4.6.3.2	Prohlížení modelu pomocí uživatelského rozhraní	25
4.6.3.3	Problém s rozhraními ve VR a AR	26
4.6.4	Prototyp uživatelského rozhraní	26
4.6.4.1	Heuristický průchod	28
5	Implementace	30
5.1	Vytvoření databáze	30
5.2	Vývoj serverové části aplikace	30
5.3	Vývoj klientské části aplikace	31
5.3.1	Vývoj interakcí s modelem	31
5.3.1.1	Obecné interakce	31
5.3.1.2	Rotace	31
5.3.1.3	Posun	32
5.3.1.4	Zoomování	33
5.3.2	Interakce v XR	34
5.3.2.1	Změna funkce posunu	34
5.3.3	Vývoj uživatelského rozhraní	35
5.3.4	Použití aplikace	36

6 Testování	41
6.1 Manuální testování	41
6.2 Uživatelské testování	41
7 Hodnocení výsledků	43
7.1 Návrh prototypu	43
7.2 Implementace	43
7.3 Testování	44
8 Závěr	45
Obsah příloh	49

Seznam obrázků

4.1	Diagram procesu otevření modelu	12
4.2	Diagram procesu přidání modelu	13
4.3	Diagram procesu mazání modelu	14
4.4	Diagram procesu manipulace s modelem	15
4.5	Doménový model	16
4.6	Relační datový model	17
4.7	F1 – evidence modelů	19
4.8	F2 – zobrazení modelu	20
4.9	F3 – manipulací s modelem	22
4.10	Prototyp rozhraní pro mobilní zařízení	27
4.11	Prototyp rozhraní pro počítače	28
5.1	Upravené rozhraní	36
5.2	Domovská stránka	37
5.3	Okno s informacemi o modelu	38
5.4	Tutoriál	39
5.5	Stránka prohlížení modelu	40

Seznam tabulek

4.1	Popis atributů entity “Model”	16
4.2	Popis atributů entity “Autor”	17
6.1	Výsledky průzkumu	42

Seznam výpisů kódu

5.1	Výpočet změny souřadnic ukazatele	32
5.2	Předání nového úhlu otáčení objektu	32
5.3	Rozpoznávání dotyku na mobilních zařízeních	32
5.4	Předání nových hodnot polohy objektu	33
5.5	Funkce přiblížení a oddálení pro myš	33
5.6	Funkce přiblížení a oddálení pro mobilní zařízení	33
5.7	Upravená funkce pro posun	34

Chtěla bych poděkovat především svému vedoucímu Ing. Petru Paušovi, Ph.D. za neocenitelnou pomoc při zpracování mé práce, za její čas a možnost častých osobních konzultací. Děkuji také mé rodině a přátelům, kteří mi byli podporou.

Prohlášení

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.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisů, zejména skutečnost, že České vysoké učení technické v Praze má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 citovaného zákona.

V Praze dne 16. května 2024

Abstrakt

Cílem této bakalářské práce je prozkoumat knihovnu A-Frame a navrhnout prototyp webové aplikace pro prohlížení 3D modelů ve smíšené a virtuální realitě s možností interakce. Práce se zaměřuje na vývoj této aplikace včetně uživatelského rozhraní.

Klíčová slova A-Frame, virtuální realita, smíšená realita, rozšířená realita, webová aplikace, JavaScript

Abstract

The main goal of this bachelor thesis is to explore the A-Frame library and to design a prototype web application for viewing 3D models in augmented or virtual reality with the possibility of interactions. The thesis focuses on the implementation of this application including the user interface.

Keywords A-Frame, virtual reality, augmented reality, extended reality, web application, JavaScript

Seznam zkratek

VR	Virtual reality (Virtualní realita)
AR	Augmented reality (Smíšená realita)
GLSL	OpenGL Shading Language (Jazyk shaderů OpenGL)
API	Application programming interface (Aplikační programové rozhraní)
HTML	HyperText Markup Language (Hypertextový značkovací jazyk)
DOM	Document Object Mode (Objektový model dokumentu)
ECS	Entity-component-system (Systém entita-komponenta)
XR	Extended reality (Rozšířená realita)
MVC	Model-view-controller (Model-pohled-řadič)
URL	Uniform Resource Locator (Jednotný lokátor zdroje)
UML	Unified Modeling Language (Jednotný modelovací jazyk)
UI	User interface (Uživatelské rozhraní)

Úvod

V současné době jsou smíšená (AR) a virtuální realita (VR) již nejen výmyslem, ale technologií, kterou může vyzkoušet skoro každý pomocí svého mobilního zařízení. V roce 2016 byla na trhu mobilních her uvedena Pokémon GO využívající smíšenou realitu a dosáhla více než 600 milionů stažení [1]. Není to jediný příklad úspěšných zábavných aplikací využívajících technologie smíšené reality: Half Life: Alyx nebo Beat Saber jsou také velmi známé hry. Ale možnosti použití smíšené reality nekončí v videoherním průmyslu. Lze najít příklady integrace digitálního a fyzického světa v hudebních představeních, výstavách a festivalech. Probíhající Signal Festival v Praze již několik let využívá smíšenou realitu jako doplňkový zážitek pro své návštěvníky. Zcela novou variantu technologie rozšířené reality představuje vydání brýlí Apple Vision Pro v únoru roku 2024, které umožňují sledovat a natáčet videa, používat messenger a dokonce měnit scény kolem člověka.

Jedním problémem většiny aplikací s využitím VR nebo AR je nutnost jejich instalace na zařízení, což zabírá vnitřní paměť. Často je aplikace smazána, protože po jejím použití, například na festivalu, již nemusí být potřebná. Ale co pokud potřebujeme na to jenom webový prohlížeč, který je na většině operačních systémů předinstalován? Zůstává jenom ověřit požadavky na hardwarové vybavení, otevřít stránku na webu a vyzkoušet moderní technologie počítačové grafiky.

Výstup této práce pomůže uživatelům, kteří se chtějí seznámit s možnostmi smíšené a virtuální reality pomocí modelů, které byly vytvořeny studenty FIT ČVUT během studia.

V práci se zabýváme analýzou možností zobrazení 3D modelů ve webových prohlížečích a zkoumáním knihovny A-Frame a jejích možností při zobrazení modelů ve virtuální nebo smíšené realitě. Představen bude návrh webové aplikace včetně návrhu jejího uživatelského rozhraní a implementace webových stránek, které umožní prohlížet 3D modely v režimech VR nebo AR.



Kapitola 1

Cíle práce

Hlavním cílem této bakalářské práce je implementace webové aplikace, která umožní prohlížení 3D modelů ve virtuální nebo smíšené realitě a provádění základních manipulací s modelem.

Zadání je rozděleno na menší úkoly. Prvním úkolem je analyzovat schopnost webových prohlížečů zobrazovat 3D grafiku. Dalším úkolem je analýza funkcí frameworku A-Frame pro prohlížení 3D modelů. Následně je třeba s využitím metod softwarového inženýrství navrhnout prototyp webové aplikace, včetně vytvoření uživatelského rozhraní. Navržený prototyp aplikace je nutné implementovat a otestovat. Výsledky práce je pak nezbytné analyzovat pro další vývoj projektu.

3D grafika ve webových prohlížečích

Tato kapitola je věnována analýze a výzkumu možností zobrazení 3D grafiky ve webových prohlížečích.

2.1 Adobe Flash

Historie grafiky ve webu se začala v roce 1996, kdy firma FutureWave Software vypustila Macromedia Flash (bývalý FutureSplash) – software zabývající se zobrazením interaktivních animací, videí a ostatních multimediálních prostředků ve webu [2]. Tento program byl později zakoupen firmou Macromedia a ještě později firmou Adobe. Aplikace dostala jméno, kterou dobře znají uživatelé webových prohlížečů – Adobe Flash. S vývojem této technologie na internetu vzniklo obrovské množství interaktivních webových stránek, animací a her.

Na začátku své existence neměl Flash možnost zobrazovat 3D grafiku. V roce 2008 byl vydán Adobe Flash 10, který tuto možnost poskytl. Bylo však možné používat pouze pseudo-3D grafiku neboli 2.5D grafiku. To je pouze způsob, jak vytvořit iluzi 3D grafiky pomocí 3D transformací na 2D objektech. Tímto způsobem bylo možné dosáhnout dojmu perspektivy. Adobe Flash neuměl pracovat přímo s 3D objekty [3].

S rozšířením trhu mobilních zařízení, Adobe Flash začal ztrácet svou popularitu, protože telefony tento software nepodporovaly. Přesto měl Adobe Flash velký vliv na design webových stránek a vytvořil základ jejich moderního vzhledu.

2.2 WebGL

Jedním z nejrozšířenějších JavaScript API pro vykreslování 3D grafiky je dnes WebGL, vyvinuté Khronos Group. Jak je uvedeno na domovské stránce společnosti, WebGL je multiplatformní, bezplatný otevřený webový standard pro nízkourovňovou 3D grafiku. API je založeno na OpenGL Embedded System a používá programovací jazyk pro psaní shaderů GLSL – OpenGL Shading Language [4]. WebGL umožňuje vytvářet 3D grafiku ve webových prohlížečích bez zásuvných modulů a k výpočtům využívá GPU zařízení.

Podle poskytnuté dokumentace je WebGL v současné době podporován všemi webovými prohlížeči, včetně mobilních prohlížečů. Jedinou výjimkou jsou některé specifické funkce, které nemusí být v jednotlivých prohlížečích přítomne [5].

WebGL pracuje přímo s grafickým procesorem zařízení a práce s ním je poměrně složitá. Protože je API nízkourovňové, práce s ním vyžaduje určitou úroveň znalosti počítačové grafiky. V současné době však existuje poměrně velký počet knihoven, které usnadňují použití WebGL. Patří sem například three.js, Babylon.js, PlayCanvas. Herní engine Unity také umožňuje vývoj aplikací založených na WebGL.

2.3 WebGPU

Dalším API pro zobrazování 3D grafiky na webu je WebGPU, které je však stále ve vývoji. Stejně jako WebGL je webovým standardem a je vyvíjen ve spolupráci několika společností, například W3C, Google, Apple Inc. a Mozilla [6]. WebGPU je nízkourovňové rozhraní, které používá vlastní jazyk shaderů, WebGPU Shading Language [7].

Ne všechny prohlížeče a systémy podporují vyvíjený standard. V systémech Android a Linux není podpora. Proces integrace WebGPU v prohlížečích, jako je Firefox a Safari, stále probíhá [8].

Stejně jako v případě WebGL, lze aplikace vyvíjet přímo pomocí WebGPU nebo použít knihovny. V současné době je jedinou knihovnou, která plně podporuje WebGPU, Babylon.js [8].

2.4 WebXR

WebXR, přesněji WebXR Device API, je standard určený k zajištění přístupu ke vstupům a výstupům zařízení virtuální a smíšené reality. Toto API umožňuje vyvíjet webové aplikace pomocí zmíněných technologií [9]. WebXR Device API spravuje výběr výstupních zařízení, vykresluje 3D scénu na vybrané zařízení a spravuje vektory pohybu vytvořené pomocí vstupních ovladačů [10]. K renderování obrázků se používají funkce WebGL.

Toto API nepodporují všechna zařízení. Dokumentace WebXR na githubu obsahuje neúplný seznam podporovaných zařízení. Kromě toho, že zařízení musí podporovat tento standard, je vyžadován také podporovaný webový prohlížeč. V současné době jsou podporovány následující prohlížeče: Chrome, Edge, Opera, Samsung Internet [10].

2.4.1 Virtuální realita

Původ pojmu “virtuální realita” (v angličtině “virtual reality”) je v mnoha zdrojích spojován se jménem Jaron Laniera, jednoho ze zakladatelů společnosti VPL Research. Tato společnost byla jednou z prvních, která vyvíjela a prodávala zařízení související s VR.

Co je však virtuální realita? V současné době neexistuje žádná jednotná nebo oficiální definice, kterou by bylo možné použít. V Encyklopedii multimédií je virtuální realita technologie, která poskytuje téměř reálné a/nebo věrohodné zážitky syntetickým nebo virtuálním způsobem [11].

Virtuální realita se vyznačuje několika klíčovými vlastnostmi:

- virtuální svět – scéna, ve které se odehrávají akce virtuální reality. Patří sem skripty, které definují chování objektů nebo postav, samotné objekty a vizuální složka;
- imerze – uživatel je ponořen do virtuálního světa nejen příběhem a obrazem, ale i fyzicky. Brýle pro virtuální realitu poskytují nejen vizuální, ale také zvukové ponoření;
- smyslová zpětná vazba – při používání brýle VR se svět kolem uživatele mění v závislosti na jeho činnostech. Například kamera se ovládá nakláněním a otáčením hlavy uživatele;
- interaktivita – pro vytvoření pohlcujícího zážitku musí vytvořený virtuální svět reagovat na akce uživatele. Například možnost zvednout nebo položit předmět nebo se pohybovat v prostoru. [12]

Virtuální realita je neoddělitelná s vybavením – speciálními brýlemi a helmami. V současné době existují brýle pro virtuální realitu pro počítače, herní konzole a dokonce i mobilní telefony.

Google Cardboards vydané společností Google a jejich ekvivalenty zpřístupňují virtuální realitu mnoha lidem. Je třeba poznamenat, že Cardboard sám o sobě není hardware, ale pouze doplněk. Veškeré výpočty se provádějí pomocí hardwaru mobilního telefonu.

2.4.2 Smíšená realita

Rozšířená realita je systém, který vylepšuje reálný svět tím, že do něj vkládá počítačem generované informace [11]. Rozšířenou realitu lze označit jako protiklad virtuální reality. Zatímco virtuální realita přenesení uživatele do fiktivního

světa, smíšená realita vkládá digitální obsah do pohledu uživatele na reálný svět [13].

Celkově lze konstatovat, že vlastnosti virtuální reality platí i pro smíšenou realitu. Rozdíl spočívá v tom, jak přesně jsou implementovány:

- virtuální svět – v případě smíšené reality je virtuální svět zasazen do reálného světa pomocí sledování okolních objektů a dalších technologií, které jsou rysem smíšené reality;
- imerze – můžeme říci, že uživatel je, stejně jako v případě virtuální reality, přenesen do vymyšleného světa. V případě smíšené reality však uživatel není “odříznut” od reality;
- smyslová zpětná vazba – ačkoli u smíšené reality neexistuje stejný efekt jako u brýlí pro virtuální realitu, svět se stále mění na základě změn polohy kamery;
- interaktivita – tato funkce je zcela stejná pro virtuální i smíšenou realitu.

K používání smíšené reality dnes stačí pouze mobilní telefon nebo počítač s připojenou kamerou. Je třeba poznamenat, že ne všechny moderní telefony podporují smíšenou realitu, protože tato technologie vyžaduje určitý hardwarový výkon.

Kapitola 3

A-Frame

Tato kapitola je věnována knihovně A-Frame, včetně popisu pojmů, struktury a hlavních součástí.

3.1 Úvod

A-Frame je webový framework založený na HTML, který umožňuje vytvářet interaktivní 3D scény včetně využití virtuální a smíšené reality. Původně byl projekt vyvíjen pouze pro VR, protože v té době neexistoval žádný standard WebXR, ale od verze 1.0.0 začal framework podporovat AR [14].

3.2 Klíčové vlastnosti

Hlavní vlastností A-Frame je to, že pracuje s jazykem HTML a objektovým modelem DOM (Document Object Model). Díky tomu framework umožňuje kompatibilitu s většinou knihoven a dalších nástrojů pro vývoj webových aplikací. [15]

Další důležitou vlastností A-Frame je, že byl vyvinut na základě jiné knihovny pro vykreslování 3D grafiky – three.js. Díky tomuto faktu při vývoji aplikace pomocí A-Frame je možné plně využít i funkcionality three.js.

3.3 Architektura

Nejjednodušší formou objektů v A-Frame jsou primitiva – elementy HTML, které obsahují základní informace o 3D objektech, jako jsou geometrie, barva, poloha atd. Příkladem takových objektů může být krychle, koule, kamera, rovina a další. Všechny tyto prvky jsou interně elementy `<a-entity>` s již nastavenými parametry. To znamená, že ve skutečnosti jsou všechny stejným typem objektu a pro všechny platí stejná pravidla použití. Ke všem entitám

v A-Frame je možné přidávat komponenty. Všechny tyto skutečnosti vedou k popisu architektury, na které jsou A-Frame a three.js postaveny. [15]

Tento framework využívá entity-component-system neboli ECS. Jedná se o návrhový vzor, který se hodí především pro použití ve 3D grafice a při vývoji her [16]. Podstatou této architektury je způsob oddělení jednotlivých prvků kódu na základě účelu.

Entita je základním blokem celé architektury a představuje všechny objekty, které lze nějakým způsobem popsat pomocí nějakých atributů. Ale samotná entita nemá žádné vlastnosti a často je reprezentována pouze indexem, aby bylo možné objekty rozlišit. Aby bylo možné chránit jednotlivé stavy nebo vlastnosti, které jsou důležité pro systém, existuje komponenta. Komponenty označují entitu jako entitu s tímto konkrétním aspektem. Poslední část architektury, systém, definuje logiku, která pracuje s komponentami. Přímo v systému jsou popsány metody pro práci s různými typy komponent. [17] [18]

3.4 Podporovaná zařízení

A-Frame nabízí podporu pro širokou škálu zařízení, avšak s určitými komplikacemi a omezeními. Podle dokumentace frameworku jsou ve verzi 1.5.0 podporována následující VR zařízení:

- HTC Vive,
- Oculus Rift,
- Oculus Rift 2,
- Oculus Rift 3,
- Oculus Go,
- Valve Index,
- Vive Focus. [19]

Co se týká podpory webových prohlížečů, aplikace implementované pomocí A-Frame fungují ve všech, které realizují specifikaci WebXR a 3D grafiku [19]. Na počítačích nebude fungovat režim XR pouze ve dvou populárních webových prohlížečích: Firefox a Safari. Na mobilních zařízeních je to podobně, není v tuto chvíli podpora webových prohlížečů Firefox a WebView pro Android zařízení a Safari pro Apple [10].

Možnost provozovat smíšenou realitu na mobilních zařízeních je však stále do značné míry závislá na hardwaru, je nutná instalace ARCore. Nejúplnější seznam zařízení, která tuto funkci podporují, lze nalézt v dokumentaci společnosti Google [20].

3.5 Použití

Závěrem můžeme konstatovat, že A-Frame lze použít jako nástroj pro tvorbu her a aplikací ve virtuální a smíšené realitě bez nutnosti jejich instalace, protože framework je určen pro vývoj webových aplikací. Samotný framework je sice poněkud omezený, ale přístup k funkcím knihovny three.js jeho funkčnost značně rozšiřuje.

Existuje poměrně dost příkladů použití tohoto frameworku. Například NASA použila A-Frame k vytvoření virtuální procházky po Marsu, natočené pomocí roveru Curiosity [21]. Evropská organizace pro jaderný výzkum použila A-Frame pro virtuální prohlídku Velkého hadronového urychlovače [22].

3.6 Výhody a nevýhody

Hlavní výhodou práce s A-Frame je snadné použití. Základní aplikace nepotřebují znalost jazyka JavaScript a nevyžadují rozsáhlé znalosti programování. Stačí mít jen trochu znalostí o tom, jak se vytvářejí stránky pomocí jazyka HTML, a stačí použít připravené elementy. Aplikace napsané pomocí A-Frame jsou multiplatformní a fungují na počítačích, mobilních zařízeních i setech VR. A-Frame podporuje práci s dalšími knihovnami a frameworky, jako je například three.js nebo React. Existuje mnoho rozšíření napsaných speciálně pro A-Frame.

Mezi nevýhody práce s A-Frame patří některá výkonnostní omezení. Načítání složitých scén může při omezených zdrojích zařízení trvat poměrně dlouho. Samotný A-Frame nemá mnoho funkcí. Pokud je potřeba psát komplikovanější aplikace, je nutná buď integrace s jinými knihovnami, což vyžaduje určitou úroveň znalostí práce s nimi, nebo vlastní psaní funkcí, což vyžaduje určité programátorské dovednosti.

Návrh prototypu

Tato kapitola se věnuje analýze požadavků kladených na webovou aplikaci, popisu vybrané architektury, funkčnosti a procesů pomocí diagramu UML a databázového schématu.

4.1 Architektura

Vzhledem k tomu, že aplikace není dostatečně rozsáhlá, nemá specifickou architekturu, ale byla inspirována principy používanými v architektuře MVC (model-view-controller). Hlavními principy této architektury je oddělení objektů, které nesou informace, od prezentace nebo uživatelského rozhraní a oddělení řadičů od prezentace [23].

Pro implementaci serverové části aplikace bylo navrženo použití Node.js a express.js. Node.js se také používá ke správě všech balíčků pomocí integrovaného systému npm. Jako databáze byla zvolena MySQL. K realizaci zobrazení 3D grafiky obecně a v režimu XR podle zadání byl zvolen A-Frame. Také pro potřeby aplikace je nutné zapojit další knihovny pro nastavení serveru takové jako https a fs pro načtení klíčů a certifikátů. Navíc bude použit body parser pro práci s JSON formáty.

4.2 Procesy

Tato část analýzy se věnuje popisu všech procesů, které musí vykonávat aplikace. Podle zadání práce by aplikace měla vyplňovat následující funkce: nějakým způsobem otevírat model ve vhodném formátu a provádět s ním základní manipulace. V dokumentaci knihovny A-Frame je uvedeno, jaké formáty lze použít, ale nejvhodnějším je glTF [24].

Mezi základní manipulace s 3D objekty patří:

- rotace,

- posun,
- zvětšení,
- zmenšení.

Manipulace by měly fungovat v režimu AR a VR.

Existuje několik způsobů, jak model otevřít. Například, uživatel může vybrat 3D objekt z nabídky, otevřít ho pomocí URL nebo ze svého zařízení. Je třeba poznamenat, že v případě výběru objektu z nabídky je lepší mít možnost přidávat nové objekty a mazat staré.

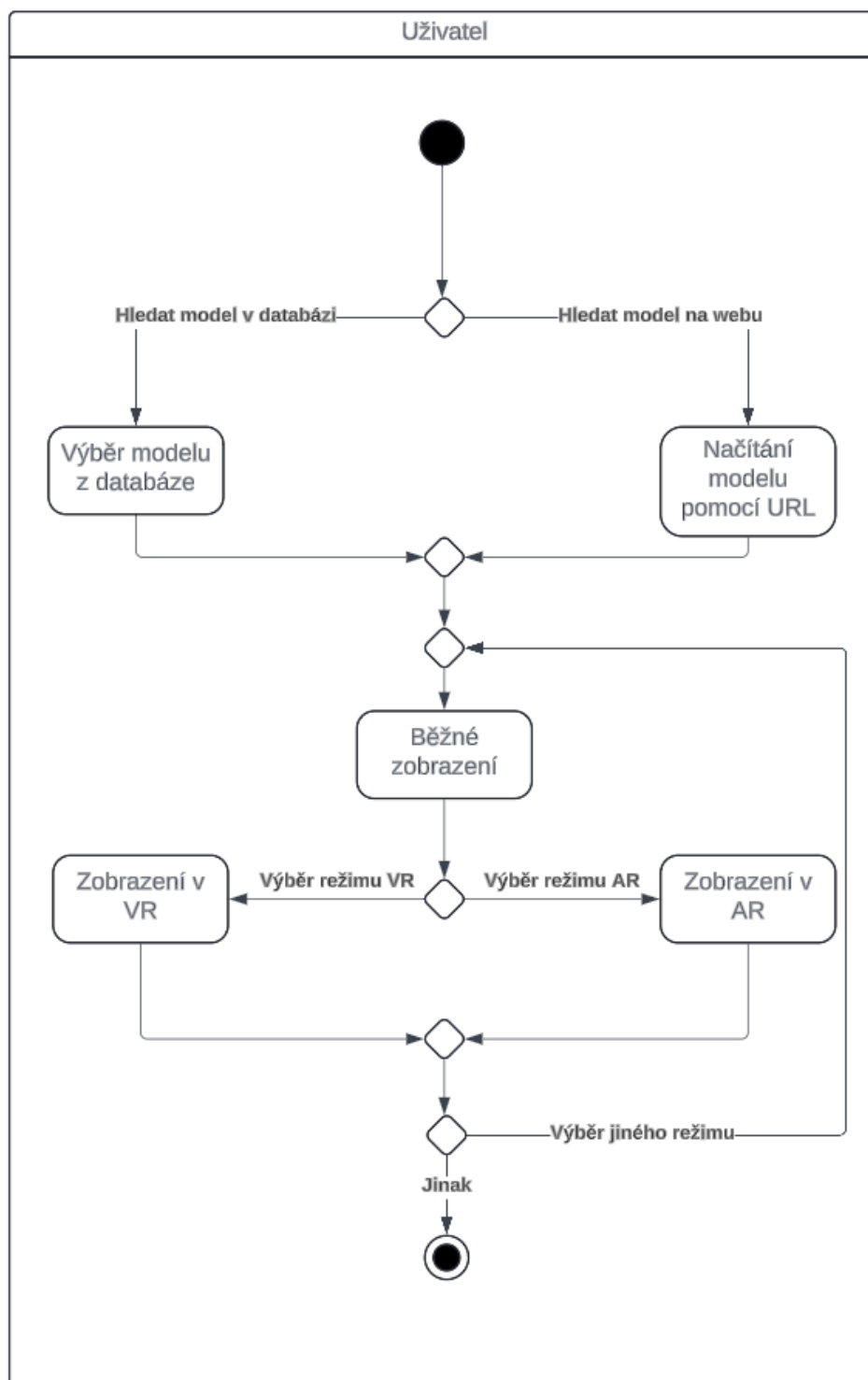
Tak byly identifikovány minimální požadavky kladené na aplikace. Z toho můžeme odnést následující procesy:

- otevření modelu z databáze nebo pomocí URL,
- přidání a mazání modelu,
- provedení základních manipulací s modelem.

4.2.1 Otevření modelu

Otevření modelu je základním procesem, bez kterého není možné v práci dále pokračovat. Umožní uživatelům vybrat objekt a prohlížet ho v tom režimu, který je nejvíce zajímá: běžné prohlížení, režim virtuální reality nebo režim smíšené reality.

Při práci s A-Frame se model vždy otevře v běžném režimu. Potom se uživatel může přepnout na VR nebo AR režim. Možnost použití určitého režimu závisí na zařízení uživatele. Příslušný UML diagram je na obrázku číslo 4.1.



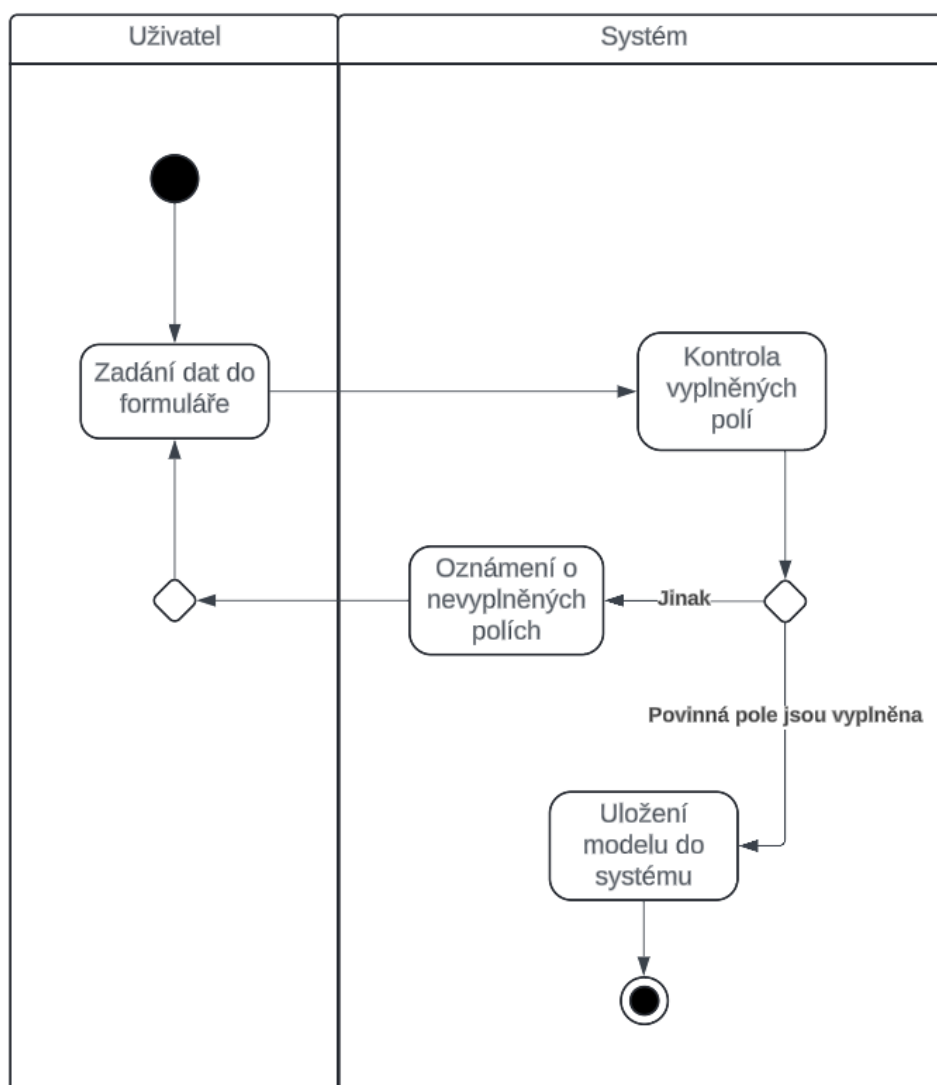
■ Obrázek 4.1 Diagram procesu otevření modelu

4.2.2 Přidání a mazání modelu

V případě práce s databází je nutné mít procesy, které povolují obnovovat informace v ní ukládané. Pro tyto účely musí aplikace přidávat a mazat modely.

4.2.2.1 Přidání modelu

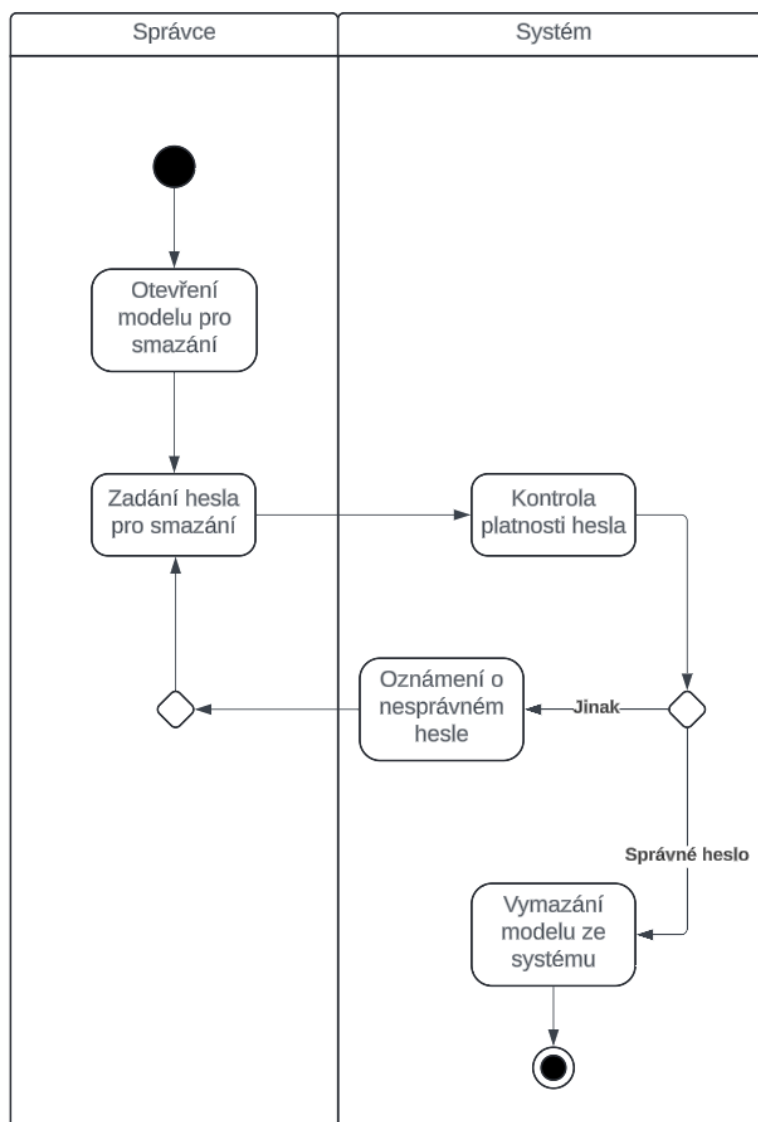
Pro přidání modelu do databáze uživatel musí vyplnit formulář. Systém ověří, zda všechna povinná pole jsou vyplněna. Pokud ano, model bude odeslán do databáze. Jinak systém oznámí uživatele o tom, že nebyla vyplněna všechna povinná pole. Příslušný UML diagram je na obrázku 4.2.



■ **Obrázek 4.2** Diagram procesu přidání modelu

4.2.2.2 Mazání modelu

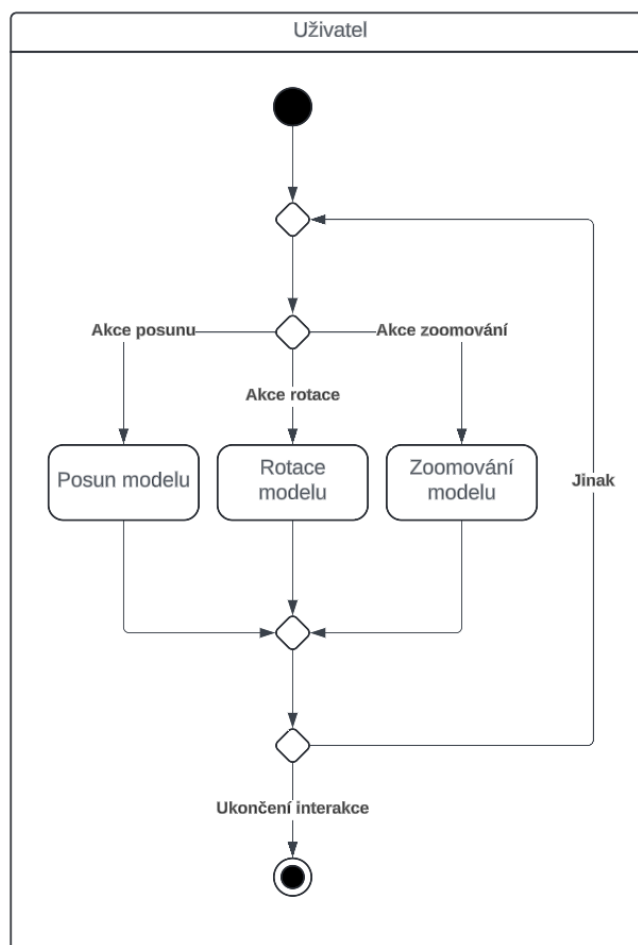
Správce otevře model, který je potřeba smazat. Pro mazání modelu je třeba znát heslo správce. Pokud zadané heslo bylo správné, model bude smazán z databáze. Jinak systém oznámí správce, že heslo není správné. Diagram mazání modelu je vidět na obrázku 4.3.



■ Obrázek 4.3 Diagram procesu mazání modelu

4.2.3 Základní manipulace

Po výběru modelu by měl uživatel být schopen provádět základní manipulace: rotace, posun a zvětšení nebo zmenšení. Uživatel může s objektem provádět interakci v kterémkoli ze tří režimů: běžném, VR nebo AR. Diagram se základními manipulací je vidět na obrázku 4.4.



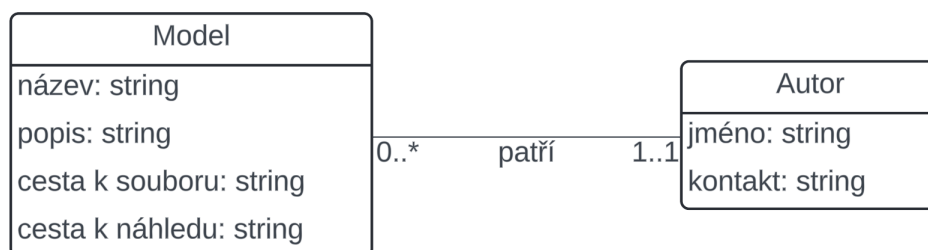
■ **Obrázek 4.4** Diagram procesu manipulace s modelem

4.3 Doménový model

Tato část práce popisuje doménový model navržené aplikace. Budou zde detailně popsány třídy pro aplikaci, aby bylo jasné, jaké objekty a informace je třeba uložit.

Hlavním objektem je tedy model. Každý model má svého autora nebo někoho, kdo má na daný model práva. Proto potřebujeme ukládat tyto dva

objekty. Celý diagram doménového modelu je na obrázku 4.5.



■ **Obrázek 4.5** Doménový model

4.3.1 Model

Třída “Model” reprezentuje 3D objekt v rámci aplikace. Každá instance této třídy definuje konkrétní 3D entitu. Modely slouží jako základ pro vizuální reprezentace a interaktivní prvky. Umožňují uživatelům aplikace otestovat pohlcující zážitky virtuální a smíšené reality.

■ **Tabulka 4.1** Popis atributů entity “Model”

Název atributu	Popis
Název	Název modelu.
Popis	Krátký popis obsahující základní informace o modelu, například vizuální popis, počet vrcholů nebo polygonu atd.
Cesta k souboru	Cesta nebo URL k souboru obsahujícímu geometrické, texturní a behaviorální atributy modelu.
Cesta k náhledu	Cesta nebo URL k souboru obsahujícímu obrázek modelu.

4.3.2 Autor

Třída “Autor” reprezentuje tvůrce nebo přispěvatele 3D modelu. Každá instance této třídy obsahuje informace o jednotlivce nebo společnosti, která stojí za vytvořením příslušného modelu.

■ **Tabulka 4.2** Popis atributů entity “Autor”

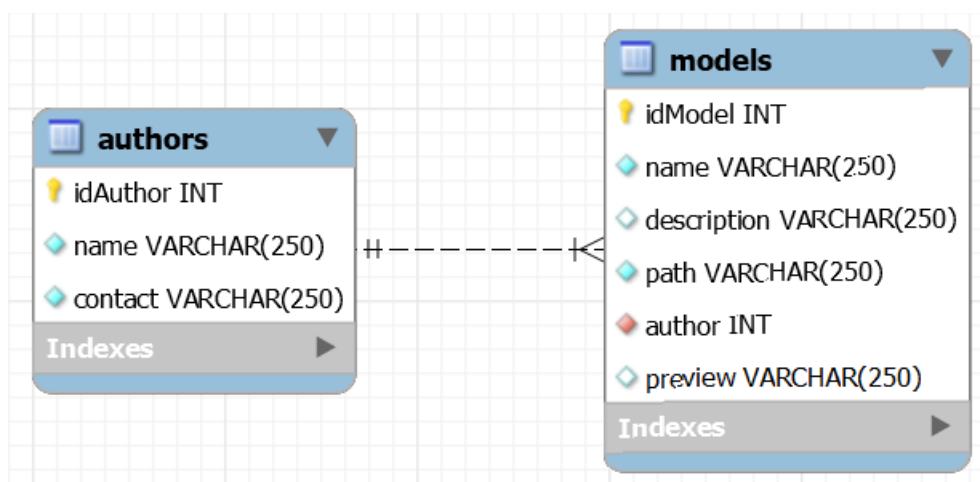
Název atributu	Popis
Jméno	Jméno osoby nebo název společnosti, která je autorem modelu.
Kontakt	Kontaktní údaje (například e-mail nebo personální webové stránky) pro komunikace s autorem modelu.

4.4 Relační datový model

Tato část práce se zabývá popisem databázového modelu. Relační datový model byl navržen na základě doménového modelu s drobnými úpravami pro funkčnost budoucí databáze, jako například cizí klíče pro spojení entit. Momentálně se skládá ze dvou entit:

1. Model – entita znázorňující v systému 3D model.
2. Autor – entita znázorňující v systému autora 3D modelu.

Na obrázku 4.6 je podrobně zobrazena relační schéma vytvořené databáze.



■ **Obrázek 4.6** Relační datový model

4.5 Případy užití

V této části bude prozkoumán model případu užití pro prohlížeč 3D modelů a bude poskytnut popis klíčových případů užití, aktéři, interakci a závislosti, které společně definují provozní tok a chování prohlížeče 3D modelů.

4.5.1 Účastníci

Předpokládá se, že aplikaci budou používat dva typy rolí: uživatel a správce.

Uživatel je osoba, která bude používat hlavní funkce aplikace. To znamená prohlížet 3D modely v různých režimech, provádět s nimi interakci, načítat vlastní modely a přidávat nové do databáze. Uživatel se nemusí registrovat, aby mohl aplikaci používat.

Role správce bude spočívat ve sledování obsahu databáze modelů, její aktualizaci přidáváním nových modelů a odstraněním nevhodných modelů.

4.5.2 Funkční požadavky

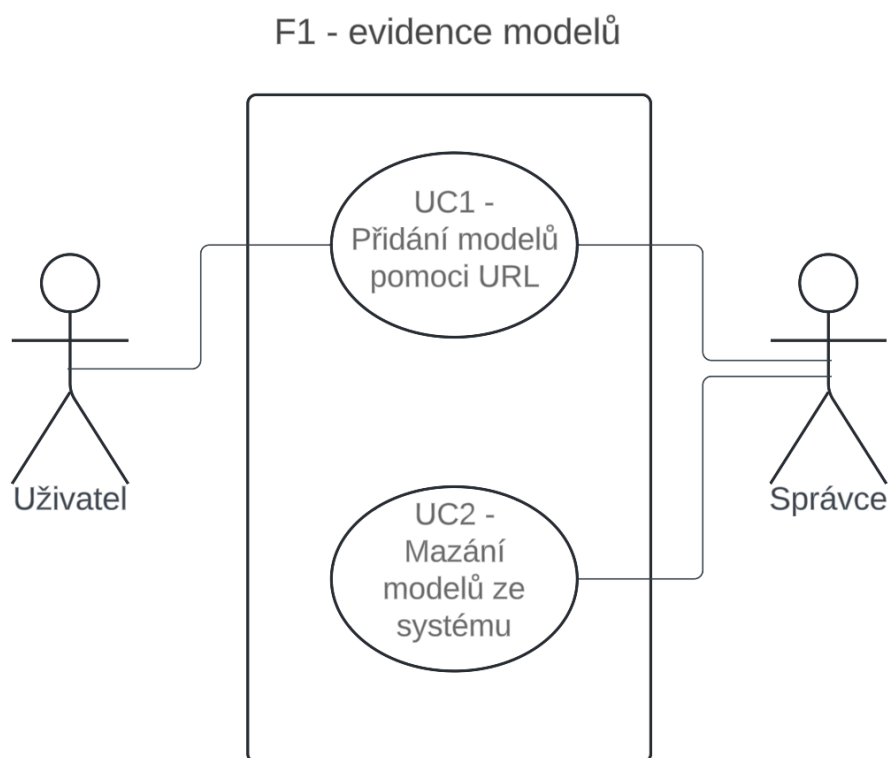
Než budou popsány případy užití aplikace, je nezbytné popsat funkce, které má aplikace vykonávat. Mezi funkční požadavky kladené na systém patří:

- F1 – evidence modelů,
- F2 – zobrazení modelu,
- F3 – manipulací s modelem.

V následujících částech budou podrobně popsány jednotlivé funkční požadavky a jejich případy užití.

4.5.2.1 F1 – evidence modelů

Následující část práce věnována popisu všech funkcností souvisejících s evidencí modelů v prohlížeči. Obrázek 4.7 ukazuje všechny případy užití pro evidence modelů.



■ **Obrázek 4.7** F1 – evidence modelů

4.5.2.1.1 UC1 – Přidání modelů pomocí URL

Umožňuje přidávat nové modely do databáze pomocí URL každému uživateli bez ohledu na roli. K provedení této akce není nutné žádné ověřování ani zvláštní oprávnění.

1. Uživatel nebo správce vybere v rozhraní systému možnost “Přidat nový model”.
2. Systém zobrazí formulář, který umožňuje vyplnit informace potřebné k přidání modelu do databáze.
3. Uživatel nebo správce vyplní povinná pole.
4. Systém zkontroluje, zda jsou vyplněna všechna povinná pole.
5. Systém informace zpracuje.
6. Nový model je přidán do databáze.

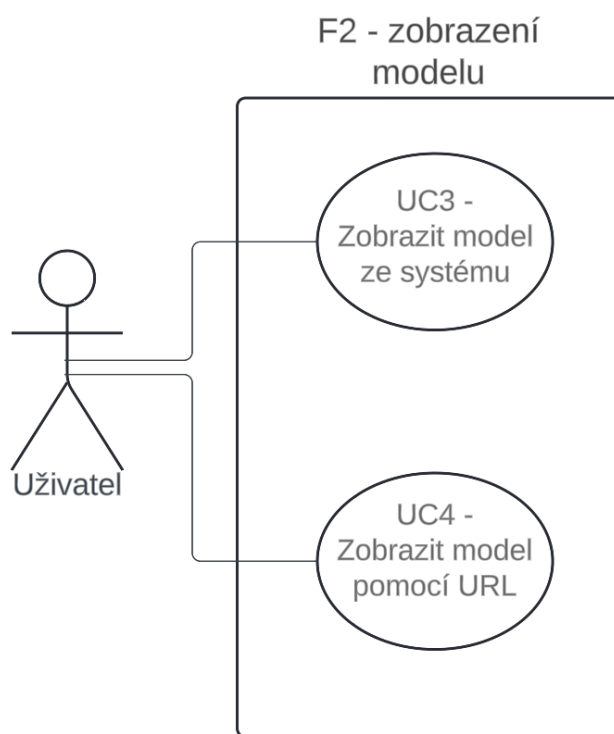
4.5.2.1.2 UC2 – Mazání modelů ze systému

Umožňuje správci odstranit modely z databáze. Pro smazání modelu je nutné znát heslo správce.

1. Správce vybere v rozhraní systému možnost “Odstranit model”.
2. Systém zobrazí formulář, který umožňuje zadat heslo správce.
3. Spávce zadá heslo.
4. Systém zkontroluje, zda je heslo správné. Pokud bylo heslo zadáno nesprávně, systém na to upozorní a vyzve k opětovnému zadání hesla.
5. Model je z databáze odstraněn.

4.5.2.2 F2 – zobrazení modelu

Následující část práce věnována popisu všech funkčností souvisejících s zobrazením modelu v prohlížeči. Obrázek 4.8 ukazuje všechny případy užití pro zobrazení modelu.



■ **Obrázek 4.8** F2 – zobrazení modelu

4.5.2.2.1 UC3 – Zobrazit model ze systému

Umožňuje začít prohlížet 3D model z databáze. Hlavním účastníkem je uživatel.

1. Uživatel vybere ze seznamu model, který ho zajímá.

2. Uživatel vybere v rozhraní systému možnost “Otevřít model”.
3. Systém otevře stránku pro zobrazení vybraného modelu.
4. Uživatel si vybere režim zobrazení: běžný, VR nebo AR.
5. Model je zobrazen.

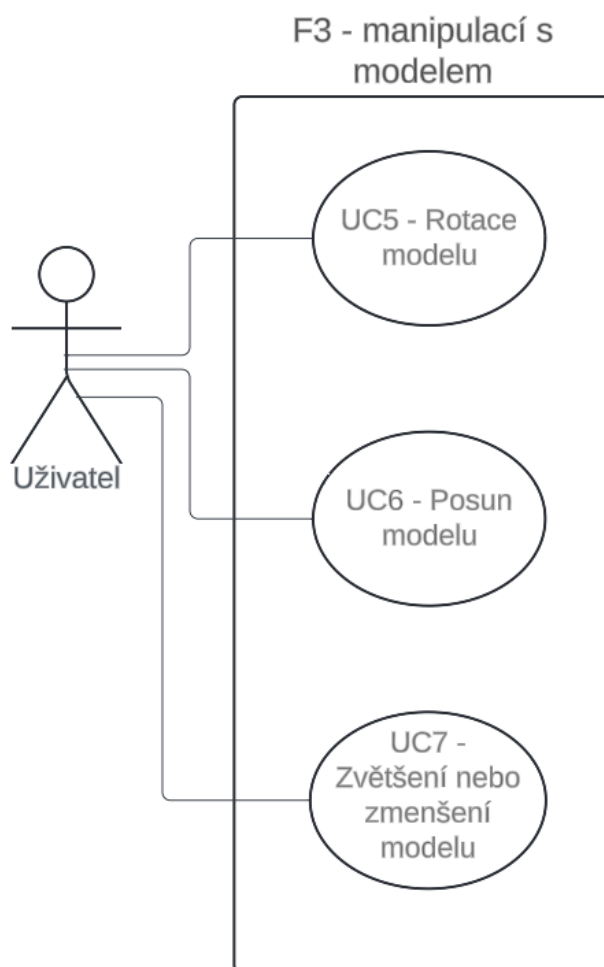
4.5.2.2.2 UC4 – Zobrazit model pomocí URL

Umožňuje začít prohlížet 3D model zadáním adresy URL. Hlavním účastníkem je uživatel.

1. Uživatel vybere v rozhraní systému možnost “Načtení modelu”.
2. Systém zobrazí formulář, který umožňuje zadat URL adresu 3D modelu.
3. Uživatel vyplní povinné pole.
4. Systém zkontroluje, zda jsou vyplněna všechna povinná pole.
5. Systém otevře stránku pro zobrazení načteného modelu.
6. Uživatel si vybere režim zobrazení: běžný, VR nebo AR.
7. Model je zobrazen.

4.5.2.3 F3 – manipulací s modelem

Tato část práce je věnována popisu všech funkcí týkajících se manipulace s modelem. K manipulaci patří rotace, posun a zvětšení nebo zmenšení modelu. Obrázek 4.9 ukazuje všechny případy užití pro manipulací s modelem.



■ **Obrázek 4.9** F3 – manipulací s modelem

4.5.2.3.1 UC5 – Rotace modelu

Umožňuje provádět změnu úhlu otáčení 3D modelů. 3D model musí být zobrazen. Hlavním účastníkem je uživatel.

1. Uživatel aktivuje režim rotace pomocí příslušného ovládacího prvku.
2. Systém začne sledovat akce uživatele.
3. Systém mění úhel otáčení modelu.
4. Uživatel vidí změny v reálném čase.
5. Uživatel dokončí použití příslušného ovládacího prvku.
6. Systém uloží nový úhel otáčení modelu.

4.5.2.3.2 UC6 – Posun modelu

Umožňuje provádět změnu polohy 3D modelu. 3D model musí být zobrazen. Hlavním účastníkem je uživatel.

1. Uživatel aktivuje režim posunu pomocí příslušného ovládacího prvku.
2. Systém začne sledovat akce uživatele.
3. Systém mění polohu modelu.
4. Uživatel vidí změny v reálném čase.
5. Uživatel dokončí použití příslušného ovládacího prvku.
6. Systém uloží novou polohu modelu.

4.5.2.3.3 UC7 – Zvětšení nebo zmenšení modelu

Umožňuje provádět změnu velikosti 3D modelu. 3D model musí být zobrazen. Hlavním účastníkem je uživatel.

1. Uživatel aktivuje režim změny velikosti pomocí příslušného ovládacího prvku.
2. Systém začne sledovat akce uživatele.
3. Systém mění velikost modelu.
4. Uživatel vidí změny v reálném čase.
5. Uživatel dokončí použití příslušného ovládacího prvku.
6. Systém uloží novou velikost modelu.

4.5.3 Nefunkční požadavky

Tato část popisuje všechny požadavky na systém, které nejsou přímo funkcemi, ale jsou nezbytné pro fungování aplikace.

4.5.3.1 N1 – responzivní design

Protože aplikace lze používat na různých zařízeních, jako jsou osobní počítače, mobilní telefony a tablety, musí mít responzivní design rozhraní. Rozhraní by se mělo přizpůsobit velikosti displeje uživatele, aby se aplikace komfortně používala.

4.5.3.2 N2 – rozumná doba renderu modelu

3D modely s nízkým počtem polygonů by se měly načítat rychle a nemělo by to trvat příliš dlouho. U 3D modelů s vysokým počtem polygonů by načítání nemělo být extrémně dlouhé. Renderování modelů nemělo být přetěžováno dalšími funkcemi.

4.5.3.3 N3 – podpora různých druhů zařízení

Aplikace musí fungovat na různých typech zařízení, včetně různých operačních systémů.

4.6 Návrh uživatelského rozhraní

Tato část se věnuje analýze požadavků na uživatelské rozhraní, jako je vzhled, funkčnost, ovládací tlačítka a gesta, a návrhu konečného rozhraní pro vývoj.

4.6.1 Analýza uživatelského rozhraní

Podle zadání bakalářské práce a navržené architektury webová aplikace by měla vyplňovat následující funkce:

- načtení modelu pomocí URL,
- výběr modelu z databáze,
- přidání modelů do databáze,
- mazání modelů z databáze,
- rotace modelu,
- zvětšení a zmenšení modelu,
- posun modelu.

Vzhledem k tomu, že aplikace by měla být použitelná na různých zařízeních, je třeba navrhnout uživatelské rozhraní jak pro mobilní tak i pro počítačovou obrazovku. Základní principy tvorby UI jsou ale stejné pro oba dva přístupy a rozhraní musí být pro uživatele jednoduché a intuitivní.

Na základě požadavků na program lze ovládací prvky v rozhraní rozdělit do dvou kategorií: navigace v menu pro výběr modelu a ovládání zobrazením modelu. Tyto kategorie je třeba při vytváření uživatelského rozhraní posuzovat odděleně, protože vyžadují různé přístupy. Menu pro výběr modelu je standardní aplikace, která pracuje s 2D obrazem, zatímco zobrazení modelu se týká 3D zobrazení.

4.6.2 Výběr modelu

Existuje několik způsobů, jak zobrazit knihovnu modelů pro výběr uživatelem. Ve většině případů jsou všechny tyto způsoby vhodné jak pro počítače, tak i pro mobilní zařízení.

Vzor Galerie se nejlépe osvědčuje při zobrazování často aktualizovaného, vysoce vizuálního obsahu, kde není předpokládána žádná hierarchie [25].

- Grid layout – rozdělení obrazovky na boxy různých nebo stejných velikostí, dostatečně velké náhledy.
- Carousel layout – pomalu prochází náhledy všech modelů nebo uživatel prochází nabídkami pro výběr.
- List layout – seznam náhledů modelů, dlouhá stránka.
- Thumbnail gallery – podobná prvnímu typu, ale náhledy jsou o něco větší, aby bylo možné vidět detaily.

Po porovnání různých typů galerií bylo rozhodnuto použít Grid layout.

4.6.3 Prohlížení modelů

Při návrhu uživatelského rozhraní stránky pro zobrazení modelu je nezbytné zaměřit se na ovládání a na to, jak se s ním uživatel seznámí. Další část této práce se bude věnovat dvěma typům možného ovládání ve 3D světě a zmíní se o problémech souvisejících s UI v smíšené a virtuální realitě.

4.6.3.1 Prohlížení pomocí gest

Pro mobilní zařízení již existují ovládací standardy pro navigace v 3D prostoru. Pokud se podíváme na hry, které používají 3D grafiku, uživatelské rozhraní neobsahuje žádná ovládací tlačítka pro navigaci. Aplikace obvykle nabízí možnost kontrolovat kameru pomocí vizualizace. Pokud je uživatel s tímto typem ovládání již obeznámen, může se obejít bez vizuálních pokynů.

Pro požadavky naší aplikace je třeba myslet na tři ovládací gesta: otáčení, posun objektu, přiblížení a oddálení. Pokud porovnáme gesta pro různé operační systémy (iPhone, Windows Phone, Android), zjistíme, že jsou ve všech případech stejná.[26]

4.6.3.2 Prohlížení modelu pomocí uživatelského rozhraní

Dalším možným způsobem návrhu uživatelského rozhraní by bylo použití tlačítek pro navigaci. To vychází z ovládacích prvků v různých specializovaných editorech, například v softwaru pro 3D modelování. V programu Blender je pro každou akci (zvětšení/zmenšení, otáčení a posun) k dispozici tlačítko. Po jeho

vyvolání se na objektu objeví tři osy, podle kterých se můžeme v modelu orientovat. Toto rozhraní je však pro běžného uživatele, který tento způsob ovládání nezná, nesrozumitelné.

Druhou možností je použití ovládacích prvků používaných v různých editorech ve 3D hrách. Jediným problémem tohoto přístupu je omezení otáčení objektu. Obvykle lze v těchto případech objektem otáčet pouze kolem jedné osy, což znemožňuje jeho zobrazení ze všech stran. To by se dalo vyřešit pomocí dalších ovládacích tlačítek, ale rozhraní by pak bylo přetížené a pro lidské vnímání příliš složité.

4.6.3.3 Problém s rozhraními ve VR a AR

Rozhraní ve VR a AR je neintuitivní a vyžaduje učení, pokud se uživatel s VR v životě nesetkal. K ovládní prohlížení modelu bude zapotřebí tutoriál.

Jak se uvádí v knize *Creating Augmented and Virtual Realities*, mnoho aplikací pro vizualizaci dat v XR nevyužívá 3D prostor a často mají ploché uživatelské rozhraní. V pokynech společnosti Apple pro lidské rozhraní (HIG) [27] se doporučuje plně využívat prostor a vyhýbat se příliš složitým ovládacím prvkům uživatelského rozhraní. Doporučuje se věnovat co největší část obrazovky prohlížení a zkoumání fyzického světa a virtuálních objektů aplikace. Je lepší se vyhnout zahlcení obrazovky ovládacími prvky a informacemi, které snižují působivost zážitku. [28]

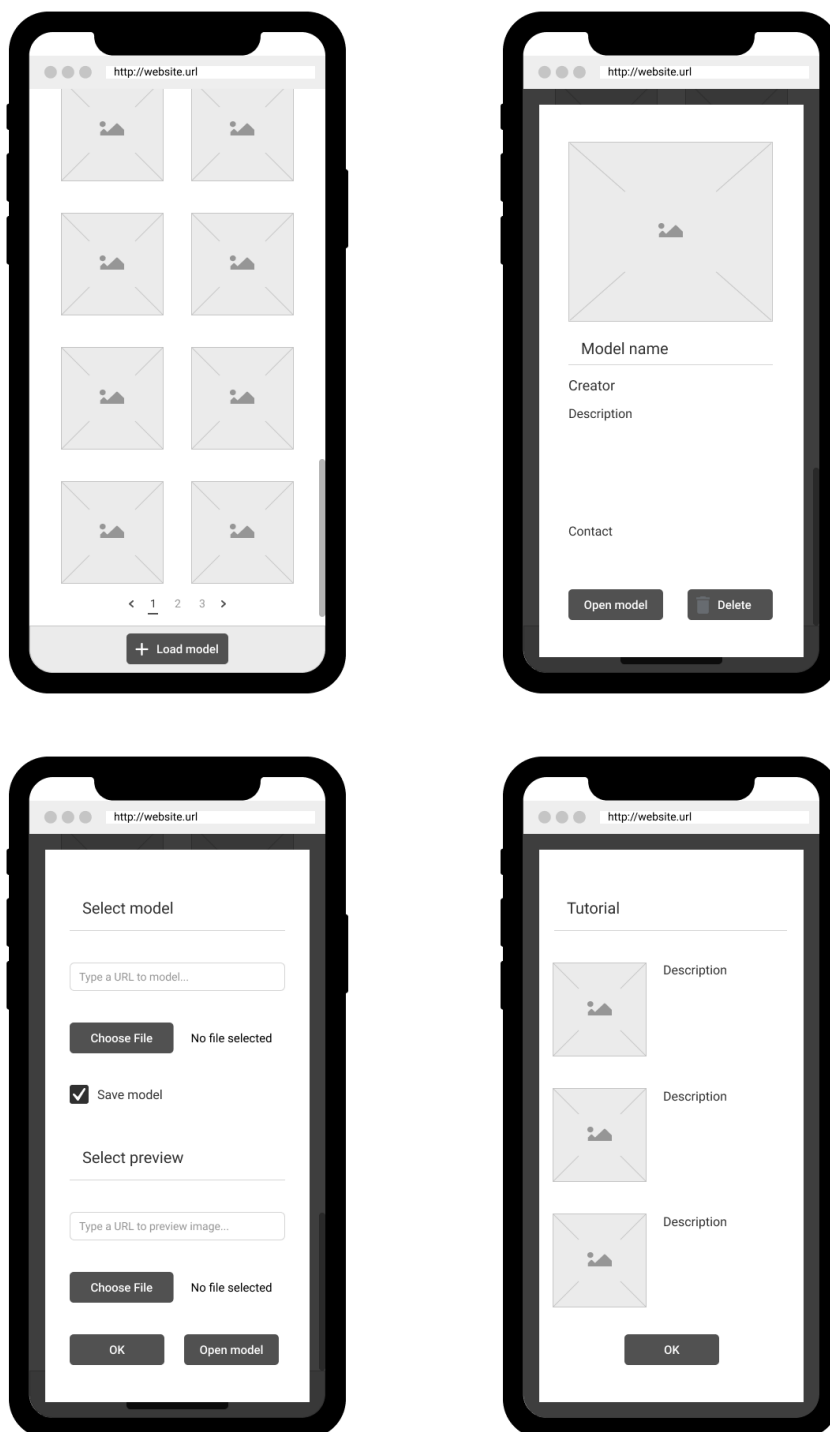
4.6.4 Prototyp uživatelského rozhraní

Na základě provedeného průzkumu bylo navrženo uživatelské rozhraní pro mobilní telefony a počítače. Pro galerie modelů byl vybrán standardní Grid Layout. Při prohlížení modelu se budou používat ovládací gesty, doplněné tutoriálem. Pro ovládání byly vybrány následující gesta:

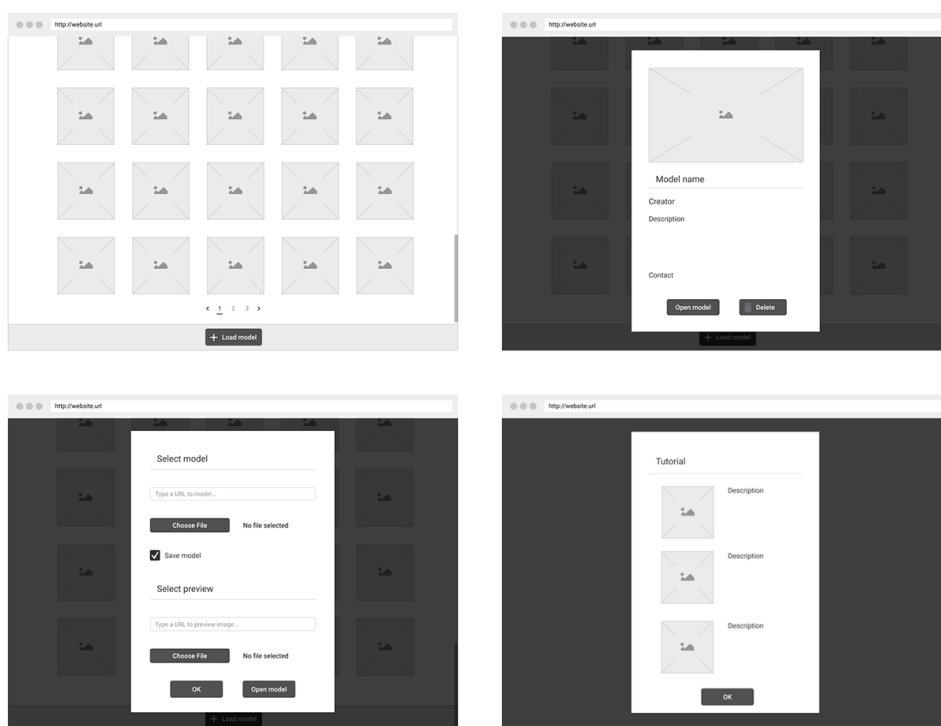
- rotace – stisknout pravé tlačítko a táhnout pro počítače, dotknout jedním prstem a táhnout pro mobilní zařízení,
- posun – stisknout levé tlačítko a táhnout pro počítače, dotknout dvěma prsty a táhnout pro mobilní zařízení,
- zoomování – točení kolečkem myši pro počítače, dotknout dvěma prsty a přiblížit je k sobě nebo je od sebe vzdálit. pro mobilní zařízení.

Po dohodě s vedoucím práce bylo rozhodnuto, že galerie modelů pro zobrazení ve virtuální realitě nebude implementována.

Pro tvorbu UI byla použita online služba Figma. Prototyp uživatelského rozhraní je vidět níže. Na obrázku 4.10 je uveden prototyp rozhraní pro mobilní zařízení a na obrázku 4.11 prototyp rozhraní pro osobní počítače.



■ Obrázek 4.10 Prototyp rozhraní pro mobilní zařízení



■ **Obrázek 4.11** Prototyp rozhraní pro počítače

4.6.4.1 Heuristický průchod

Pro identifikaci problémů s navrženým uživatelským rozhraním byla vybrána metoda heuristického vyhodnocení. Pro průchod byla využita sada pravidel, která byla napsána Jakobem Nielsenem.

- “Visibility of system status” – v každé situaci musí uživatel vědět, v jakém stavu se systém nachází. V případě této aplikace existuje pouze několik stavů: galerie modelů, okno s podrobnějším popisem každého modelu, výběr a nahrávání modelu, samotný prohlížeč modelu, tutorial ovládání.
- “Match between system and the real world” – design musí být srozumitelný pro uživatele. Vzhledem k tomu, že cílovou skupinou jsou zkušení uživatelé, UI by nemělo způsobovat problémy.
- “User control and freedom” – pro uživatele musí být k dispozici možnost opustit jakýkoli stav systému. Tato situace nebyla zohledněna a vyžaduje opravu ve finální verzi rozhraní.
- “Consistency and standards” – uživatel by měl okamžitě pochopit, co znamená každé tlačítko a jak působí. Hlavní osobou je uživatel, který již má zkušenosti s webovými aplikacemi, takže UI by nemělo způsobit problém.

- “Error prevention” – systém musí informovat uživatele o problémech a žádat o potvrzení. Do uživatelského rozhraní by měla být přidána tato oznámení: nahrání modelu do systému a mazání. Systém by měl poskytovat vhodné chybové hlášky v případě selhání otevření modelu, nahrávání či mazání.
- “Recognition rather than recall” – všechny možnosti musí být viditelné, včetně menu systému. V rámci této bakalářské práce existují pouze dvě stránky, a to jsou galerie a prohlížeč
- “Flexibility and efficiency of use” – design pro zkušeného uživatele a naopak. Vzhledem k požadavkům na systém není potřeba se zabývat implementací takových případů.
- “Aesthetic and minimalist design” – uživatelské rozhraní by nemělo obsahovat irelevantní informace. V současném stavu aplikace neobsahuje žádné prvky, které by patřily do této kategorie.
- “Help users recognize, diagnose, and recover from errors” – oznamování o chybě by mělo obsahovat popis problému a také navržený způsob řešení problému. V současné době uživatelské rozhraní tuto funkci nezahrnuje, ale tento požadavek bude zaznamenán.
- “Help and documentation” – pokud systém potřebuje doplňující informace, musí být přístupné pro uživatele. Jediná situace, kdy uživatel může potřebovat pomoc, je při ovládání prohlížeče. [29]

5.3 Vývoj klientské části aplikace

Klientská strana neboli frontend aplikace se skládá ze všech částí aplikace, se kterými bude uživatel přímo komunikovat. To znamená, že se jedná o stránky webové aplikace, skripty chování nebo v našem případě interakce s 3D objektem na scéně, webové požadavky na server a také uživatelské rozhraní.

5.3.1 Vývoj interakcí s modelem

Po vytvoření 3D objektu ve scéně je tento objekt vnořen na své místo. Pro provádění interakcí s modelem je potřeba nějakým způsobem měnit jeho parametry, jako je velikost nebo poloha. Pro tyto účely framework A-Frame nabízí komponenty – znovupoužitelné části dat připojené k objektu, které umožňují přidávat vlastnosti [31]. Lze tak zadávat různé atributy pro objekt jako součást HTML kódu, nebo jako skript. Pro vývoj interakcí byl použit právě druhý typ komponent napsaný v jazyce JavaScript. Výsledná komponenta pro interakce byla napsána speciálně pro prohlížeč 3D modelů.

Manipulace s modelem lze rozdělit na dva různé typy podle toho, v jaké situaci se používají: v režimu XR nebo ne. Pro tyto režimy dá se aplikovat různé přístupy k tomu, jak vypočítat změny v parametrech modelu.

5.3.1.1 Obecné interakce

Hlavním rysem interakcí, které se neprovádějí v režimu XR, je jednoduchost jejich implementace. Když si uživatel prohlíží webovou stránku, je kamera ve 3D světě vždy na stejném místě a nemění svou polohu ani úhel pohledu. Model se nachází na pevně daném místě vzhledem k viewportu, a tak ve spojení s pevnou polohou kamery se ztrácí hloubka 3D světa. Kombinace všech manipulací s objekty však může vytvořit dojem hloubky obrazu.

Z tohoto důvodu pro realizaci interakcí nebylo nutné brát v úvahu nic jiného než závislost mezi souřadnicemi ukazatele na obrazovce a polohou 3D objektu.

5.3.1.2 Rotace

Vzhledem k tomu, že na obrazovce jsou k dispozici pouze 2 osy, je tím značně omezena možnost ovládání objektu. Kvůli tomu můžeme vybrat pouze 2 osy pro rotaci objektu. V projektu byly realizovány změny úhlu podle osy X a osy Y . Volba vycházela ze skutečnosti, že pohyby ukazatele budou odpovídat lidskému vnímání.

Samotná funkce pro rotaci se skládá ze dvou částí: zachycení stisknutí levého tlačítka myši nebo dotyku na obrazovce a zachycení pohybu. Vypočte se změna polohy ukazatele a nové hodnoty se předají modelu. Hodnoty jsou upraveny pro plynulejší animaci. V příkladu kódu 5.1 je vidět, jak se počítá

změna souřadnic ukazatele, a také nastavení proměnné pro úpravu konečných hodnot. Kód 5.2 ukazuje, jak jsou nové hodnoty předány objektu.

■ **Výpis kódu 5.1** Výpočet změny souřadnic ukazatele

```
let deltaX = event.clientX - this.data.prevClientX
let deltaY = event.clientY - this.data.prevClientY
let adj = 0.01
```

■ **Výpis kódu 5.2** Předání nového úhlu otáčení objektu

```
this.data.modelEl.object3D.rotation.x += deltaY * adj
this.data.modelEl.object3D.rotation.y += deltaX * adj
```

5.3.1.3 Posun

Funkce posunu je velmi podobná funkci rotace a využívá stejných principů. V tomto případě pro mobilní zařízení je posun vázán na přetažení dvěma prsty (two finger drag). Vývoj tohoto gesta způsobil problémy vzhledem k tomu, že se nejedná o jediné gesto, které používá 2 dotyky najednou. Jediný rozdíl oproti zoomu spočívá v tom, že se vzdálenost mezi dotyky nemění nebo jsou změny zanedbatelné. Při testování aplikace byly nastaveny nejvhodnější hodnoty pro určení požadované vzdálenosti pro definování konkrétního gesta. Výpočty se provádějí podle standardního vzorce pro vzdálenost mezi dvěma body v rovině:

$$|AB| = \sqrt{(x_B - x_A)^2 + (y_B - y_A)^2},$$

kde $|AB|$ je vzdálenost mezi body A a B , $[x_A, y_A]$ jsou souřadnice bodu A a $[x_B, y_B]$ jsou souřadnice bodu B . Kód 5.3 ukazuje, jak je tento vzorec implementován v aplikaci a jak je následně definován typ interakce.

■ **Výpis kódu 5.3** Rozpoznávání dotyku na mobilních zařízeních

```
if (event.touches.length === 2) {
  let dist = Math.hypot(
    event.touches[0].clientX - event.touches[1].
      clientX,
    event.touches[0].clientY - event.touches[1].
      clientY)

  if (dist < 150)
    this.data.mobileMove = true
  else
    this.data.mobileZoom = true

  this.data.prevDistance = dist
}
```

Samotný výpočet nové pozice závisí na změně polohy ukazatelů. Ale souřadnice ukazatelů na displeji používají pouze nezáporné hodnoty. Bod

$[0, 0]$ se nachází v levém horním rohu obrazovky. Proto se při pohybu po ose X zleva doprava hodnota souřadnice x zvyšuje. U osy Y však hodnota souřadnice y roste, jak se posouváme shora dolů. A-Frame používá pravotočivý souřadnicový systém se zápornou osou Z ukazující do obrazovky [32]. Pro 3D objekt kladná osa Y ukazuje nahoru a záporná dolů. Z tohoto důvodu musíme při změně polohy modelu změnit znaménko u souřadnice y . Jak se výsledné hodnoty předávají modelu, je vidět v kódu 5.4.

■ **Výpis kódu 5.4** Předání nových hodnot polohy objektu

```
this.data.modelEl.object3D.position.x += deltaX *
  adj
this.data.modelEl.object3D.position.y -= deltaY *
  adj
```

5.3.1.4 Zoomování

Velikost 3D objektu se mění pomocí kolečka myši na počítačích. Podle toho, jakým směrem se kolečko myši otáčí, se určuje, zda je model přiblížen nebo oddálen. Kód 5.5 ukazuje, jak je tato funkce implementována pro počítače.

■ **Výpis kódu 5.5** Funkce přiblížení a oddálení pro myš

```
onMouseWheel: function (event) {
  let scale = event.deltaY > 0 ? -0.1 : 0.1
  this.data.modelEl.object3D.scale.multiplyScalar(
    scale + 1)
}
```

U mobilních zařízení je to velmi podobné, ale mají trochu jiné hodnoty a používá se gesto zoomování. V závislosti na tom, jestli je vzdálenost mezi dotyky menší než byla předtím, tak objekt se bude zmenšovat a pokud naopak – bude se zvětšovat. Konečná funkce je uvedena v kódu 5.6.

■ **Výpis kódu 5.6** Funkce přiblížení a oddálení pro mobilní zařízení

```
adj = 0.05

let dist = Math.hypot(
  event.touches[0].clientX - event.touches[1].
  clientX,
  event.touches[0].clientY - event.touches[1].
  clientY)

if (this.data.prevDistance > dist)
  this.data.modelEl.object3D.scale.multiplyScalar(-
    adj + 1)

else if (this.data.prevDistance < dist)
  this.data.modelEl.object3D.scale.multiplyScalar(
    adj + 1)
```

```
this.data.prevDistance = dist
```

5.3.2 Interakce v XR

Hlavním problémem při implementaci interakcí v XR bylo zprovoznit funkce pro režim smíšené reality. V průběhu vývoje se objevil problém s tím, že WebXR nepovoluje zachycení takzvaných pointer eventů [33]. Bez použití pointer eventů není možné provádět manipulace s toho důvodu, že žádná akce uživatele nebude mít vliv.

Při zkoumání možností A-Frame pro tento případ byla nalezena komponenta `webxr`, která povoluje zachycení pointer eventů. Tato komponenta má vlastnost `overlayElement`, která umožňuje vytvořit další vrstvu nad zobrazením scény v režimu AR [34]. Pomocí této vrstvy bylo možné používat interakce v režimu smíšené reality.

5.3.2.1 Změna funkce posunu

První verze aplikace používala stejný přístup k výpočtu nové polohy, velikosti a úhlu otáčení objektu, jak bylo popsáno výše. Kvůli tomu, že prohlížeč modelů používal funkce založené na myšlence o tom, že kamera se nachází na pevně daném místě, objevil se problém s posunem modelu. V smíšené a virtuální realitě se poloha viewportu mění spolu s polohou zařízení, ať už jde o mobilní zařízení, kameru nebo VR brýle. Pokud se mění poloha kamery, mění se i poloha světových os vzhledem k této kameře. Pokud tedy například uživatel obejde model z boku, posun objektu se chová zvláště a neintuitivně. Z tohoto důvodu bylo nutné přepsat dříve napsanou funkci pohybu objektů pro režim XR.

Musíme vzít v úvahu, že 3D objekt se posouvá s ohledem na kameru a její osy. Díky tomu, že A-Frame je založen na knihovně `three.js`, můžeme volně používat jakékoliv funkce z této knihovny. Tímto způsobem získáme všechny prvky, které potřebujeme pro výpočty: “up” vektor a “look” vektor. Vektorový součin těchto dvou vektorů nám dá třetí vektor – “right” vektor. Máme tedy směr všech tří os kamer. Takovým způsobem můžeme přemísťovat model s ohledem na polohu kamery. Uvedený postup byl implementován a je ukázán na kódu 5.7.

■ Výpis kódu 5.7 Upravená funkce pro posun

```
let camera = document.querySelector('[camera]').
  object3D

// get camera direction vector
let cameraDirection = new THREE.Vector3()
camera.getWorldDirection(cameraDirection)
```

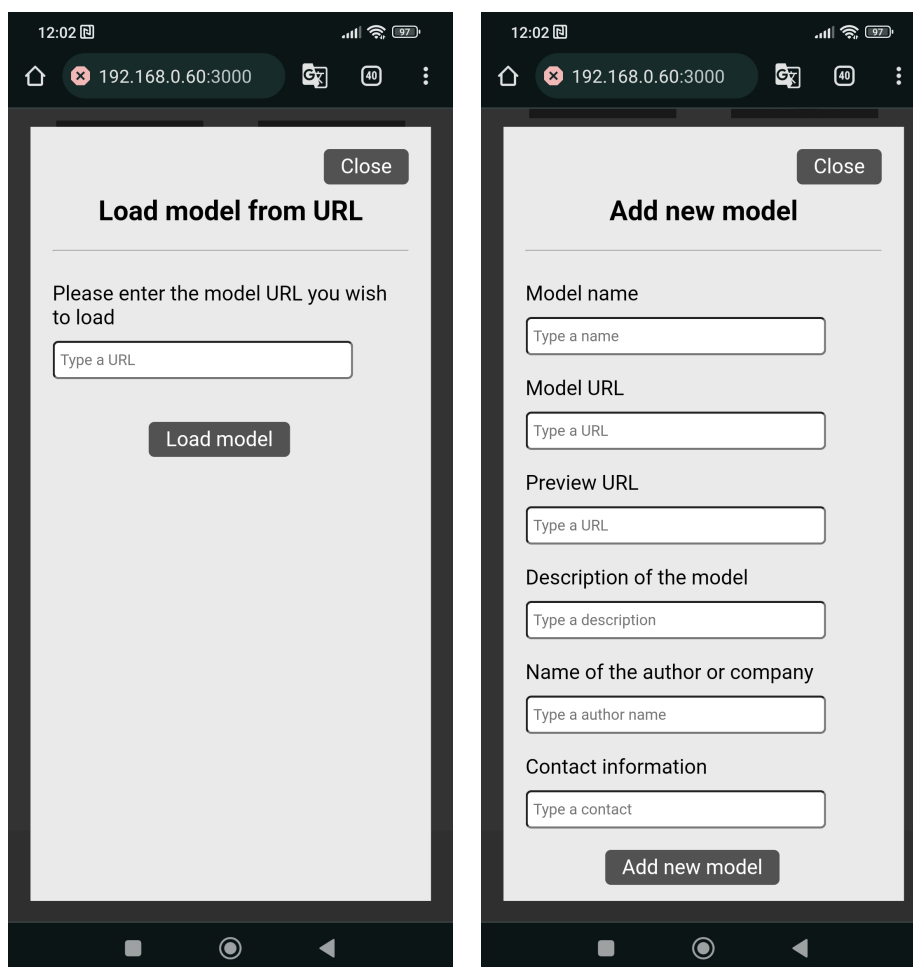
```
// get camera up vector
let cameraUp = camera.up

// compute camera right vector
let cameraRight = new THREE.Vector3()
cameraRight.crossVectors(cameraDirection, cameraUp)

// if deltaX is not negative -> inverse controls
this.data.modelEl.object3D.position.addScaledVector(
  cameraRight, -(deltaX * adj))
this.data.modelEl.object3D.position.addScaledVector(
  cameraUp, -(deltaY * adj))
```

5.3.3 Vývoj uživatelského rozhraní

Uživatelské rozhraní bylo vyvinuto na základě dříve poskytnutého návrhu a s odhalením některých zjištěných problémů. Kromě toho byly funkce, jako je přidání nového modelu do databáze a načtení a zobrazení modelu bez jeho uložení, rozděleny do dvou samostatných oken. Původní myšlenkou bylo mít v okně pro načítání modelu zaškrtačací políčko, které by určilo, zda se má model uložit nebo ne. Tímto způsobem bychom však museli znefunkčnit všechna pole, která není třeba vyplňovat v případě, že model není třeba přidávat do databáze. Vzhledem k tomu, že přidání modelu do aplikace a jeho prosté načtení v prohlížeči jsou v podstatě dvě zcela odlišné funkce, je rozumnější je oddělit. Příklad finálního rozhraní je vidět na obrázky 5.1(a) a 5.1(b).



(a) Okno pro načtení modelu pomocí URL (b) Okno pro přidání nového modelu

Obrázek 5.1 Upravené rozhraní

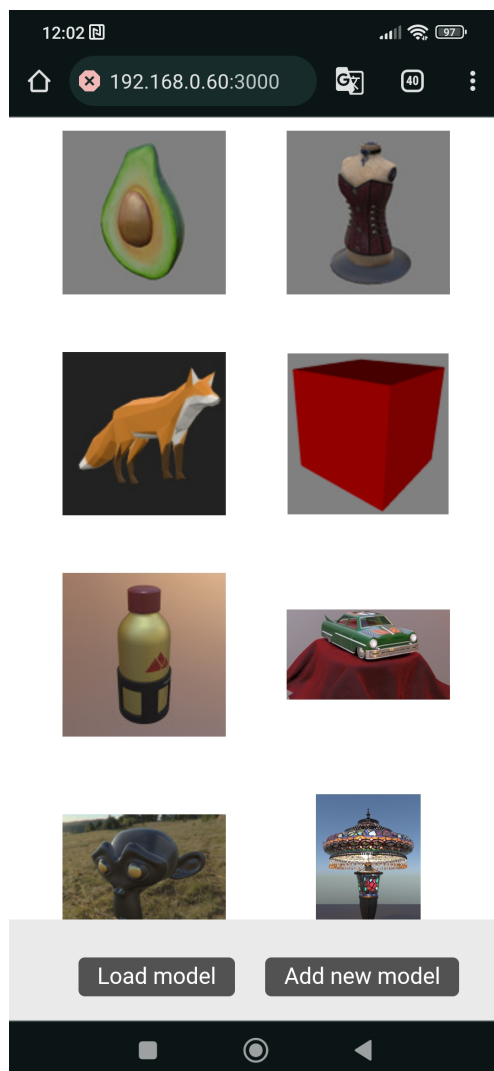
Vzhledem k tomu, že hlavní verzí je mobilní aplikace, byl návod přidán pouze pro tuto verzi. Obrázky pro tento návod byly převzaty z příručky dotykových gest, kterou vytvořili Craig Villamor, Dan Willis a Luke Wroblewski [26].

5.3.4 Použití aplikace

Po přechodu na domovskou stránku, otevře se galerie s náhledy modelů nahraných do databáze. V dolní části stránky jsou 2 tlačítka:

- načíst 3D model pomocí URL – tlačítko “Load model”,
- přidat nový 3D model do databáze – tlačítko “Add new model”.

Menu domovské stránky je vidět na obrázku 5.2.



■ **Obrázek 5.2** Domovská stránka

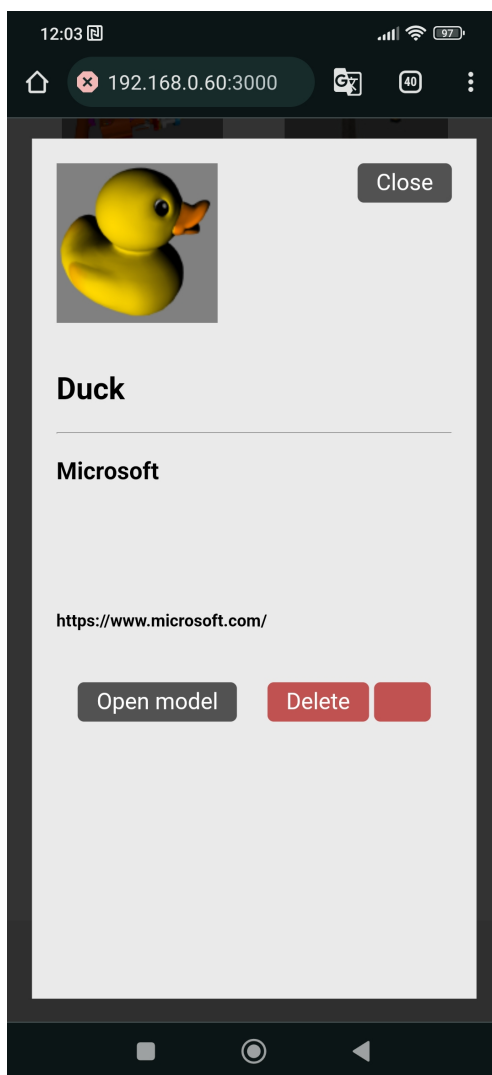
Kliknutím na některý z náhledů modelu se otevře okno s podrobnými informacemi. Obsahuje název 3D modelu, jeho autora, kontakt a popis modelu, pokud je k dispozici. Pokud je obrázek modelu dostatečně velký, zobrazí se zvětšený, aby byly vidět detaily. Pod informacemi se nacházejí 2 tlačítka:

- tlačítko “Open model” otevře 3D model v režimu zobrazení,
- tlačítko “Delete” odstraní model.

Je třeba zmínit, že odstranění modelu vyžaduje správně zadané heslo. Pole pro zadání hesla se nachází vpravo od tlačítka. Pokud uživatel nebo správce

zadá heslo nesprávně, systém jej odešle na stránku se zprávou. V současné době vypadá heslo pro vymazání takto: ID modelu v databázi a jeho název zadané bez mezery.

Okno s informacemi o modelu se nachází na obrázku 5.3.



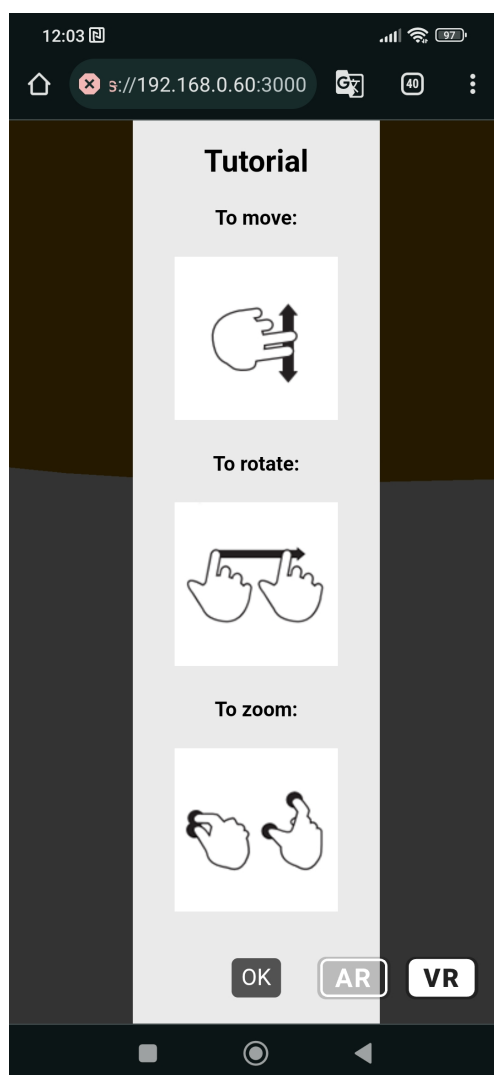
■ **Obrázek 5.3** Okno s informacemi o modelu

Při otevření okna pro načtení 3D modelu pomocí URL se uživateli zobrazí formulář. Do pole je třeba zadat odkaz vedoucí k modelu. Po kliknutí na tlačítko “Load model” bude uživatel přesměrován na stránku pro zobrazení modelu. Okno lze prohlédnout na obrázku 5.1(a) výše.

Při otevření okna pro přidání nového 3D modelu se uživateli zobrazí formulář. Pro přidání modelu do databáze je nutné zadat následující informace: název, URL souboru modelu, URL souboru náhledu, popis modelu, autora a

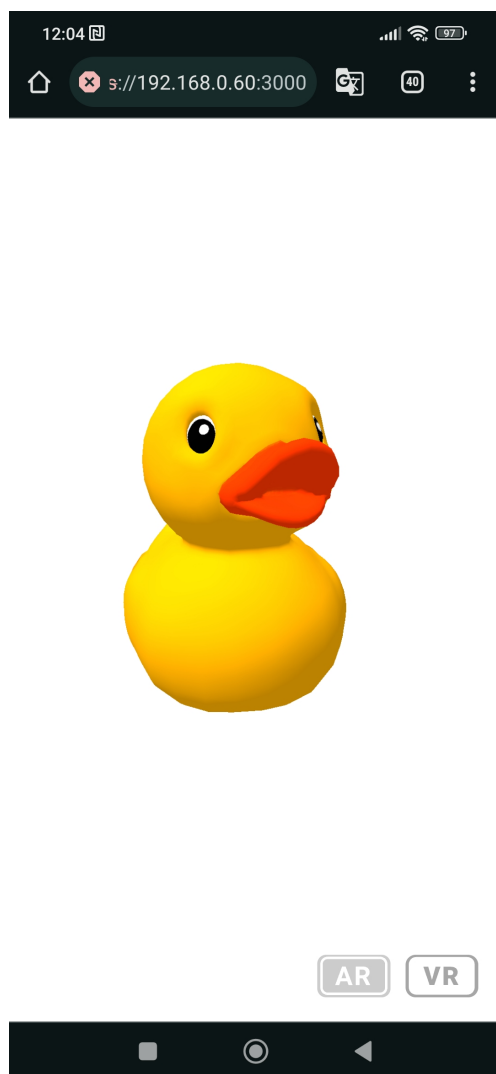
kontakt. Popis modelu je nepovinný. Pokud některé z povinných polí nebudou vyplněny, aplikace o tom bude informovat. Po kliknutí na tlačítko “Add new model” v případě správně vyplněného formuláře bude model přidán do databáze. Okno pro přidání modelu lze nalézt na obrázku 5.1(b) výše.

Po vstupu do režimu zobrazení modelu se zobrazí okno tutoriálu pro mobilní zařízení. V dolní části je tlačítko “OK”, po jehož stisknutí se okno zavře. Okno tutoriálu je na obrázku 5.4.



■ Obrázek 5.4 Tutoriál

V dolní části stránky zobrazení modelu jsou 2 tlačítka pro přepnutí do jednoho z režimů XR. Chce-li uživatel je ukončit, stačí se vrátit na předchozí stránku. Tam bude možnost zvolit jiný režim. Interakce fungují ve všech režimech. Stránku zobrazení naleznete na obrázku 5.5.



■ **Obrázek 5.5** Stránka prohlížení modelu

Kapitola 6

Testování

Tato kapitola popisuje proces testování vyvinuté aplikace, použité metody a problémy a chyby zjištěné během testování.

6.1 Manuální testování

V průběhu vývoje byla aplikace ručně testována na nejrůznějších zařízeních: počítačích a různých modelech zařízení se systémem Android. Hlavní metodou bylo simulovat možné akce uživatele. Pomocí tohoto typu testování byl zjištěn závažný problém s interakcemi s modelem. Popis tohoto problému byl podrobně popsán v části týkající se provádění manipulací v režimu XR.

6.2 Uživatelské testování

Jednu verzi aplikace testovali uživatelé na jednom ze dnů otevřených dveří FIT ČVUT během podzimu roku 2023. Kromě možnosti používat aplikaci byl vytvořen malý dotazník, který mohli návštěvníci akce dobrovolně vyplnit. Dotazník se skládá ze tří otázek:

1. Vyhovuje Vám ovládání prohlížeči modelů?
2. Chtěli byste, aby rozmisťování modelů bylo závislé na plochy v reálném světě?
3. Chtěly byste vidět v AR osvětlení modelu, které by odpovídalo reálnému světu?

Dotazník měl dva cíle: ověřit použitelnost ovládání a zjistit další možnosti rozvoje aplikace. Výsledky průzkumu jsou vypsane v tabulce 6.1.

■ **Tabulka 6.1** Výsledky průzkumu

Odpověď	Otázka 1	Otázka 2	Otázka 3
Ano	11	11	10
Ne	1	1	2

Z odpovědí lze vyvodit závěr, že ovládání prohlížeče modelu nevyžaduje změny a že většina by uvítala realističtější využití 3D objektu v režimu AR.

Hodnocení výsledků

Tato kapitola se zabývá shrnutím výsledků práce. Kapitola je rozdělena do několika částí, z nichž každá se zabývá jiným úkolem a rozsahem jeho splnění.

7.1 Návrh prototypu

Návrh prototypu aplikace obsahuje všechny minimální požadované prvky podle zadání. Několik věcí by se dalo vylepšit, například přidat možnost načítat modely pomocí souboru ze zařízení uživatele, vylepšit tutoriál a přidat možnost kdykoli ho zapnout.

Návrh uživatelského rozhraní je plně dokončen a byl nakonec realizován s drobnými změnami oproti původně navrhované verzi. Po analýze možných problémů s rozhraním pomocí heuristické metody byly ty nejdůležitější v konečném návrhu opraveny. Například byla přidána tlačítka, aby bylo možné se vrátit do předchozího stavu z libovolného stavu v aplikaci, a částečně byly opraveny problémy v případě nesprávně zadaných uživatelských údajů. Pro snadnější používání byl přidán návod při zobrazení modelu.

7.2 Implementace

Podle zadání byla plně implementována webová aplikace pro prohlížení 3D modelů v smíšené a virtuální realitě. V této fázi aplikace plní následující funkce: poskytuje galerii pro výběr modelu k prohlížení, umožňuje uživatelům přidávat a mazat modely a také prohlížet modely bez jejich přidání do databáze.

O každém z 3D modelů lze zobrazit podrobné informace, včetně autorů modelů a případně kontaktních údajů. Pro přehlednost je možné manuálně přidat ke každému modelu jeho velikost při zobrazení v případech, kdy jsou příliš malé nebo příliš velké.

Pro zobrazení modelů v režimu XR byly implementovány všechny základní

manipulace – posun, rotace a změna velikosti. V této verzi aplikace není k dispozici ovládání modelu pro mobilní zařízení v režimu VR, protože v tomto případě zařízení slouží jako obrazovka pro VR brýle.

7.3 Testování

Během vývoje byla aplikace testována převážně manuálně, protože přesné testy nejsou k dispozici a jejich vývoj nebyl cílem této práce. Při této metodě testování však byly pomocí simulace chování uživatelů zjištěny a vyřešeny problémy se zobrazením.

Aplikace byla testována na dni otevřených dveří FIT ČVUT, který se konal během podzimu roku 2023. Všichni návštěvníci mohli aplikaci použít a podívat se na studentské práce. Poté byli návštěvníci požádáni o vyplnění anonymního dotazníku. Uživatelé by si přáli realističtější začlenění digitálního světa do reálného světa prostřednictvím realistického umístění na površích a osvětlení.

Hlavním cílem této práce bylo vyvinout webovou aplikaci využívající knihovnu A-Frame pro zobrazení 3D objektů s možností využití základních interakcí a otestovat konečný výsledek. Tomu předcházelo zkoumání možností zobrazování 3D grafiky ve webových prohlížečích, analýza možností knihovny A-Frame a návrh finální aplikace včetně jejího uživatelského rozhraní.

Byly prostudovány různé metody zobrazování 3D grafiky ve webových prohlížečích. Byly zmíněny dříve populární metody a také nové vznikající technologie. Vzhledem k tomu, že se práce zaměřuje především na VR a AR, nebylo možné nezmínit nástroje pro využití této technologií ve webových aplikacích.

Při studiu frameworku A-Frame jsme se seznámili se základními funkcemi, dozvěděli se o architektuře, podporovaných zařízeních a příkladech použití. Analyzovali jsme také výhody a nevýhody použití tohoto rámce.

Závěrem můžeme konstatovat, že A-Frame je velmi dobrý nástroj pro tvorbu webových aplikací. Nelze si však nevšimnout, že pro některé situace jeho funkcionality nebude stačit, jak ukázal vývoj finální aplikace. Všechny funkce, které byly navrženy v návrhu prototypu, byly implementovány. Některé prvky se vizuálně liší od zamýšleného designu, ale všechny změny oproti původnímu plánu byly provedeny s ohledem na použitelnost aplikace. Hlavním cílem výsledné aplikace byla možnost prohlížení modelů ve smíšené nebo virtuální realitě a implementace základních interakcí s 3D objektem. Přestože prohlížeč 3D modelů splnil svůj úkol, je možné ho dále rozvíjet, jak ukázala zpětná vazba po představení aplikace na dni otevřených dveří. Dalším důležitým krokem by bylo zveřejnění aplikace na internetu.

Bibliografie

1. IQBAL, Mansoor. Pokémon Go Revenue and Usage Statistics [Online]. 2024. Dostupné také z: <https://www.businessofapps.com/data/pokemon-go-statistics/>. [cit. 2024-05-10].
2. ADOBE SYSTEMS INCORPORATED. The History of Flash: The Dawn of Web Animation [Online]. 2015. Dostupné také z: https://web.archive.org/web/20150320140721/http://www.adobe.com/macromedia/events/john_gay/page04.html. [cit. 2023-09-15].
3. ADOBE SYSTEMS INCORPORATED. *Používání aplikace Adobe® Flash® CS4 Professional* [Online]. 2008. Dostupné také z: https://help.adobe.com/archive/cs_CZ/flash/cs4/flash_cs4_help.pdf. [cit. 2023-09-15].
4. THE KHRONOS® GROUP INC. *WebGL* [Online]. 2024. Dostupné také z: <https://www.khronos.org/webgl/>. [cit. 2024-04-29].
5. MDN. *WebGL: 2D and 3D graphics for the web: Browser compatibility* [Online]. [B.r.]. Dostupné také z: https://developer.mozilla.org/en-US/docs/Web/API/WebGL_API#browser_compatibility. [cit. 2024-04-29].
6. WORLD WIDE WEB CONSORTIUM. *WebGPU* [Online]. 2024. Dostupné také z: <https://www.w3.org/TR/webgpu/>. [cit. 2024-05-01].
7. WORLD WIDE WEB CONSORTIUM. *WebGPU Shading Language* [Online]. 2024. Dostupné také z: <https://gpuweb.github.io/gpuweb/wgsl/>. [cit. 2024-05-01].
8. BEAUFORT, François. *Overview of WebGPU* [Online]. 2024. Dostupné také z: <https://developer.chrome.com/docs/web-platform/webgpu/overview?hl=en>. [cit. 2024-05-02].
9. IMMERSIVE WEB COMMUNITY GROUP; IMMERSIVE WEB WORKING GROUP. *WebXR* [Online]. 2024. Dostupné také z: <https://immersiveweb.dev/>. [cit. 2024-05-02].

10. MDN. *WebXR Device API* [Online]. 2023. Dostupné také z: https://developer.mozilla.org/en-US/docs/Web/API/WebXR_Device_API. [cit. 2024-05-03].
11. FURHT, Borko. *Encyclopedia of Multimedia*. 2nd Edition. New York, NY: Springer Science+Business Media, LLC, 2008. ISBN 978-0-387-78415-1.
12. SHERMAN, William R.; CRAIG, Alan B. *Understanding Virtual Reality: INTERFACE, APPLICATION, AND DESIGN*. San Francisco, CA: Morgan Kaufmann Publishers, 2003. ISBN 1-55860-353-0.
13. PEDDIE, Jon. Augmented Reality: Where We Will All Live. In: Cham, Switzerland: Springer International Publishing AG, 2017, s. 20. ISBN 978-3-319-54501-1.
14. A-FRAME. *Introduction* [Online]. 2023. Dostupné také z: <https://aframe.io/docs/1.5.0/introduction/>. [cit. 2024-04-05].
15. A-FRAME. *HTML & Primitives* [Online]. 2023. Dostupné také z: <https://aframe.io/docs/1.5.0/introduction/html-and-primitives.html>. [cit. 2024-04-05].
16. A-FRAME. *Entity-Component-System* [Online]. 2023. Dostupné také z: <https://aframe.io/docs/1.5.0/introduction/entity-component-system.html>. [cit. 2024-04-06].
17. MARTIN, Adam. *Entity Systems are the future of MMOG development – Part 2* [Online, blog]. 2007. Dostupné také z: <https://t-machine.org/index.php/2007/11/11/entity-systems-are-the-future-of-mmog-development-part-2/>. [cit. 2024-04-06].
18. LORD, Richard. *Why use an Entity Component System architecture for game development?* [Online, blog]. 2012. Dostupné také z: <https://www.richardlord.net/blog/ecs/why-use-an-entity-framework.html>. [cit. 2024-04-06].
19. A-FRAME. *VR Headsets & WebXR Browsers* [Online]. 2023. Dostupné také z: <https://aframe.io/docs/1.5.0/introduction/vr-headsets-and-webxr-browsers.html>. [cit. 2024-04-06].
20. GOOGLE LLC. *ARCore supported devices* [Online]. 2024. Dostupné také z: <https://developers.google.com/ar/devices>. [cit. 2024-04-06].
21. OPSLAB JPL. *Access Mars* [Online]. 2018. Dostupné také z: <https://accessmars.withgoogle.com/>. [cit. 2024-04-06].
22. A-FRAME. *Showcase* [Online]. 2023. Dostupné také z: <https://aframe.io/showcase/>. [cit. 2024-04-06].
23. FOWLER, Martin. Patterns of Enterprise Application Architecture. In: Addison-Wesley, 2003, s. 330–332. ISBN 978-0-321-12742-6.

24. A-FRAME. *3D Models* [Online]. 2023. Dostupné také z: <https://aframe.io/docs/1.5.0/introduction/models.html>. [cit. 2024-04-06].
25. NEIL, Theresa. *Mobile Design Pattern Gallery: UI Patterns for Smartphone Apps*. Second Edition. Sebastopol, CA: O'Reilly Media, Inc., 2014. ISBN 978-1-4493-6363-5.
26. WROBLEWSKI, Luke. *Touch Gesture Reference Guide* [Online, blog]. 2010. Dostupné také z: <https://www.lukew.com/ff/entry.asp?1071>. [cit. 2023-11-15].
27. APPLE INC. *Human Interface Guidelines* [Online]. 2024. Dostupné také z: <https://developer.apple.com/design/human-interface-guidelines>. [cit. 2024-04-09].
28. PANGILINAN, Erin; LUKAS, Steve; MOHAN, Vasanth. *Creating Augmented and Virtual Realities: Theory and Practice for Next-Generation Spatial Computing*. First Edition. Sebastopol, CA: O'Reilly Media, Inc., 2019. ISBN 978-1-492-04419-2.
29. NIELSEN, Jakob. *10 Usability Heuristics for User Interface Design* [Online, blog]. 2020. Dostupné také z: <https://www.nngroup.com/articles/ten-usability-heuristics/>. [cit. 2023-12-11].
30. GROUP, The Khronos. *glTF Sample Models* [Online, GitHub repository]. 2023. Dostupné také z: <https://github.com/KhronosGroup/glTF-Sample-Models?tab=readme-ov-file>.
31. A-FRAME. *Component* [Online]. 2023. Dostupné také z: <https://aframe.io/docs/1.5.0/core/component.html>. [cit. 2024-04-25].
32. A-FRAME. *Position* [Online]. 2023. Dostupné také z: <https://aframe.io/docs/1.5.0/components/position.html>. [cit. 2024-04-25].
33. CONSORTIUM, World Wide Web. *WebXR Device API: XR device* [Online]. 2024. Dostupné také z: <https://www.w3.org/TR/webxr/#xr-device-concept>. [cit. 2024-05-10].
34. A-FRAME. *webxr* [Online]. 2023. Dostupné také z: <https://aframe.io/docs/1.4.0/components/webxr.html>. [cit. 2023-10-19].

Obsah příloh

	readme.txt	stručný popis obsahu média
	src	
	app.....	zdrojové kódy implementace
	thesis.....	zdrojová forma práce ve formátu L ^A T _E X
	text.....	text práce
	thesis.pdf.....	text práce ve formátu PDF