

Vývoj AR hry kombinující fyzické a virtuální prvky v enginu Unity

Jakub Šišák

Bakalářská práce
2024



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky
Ústav informatiky a umělé inteligence

Akademický rok: 2023/2024

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: **Jakub Šišák**
Osobní číslo: **A21357**
Studijní program: **B0613A140020 Softwarové inženýrství**
Forma studia: **Prezenční**
Téma práce: **Vývoj AR hry kombinující fyzické a virtuální prvky v enginu Unity**
Téma práce anglicky: **Development of an AR Game Combining Physical and Virtual Elements in the Unity Engine**

Zásady pro vypracování

- Definujte konkrétní herní mechaniky, které budete implementovat.
- Nastudujte a popište nástroje a rozšiřující pluginy pro vývoj aplikací v rozšířené realitě pro engine Unity.
- Zvolte vhodné fyzické objekty, které budete v rozšířené realitě detekovat.
- Implementujte hru v enginu Unity.
- Otestujte vyvinuté řešení na zvolené platformě.

Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam doporučené literatury:

1. KIPPER, Gregory a RAMPOLLA, Joseph. Augmented reality: an emerging technologies guide to AR. Waltham: Syngress, c2013. ISBN <https://www.sciencedirect.com/book/9781597497336/augmented-reality>.
2. GRUBERT, Jens a Raphaël GRASSET. Augmented reality for Android application development : learn how to develop advanced augmented reality applications for Android Bibliographic. Birmingham ; Mumbai: Packt Publishing, 2013. ISBN 9781782168553.
3. MURRAY, Jeff W. C# game programming cookbook for Unity 3D. Boca Raton ; London ; New York. CRC Press, Taylor & Francis Group, 2014. ISBN 978-1-4665-8140-1.
4. HOCKING, Joseph. Unity in Action: Multiplatform Game Development in C# [online]. 3rd ed. Manning Publications, 2022 [cit. 2023-11-12]. ISBN 978-1-61729-933-9. Dostupné z: [https://sd.blackball.lv/library/Unity_in_Action_3rd_Edition_\(2022\).pdf](https://sd.blackball.lv/library/Unity_in_Action_3rd_Edition_(2022).pdf)
5. DORAN, John P. Unity 2020 mobile game development : discover practical techniques and examples to create and deliver engaging games for Android and iOS. 2nd ed. Birmingham ; Mumbai: Packt Publishing, 2020. ISBN 978-1-83898-733-6.

Vedoucí bakalářské práce:

Ing. Štěpán Dlabaja
Ateliér Průmyslový design

Datum zadání bakalářské práce:

5. listopadu 2023

Termín odevzdání bakalářské práce:

13. května 2024

doc. Ing. Jiří Vojtěšek, Ph.D. v.r.
děkan



prof. Mgr. Roman Jašek, Ph.D., DBA v.r.
ředitel ústavu

Ve Zlíně dne 5. ledna 2024

Prohlašuji, že

- beru na vědomí, že odevzdáním bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně;
- byl/a jsem seznámen/a s tím, že na moji bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – bakalářskou práci nebo poskytnout licenci k jejímu využití jen připouští-li tak licenční smlouva uzavřená mezi mnou a Univerzitou Tomáše Bati ve Zlíně s tím, že vyrovnání případného přiměřeného příspěvku na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše) bude rovněž předmětem této licenční smlouvy;
- beru na vědomí, že pokud bylo k vypracování bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem bakalářské práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

- že jsem na bakalářské práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze bakalářské práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně, dne

Jakub Šišák, v.r.
podpis studenta

ABSTRAKT

Práce se zabývá vývojem hry s použitím rozšířené reality (AR) v enginu Unity. Hlavním cílem je vytvořit funkční prototyp hry, která bude připravena na další rozšíření . Tato práce se snaží využít nejnovější technologie a trendy v oblasti AR a gamifikace. K tomu bude využít herní engine Unity, včetně jeho knihovny Unity Asset store. Práce bude zahrnovat vysvětlení důležitých teorií, analýzu existujících AR her, popis postupu vývoje, implementaci herních mechanik a interakcí a testování hry na mobilním zařízení.

Klíčová slova: Unity, Rozšířená, realita, hra, marker, Android

ABSTRACT

The thesis deals with the development of a game using augmented reality (AR) in the Unity engine. The main goal is to create a functional prototype of a game that will be ready for further expansion. This thesis aims to utilize the latest technologies and trends in the field of AR and gamification. To achieve this, the Unity game engine will be utilized, including its Unity Asset Store library. The thesis will include explanations of important theories, analysis of existing AR games, description of the development process, implementation of game mechanics and interactions, and testing of the game on mobile devices.

Keywords: Unity, augmented, reality, game, marker, Android

Děkuji vedoucímu práce panu Ing. Štěpánu Dlabajovi za jeho cenné rady, čas, který mi věnoval a v neposlední řadě také za jeho trpělivost. Dále bych chtěl poděkovat členům své rodiny za pomoc při testování vytvořené hry.

Prohlašuji, že odevzdaná verze bakalářské práce a verze elektronická nahraná do IS/STAG jsou totožné.

Prohlašuji, že při tvorbě této práce jsem použil nástroj generativního modelu AI [ChatGPT; <https://chatgpt.com/>] za účelem přehlednější formulace textu. Po použití tohoto nástroje jsem provedl kontrolu obsahu a přebírám za něj plnou zodpovědnost.

OBSAH

ÚVOD	10
I TEORETICKÁ ČÁST	11
1 ROZŠÍŘENÁ REALITA	12
1.1 REÁLNÁ, VIRTUÁLNÍ A ROZŠÍŘENÁ REALITA	12
1.1.1 REÁLNÉ A VIRTUÁLNÍ PROSTŘEDÍ	12
1.1.2 VIRTUÁLNÍ REALITA	13
1.1.3 ROZŠÍŘENÁ REALITA	13
1.1.4 HISTORIE ROZŠÍŘENÉ REALITY	13
1.2 ZOBRAZOVÁNÍ AR	14
1.2.1 DETEKOVÁNÍ ZALOŽENÁ NA MARKERECH	14
1.2.2 MARKERLESS DETEKCE	15
1.2.3 POROVNÁNÍ	16
2 MOBILNÍ AR APLIKACE	17
2.1 POUŽITÍ AR V PRAXI	17
2.1.1 AR V PROHLÍŽEČI	17
2.1.2 NAVIGACE V DOPRAVĚ	17
2.1.3 ROZPOZNÁNÍ TEXTU	18
2.1.4 AR PRŮVODCI.....	18
2.1.5 ROZPOZNÁNÍ OBLIČEJE	18
2.2 MOBILNÍ AR HRY	19
2.2.1 POKÉMON GO	19
2.2.2 HARRY POTTER: WIZARDS UNITE	19
2.2.3 AR BASKETBALL.....	20
2.2.4 JURASSIC WORLD ALIVE	21
2.2.5 KINGS OF POOL.....	22
2.2.6 STACK AR.....	22
2.2.7 BRICKSCAPE	22
3 VYVÍJENÍ MOBILNÍCH AR APLIKACÍ	23
3.1 UNITY	23
3.1.1 ASSET STORE.....	24
3.1.2 LICENCE	24
3.2 NÁSTROJOVÉ SADY PRO VÝVOJ AR MOBILNÍCH APLIKACÍ	25
3.2.1 ARCORE OD GOOGLU	25
3.2.2 ARKIT OD APPLU	27

3.2.3	VUFORIA	30
3.2.4	ARTOOLKIT.....	30
3.2.5	AR FOUNDATION.....	31
3.2.6	VYHODNOCENÍ	31
3.3	ROZŠIŘUJÍCÍ PLUGINY PRO VÝVOJ	32
3.3.1	PARTICLE SYSTEMS	32
3.3.2	INPUT SYSTEM.....	32
3.3.3	PROBUILDER	32
II	PRAKTICKÁ ČÁST	33
4	PROCES TVORBY.....	34
5	POPIS HRY A IMPLEMENTOVANÝCH FUNKCÍ.....	35
5.1	KONCEPT HRY.....	35
5.2	IMPLEMENTOVANÉ MECHANIKY, MOŽNOSTI HRY A JEJÍ POPIS.....	36
5.3	UŽIVATELSKÉ ROZHŘANÍ	37
5.3.1	SCÉNA MAINMENU	37
5.3.2	SCÉNA GAMESCENE	38
5.4	VYTVÁŘENÍ MAPY	39
5.4.1	DETEKCE MARKERŮ.....	39
5.4.2	SPRÁVNÉ UMÍSTĚNÍ MARKERŮ.....	40
6	POPIS JEDNOTLIVÝCH SKRIPTŮ	41
6.1	OVLÁDÁNÍ MENU	41
6.1.1	CARSELECTION.CS.....	41
6.1.2	CARANIMATION.CS	41
6.2	OVLÁDÁNÍ AUTA	41
6.2.1	CARCONTROLLER.CS.....	41
6.2.2	JOYSTICKCARCONTROLLER.CS	43
6.2.3	PLAYERINPUTCONTROLLER.CS	43
6.2.4	HIDESHOWUI.CS A OPTIONS.CS	44
6.3	VYTVÁŘENÍ MAPY A UMÍSTĚNÍ AUTA NA DETEKOVANOU ROVINU	45
6.3.1	SCANCHECKPOINTS.CS	45
6.3.2	IMAGETRACKING.CS.....	45
6.3.3	CHECKPOINTCONTROLLER.CS	46
6.3.4	PLACEMENTONMESH.CS	48
7	TESTOVÁNÍ	49

7.1 POZNATKY Z TESTOVÁNÍ	49
8 POUŽITÉ ASSETY	50
ZÁVĚR	51
SEZNAM POUŽITÉ LITERATURY.....	52
SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK.....	56
SEZNAM OBRÁZKŮ	57
SEZNAM ÚRYVKŮ KÓDU.....	58
SEZNAM TABULEK.....	59
SEZNAM PŘÍLOH.....	60
PŘÍLOHA P I: UNITY PROJEKT VE FORMÁTU .ZIP.....	61
PŘÍLOHA P II: MARKERY POUŽÍVANÉ KE GENEROVÁNÍ MAPY VE FORMÁTU .ZIP.....	62

ÚVOD

S koncem minulého století se pokrok v technologiích stal nedílnou součástí lidského života a s každým novým dnem postupuje stále hlouběji do všech oblastí našeho světa. Od revoluce v komunikaci a informacích, přes medicínské a průmyslové inovace, až po způsob, jakým se bavíme a vnímáme realitu. Právě v oblasti zábavy a herního průmyslu se v posledních letech otevírá fascinující svět rozšířené reality (AR), která propojuje digitální svět s tím reálným a umožňuje nám zažívat hry a interaktivní obsahy zcela novým způsobem.

Nelze popřít, že trh s AR v posledních letech zažívá strmý růst a proniká do všech klíčových průmyslových odvětví. Tato dynamika je poháněna především rozvojem mobilního AR, jelikož drtivá většina populace dnes disponuje chytrým telefonem, tabletem nebo jiným mobilním zařízením, které umožňuje snadné použití AR technologií. Například se odhaduje, že v roce 2028 AR dosáhne 126.5 milionu uživatelů, a tedy nového vrcholu (Thomas Alsop, 2024) *Statista* [1].

Vzhledem k tomu, že se zajímám o vývoj počítačových her a rozvoj nových technologií, jako je rozšířená realita, tak jsem se rozhodl tyto dvě věci zkombinovat a vytvořit mobilní hru s použitím rozšířené reality.

Vytvořená hra by se mohla zařadit do žánru závodních her. Dráhu si hráč vytvoří kdekoliv na ploché zemi pomocí naskenování kódů rozložených kdekoliv na podlaze. Cílem hry je projet dráhu virtuálním autem v co nejkratším čase.

V této práci objasním rozdíl mezi virtuální a rozšířenou realitou, rozeberu jak přesně rozšířená realita funguje, shrnu její historii a popíši jak funguje detekování reálných objektů.

V další kapitole vysvětlím využití rozšířené reality v různých průmyslových oblastech a představím zajímavé příklady aplikací a her využívající tuto technologii. Dále popíši nástrojové sady a zajímavé rozšiřující pluginy pro vývoj aplikací v AR.

Kromě toho jsou v dokumentu obsaženy další čtyři kapitoly, které se zabývají procesem tvorby hry, podrobným popisem hry a implementovaných funkcí, a také veškerým obsahem, který hra zahrnuje. Také jsou zde popsány skripty a jejich funkce ve hře a proces testování vytvořené hry, zahrnující zpětnou vazbu z různých perspektiv uživatelů. V poslední kapitole jsou vypsány všechny assety použité při vývoji hry.

I. TEORETICKÁ ČÁST

1 ROZŠÍŘENÁ REALITA

Rozšířená realita je technologie, která překrývá reálný svět digitálními informacemi. Toho se dosáhne pomocí kamery zařízení, jako je chytrý telefon nebo tablet, která snímá reálný svět a vkládá do něj virtuální obsah. AR může poskytnout další informace a umožnit realističtější zážitek z interakce s virtuálním obsahem. Mnoho vývojářů se začalo soustředit na rozšířenou realitu v oblastech, jako je cestovní ruch, nakupování, vzdělávání a zábava, protože poptávka po nich čím dál více roste [2]. V této kapitole bude vysvětleno, co to vlastně rozšířená a virtuální realita znamená a krátce si shrneme historii rozšířené reality. Dále se ponoříme do detailů, ve kterých bude vysvětleno zobrazování AR.

1.1 Reálná, Virtuální a Rozšířená realita

I když se jedná o technologie s podobným názvem, principy a fungování AR a VR (Virtuální) se značně liší. Jedná o dvě odlišné technologie s vlastním zaměřením a způsobem fungování. K pochopení pojmů AR a VR je nejprve nutné definovat, co vlastně realita a virtualita znamenají.

1.1.1 Reálné a Virtuální prostředí

Na obrázku 1, v podání Milgrama a spol. [3], virtuální a reálné prostředí představují dva protilehlé póly spektra. Ačkoliv se filozofové zabývají složitým konceptem reality, v běžném chápání vnímáme jako reálné prostředí to, které je podřízeno fyzikálními zákonům. Věci v reálném světě můžeme vnímat přímo, tak jak skutečně existují. Naopak virtualita je počítačově simulované prostředí, které se může a nemusí řídit fyzikálními zákony reálného světa [4].



Obrázek 1 Model hybridní reality [3]

1.1.2 Virtuální realita

Virtuální realita je svět vytvořený pomocí počítačových technologií, které omezují přístup uživatele do reálného světa. Zařízení a software VR prošly významnou evoluční cestou a zaznamenaly již několik růstových skoků. Technologie VR se mohou uplatnit nejen v oblasti zábavy, her a kinematografii ale také při navrhování, školení a rekvalifikaci odborníků, softwarových produktů pro inženýry, architekty, designéry, realitní kanceláře a obchodníky [5].

1.1.3 Rozšířená realita

Rozšířená realita (AR) je technologie, která v reálném čase obohacuje vnímání okolního světa o digitální informace. Tyto informace se vkládají do reálného prostředí a mohou zahrnovat 3D modely, animace, videa, zvukové efekty a další interaktivní prvky. Ty jsou do světa umístěny tak, aby vytvářely dojem, že jsou skutečnou součástí prostředí. Toho se dosahuje například tím, že vyzařují stín na reálné objekty nebo že intenzita a teplota barev světla, které na ně dopadají, co nejpřesněji odpovídají skutečnému světlu. Doplněním těchto dvou realit můžeme dosáhnout zlepšení kvality informací skutečného světa [6]. K tomu, aby AR mohla správně fungovat potřebuje pomoc senzorů (např. kameru, mikrofon, GPS) a haptických zařízení [4].

1.1.4 Historie rozšířené reality

Historie rozšířené reality je bohatá a zahrnuje několik klíčových událostí a osobností, které přispěly k vývoji tohoto konceptu. V padesátých letech 20. století se poprvé objevil termín "rozšířená realita", když Morton Heilig, kameraman, usiloval o to, aby filmy dokázaly diváka ponořit přímo do děje. O deset let později, v roce 1962, Heilig představil model "Kina budoucnosti", známý jako Sen-sorama, který měl tento koncept realizovat. Po šesti letech, v roce 1968, Ivan Sutherland představil funkční prototyp prvního systému rozšířené reality s hlavovým displejem, čímž posunul možnosti interakce s virtuálním prostředím o krok dál. V roce 1975 Myron Krueger založil laboratoř umělé reality, kde se poprvé mohlo experimentovat s interakcí s virtuálními prvky v reálném čase. Klíčovým mezníkem byl rok 1997, kdy Ronald Azuma provedl první průzkum rozšířené reality a definoval ji jako spojení reálného a virtuálního prostředí v reálném čase, což položilo základy pro další výzkum a vývoj v této oblasti. V roce 2000 Bruce Thomas vynalezl první mobilní AR hru, což přineslo rozšíření možností pro využití AR v mobilních zařízeních. Postupem času, zejména od roku 2007, začaly AR aplikace nabývat na popularitě a v roce 2008 se

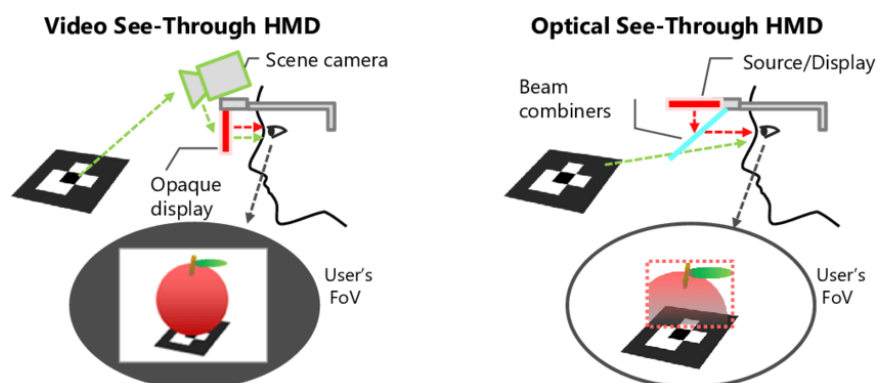
Wikitude AR Travel Guide stal známým, což ukázalo na rostoucí zájem veřejnosti o tento nový způsob interakce s digitálním světem. Společnost Gartner Inc. v roce 2008 předpověděla vzestup rozšířené reality, což potvrdilo rychlý nárůst dostupných aplikací s AR od té doby [7].

1.2 Zobrazování AR

Současné technologie zobrazení pro AR se dělí do dvou kategorií: optická průhledová technologie (OST) a technologie průhledného videa (VST). OST využívá poloprůhlednou obrazovku, na kterou se promítá digitální obsah, přičemž okolní svět je stále viditelný. To umožňuje prolínání reálného a virtuálního světa na sítnici pozorovatele bez ovlivnění vnímání reality, čímž se snižuje riziko fyziologických nežádoucích účinků.

Naproti tomu VST zobrazuje obraz reálného světa pořízený kamerou na displeji. Reálný a virtuální svět se tak slučují před zobrazením, což může vést k řadě komplikací. Mezi ně patří úzké zorné pole kamery, časové zpoždění, nízká rychlost aktualizace, špatná viditelnost a rozlišení, v případě stereoskopie i prostorový stereoskopický nesoulad. Tyto faktory pak mohou způsobovat nepříjemné fyziologické pocity u pozorovatele [8][9].

V mobilní AR se používá technologie VST, a proto se zbytek téhle práce bude spíše zabírat technologií VST.



- a) Optická průhledová technologie (OST) b) technologie průhledného videa (VST)

Obrázek 2 Nejčastěji způsoby zobrazení AR. Zdroj: autor neznámý, s. 3, [9]

1.2.1 Detekování založená na markerech

Rozšířená realita založená na markerech (značkách) vyžaduje, aby uživatelé umístili do místnosti značku, která slouží jako referenční bod pro virtuální objekty, které se objeví po naskenování značky. Algoritmy v sadách SDK (software development kit) pro rozšířenou

realitu získávají ze značky prvky, které propojují virtuální produkty, a umožňují tak uživatelům zobrazit rozšířený obsah v reálném prostředí. Vztah relativní polohy mezi virtuálními objekty a reálným světem je stanoven prostřednictvím markeru [10].

Systém počítačového vidění dokáže rychle a spolehlivě detekovat a identifikovat kvalitní markery. Černé a bílé markery jsou nejlépe detekovatelné objekty za jakýchkoliv světelných podmínek. Proces identifikace značky na pořízeném snímku kamery spočívá v kombinaci jednodušších algoritmů, jako je například detekce hran nebo čar. Začíná binarizací snímku pomocí prahování, což je usnadněno díky vysokému kontrastu na obrázku markeru. Toto zjednodušuje proces sledování „pipeline“. Dvě nejčastější použití markerů v této technice jsou Template a 2D-Barcode markery. Template marker je černobílý obrázek uvnitř ohraničeného bloku. 2D marker se skládá z černobílých datových buněk nesoucích informace o daném objektu, je také známý jako datový marker nebo ID marker. Dobrymi příklady 2D-Barcode markerů jsou QR (Quick Response) kód, Data Matrix kód, Aztec Code a MaxiCode, které můžete vidět na obrázku 3 [11].



a) QR (Quick Response) code b) Data Matrix code c) Aztec Code d) MaxiCode

Obrázek 3 Ukázka různých 2D-Barcode markerů používaných v průmyslu

Rámové markery jsou používány v AR SDK (Software Development Kit), obvykle jako šablony k přizpůsobení potřebám vývojáře, který pracuje s konkrétním SDK. Šablona markeru je černobílý marker, který se skládá z obrázku uvnitř černého blokového okraje. Vnitřní část okraje je složena ze speciálního designu určující ID markeru.

1.2.2 Markerless detekce

Tato metoda je velmi účinná. Cíl nemusí mít omezující vlastnosti jako černobílou barvu nebo blokovou hranici, ale nemusí být dokonce znám předem, protože uživatel může cíl definovat během hry, i když je užitečné, pokud dodržuje určitá pravidla, jako jsou žádné

opakující se vzory, vysoký kontrast, bez odrazivých povrchů, atd. V porovnání s přístupem založeným na markerech vyžaduje významně více zpracování. Pokud je cílem 2D obrázek, pak se skenování obrázku nazývá jako AR založená na přirozených vlastnostech. Každý cíl je třeba nejprve přidat do databáze ve formě sady deskriptorů funkcí [12].

1.2.3 Porovnání

Zřejmě každá metoda pro detekci 3D objektů má své výhody a nevýhody. Neexistuje žádné ideální řešení, a proto se vývojáři musí nejprve zaměřit na to, který přístup je nejvhodnější pro oblast jejich problému. Protože každému přístupu něco chybí (schopnost zachytit obraz, přesnost informací ze senzorů, dostupnost dat v síti, spotřeba energie, výpočetní náklady, citlivost na světelné podmínky atd.), je velmi běžné kombinovat 2 nebo 3 z nich, aby se dosáhlo požadovaných výsledků.

2 MOBILNÍ AR APLIKACE

Mobilní aplikace s rozšířenou realitou otevírají dveře k novým možnostem v digitálním prostoru. V této kapitole se budeme zabývat praktickým využitím rozšířené reality a představíme několik příkladů aplikací, které tuto technologii využívají. Od navigace v dopravě a průvodců pro turisty po hry a rozpoznávání textu. Mobilní AR aplikace pronikají do různých aspektů našich životů a obohacují je o interaktivní prvky.

2.1 Použití AR v praxi

2.1.1 AR v prohlížeči

AR v prohlížeči, známá také jako WebAR, umožňuje uživatelům používat rozšířenou realitu přímo v jejich prohlížeči bez nutnosti stahování a instalace další aplikace.

V minulosti prohlížeče využívaly pouze location-based AR. Postupem času lepší z nich začaly pracovat také s marker AR. Marker AR se stal pro prohlížeče prioritou, protože nabízí větší možnosti pro komerční využití, které je pro prohlížeče výchozí. Nabízelo například zobrazení 3D modelů produktů nebo doplňkových informací v reálném prostředí.

Původně se location-based AR využívalo nejčastěji u průvodců, takzvaných POI (Point Of Interests). Jedná se o památky, známá místa, restaurace, nebo různé instituce, které mohou uživatele zajímat. Uživatel může prostřednictvím kamery smartphonu nahlédnout na památku s POI a zobrazit si o ní informace, které jsou k dané památce dostupné [13].

2.1.2 Navigace v dopravě

AR v budoucnu může zlepšit výkonnost řízení a umožnit řidičům snadněji interpretovat a následovat pokyny ve srovnání s tradičními navigačními systémy. Tímto způsobem AR může usnadnit navigaci v neznámých oblastech a snížit stres spojený s řízením v novém prostředí. Už teď můžeme AR vidět v moderních automobilech při parkování, kdy se na zadní kameře zobrazí rozšířené pole. Kamera, ze které se AR generuje, zobrazí distanční čáry, což dává řidiči lepší představu o tom, kde se nachází a zda se jeho auto vejde na parkovací místo nebo ne [14].

V rámci tzv. chytrých měst někdy dochází k implementaci rozhraní pro AR, kde je uživateli po zadání názvu místa, klíčového slova nebo bodu na mapě zobrazena trasa, doplňující informace a zajímavosti. Dochází pak k rychlejšímu dosažení cílové destinace.

Pro provozovatele místních podniků je pak dosaženo zvýšené efektivity reklam a nárůstu příjmů [15].

2.1.3 Rozpoznání textu

Rozpoznání textu umožňuje automatické čtení a interpretaci textových dat pomocí počítačů a algoritmů. Uplatňuje se například pro digitalizaci historických dokumentů nebo v běžném použití jako překlad z cizího jazyka, obzvláště když je jazyk psaný v jiném psacím systému. Proto je nejpoužívanější aplikací pro rozpoznání textu Google Translator (dostupný na IOS a Androidu), který text naskenuje, přeloží a zobrazí uživateli v reálném čase [16].

V průběhu let se technologie rozpoznávání textu výrazně zdokonalila díky pokroku v oblasti umělé inteligence a strojového učení.

2.1.4 AR průvodci

AR průvodci představují inovativní způsob, jak se vyznat v novém prostředí nebo obohatit cestovní zážitky. AR průvodce můžeme nejčastěji najít na známých turistických místech, kde lidé používají tuto technologii k orientaci a lepšímu porozumění okolnímu prostředí. Pomocí AR průvodce mohou uživatelé objevovat historické památky, informace o zajímavých místech a další zajímavosti, které jsou zobrazovány přímo na obrazovce jejich zařízení.

2.1.5 Rozpoznání obličeje

Rozpoznávání obličeje se stalo všudypřítomnou technologií, která se integruje do široké škály oblastí našeho života. Od odemykání chytrých telefonů po kontrolu identity na letištích a sledování pohybu na veřejných místech. Nicméně nejvíce rozšířenou oblastí jsou stále sociální média. Tam se s rozpoznáním obličeje můžeme setkat u fotek nebo videí, kde počítačem generovaný obsah tvoří tzv. filtry, což jsou emoce nebo masky překrývající obličej uživatele. Snapchat (na IOS a Androidu) je nejpoblárnější aplikace používající tyto funkce a mnoho dalších, jako je editování obličeje na videu v reálném čase, které může být posléze posláno přátelům [17].

2.2 Mobilní AR hry

Mobilní hry s rozšířenou realitou představují novou rozvíjející se oblast herního průmyslu. Spojují virtuální svět s reálným prostředím, což vytváří zcela nový zážitek pro hráče. Zároveň nabízí nové herní mechaniky a otevírají designerům her nové možnosti, jak pracovat s prostředím.

2.2.1 Pokémon GO

Pokémon GO je mobilní hra dostupná na IOS a Androidu, která se rozšířila v roce 2016 po celém světě. Umožnila totiž fanouškům legendární franšizy Pokémon stát se trenéry a prozkoumat herní svět přímo ve svých mobilních zařízeních. Hráči se vydávají do lesů a parků, aby pomocí GPS a kamery svého telefonu chytali pokémony skrývající se v reálném světě, jako můžete vidět na obrázku 4. Na rozdíl od ostatních mobilních her, které nevyžadují velkou fyzickou námahu Pokémon GO podporuje průzkum okolního prostředí, sociální interakci a fyzický pohyb, čímž sblížuje generace a bojuje proti sedavému způsobu života. Pokémon GO pomohl zpopularizovat hry s rozšířenou realitou a ukázal jejich potenciál pro herní průmysl. Otevřel také dveře novým herním konceptům, které budou v budoucnu stále více ovlivňovat naše životy [18].



Obrázek 4 Chytání pokémonů ve hře Pokemon GO [21]

2.2.2 Harry Potter: Wizards Unite

Po velkém úspěchu hry Pokémon GO uvedla společnost Niantic, Inc. v roce 2019 na trh novou hru inspirovanou světem čarodějů J. K. Rowlingové a franšízou Harryho Pottera.

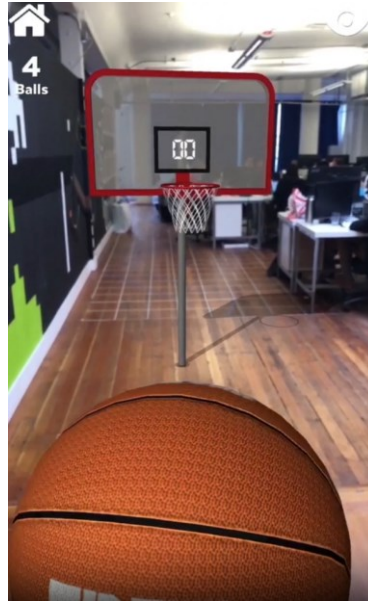
Tato hra nabízela prozkoumání reálného světa v herním prostředí (možné vidět na obrázku 5). Hráči v ní mohli nalézt skryté magické předměty, stopy a tajemství a utkat se s různými bytostmi nebo s jinými hráči. Také se jim naskytla možnost navštívit ikonická místa z knih a filmů o Harrym Potterovi a prožít tak kouzlo magického světa. Technologie rozšířené reality jim umožňovala vnímat magické objekty a bytosti v reálném světě, čímž se interakce s prostředím i s ostatními hráči stala ještě zábavnější [19]. Tato hra v roce 2022 nadobro zmizela z obchodů Google Play a App store poté, co se autoři rozhodli ukončit příběhový oblouk [20].



Obrázek 5 Ukázka prostředí hry Harry Potter: Wizards Unite [19].

2.2.3 AR Basketball

Dostupné na IOS. Jedná se o jednoduchou basketbalovou hru, ve které má hráč za úkol hodit míč do basketbalového koše pomocí gesta přejetí prstu zdola nahoru. Hráč nejdříve musí naskenovat podlahu a poté na ni položit basketbalový koš. Hra je ze začátku velmi zábavná, ale po delším hraní si už hráč zvykne na nastavení a hra rychle omrzí [21].



Obrázek 6 Ar Basketball [21]

2.2.4 Jurassic World Alive

Dostupná na IOS a Androidu. Jurassic World Alive je mobilní hra s rozšířenou realitou, která umožňuje sbírat dinosaury v reálném světě. Cílem hry je hledat dinosaury ve svém okolí, poté je naskenovat virtuálním dronem a posbírat jejich DNA. Čím více DNA má hráč nasbíráno, tím silnější dinosaury může vytvořit. Až má dostatek dinosaurů, může z nich vytvořit tým, který se pak utká proti týmům ostatních hráčů v PvP arénách. Kromě sbírání DNA a soubojů hra nabízí také plnění úkolů, za které hráč dostává ocenění a to vše v rámci poutavého světa dinosaurů [21].



Obrázek 7 Virtuální dinosaurus v reálném prostředí ve hře Jurassic World Alive [21].

2.2.5 Kings of Pool

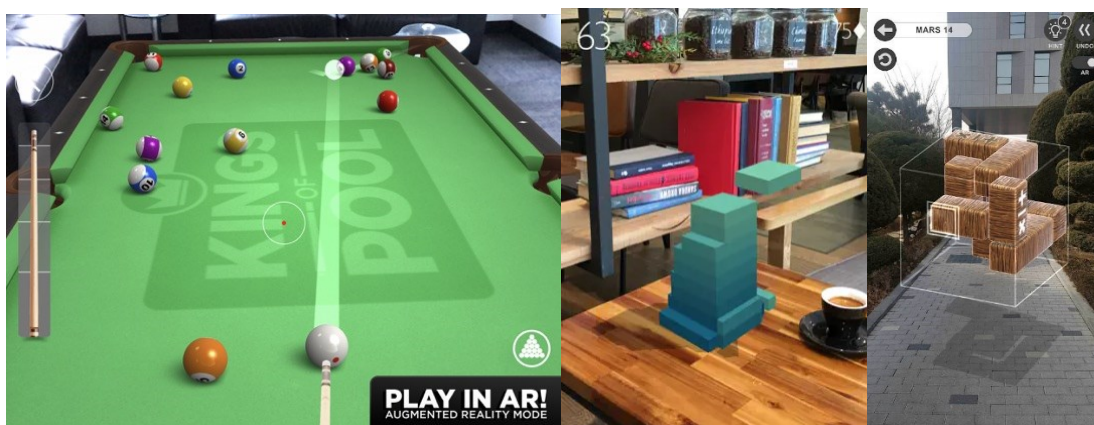
Kings of Pool, dostupná na IOS a Androidu, je kulečnicková mobilní hra, ve které se hráči mohou utkat s soupeři z celého světa v klasických herních režimech a turnajích. Kromě normálního virtuálního režimu hra nabízí AR mód, kde si hráč může položit kulečnick kdekoliv do reálného světa a pomocí gest natahovat kulečnickovou holi. Tato hra představuje skvělou platformu pro hráče, kteří chtějí zdokonalovat své dovednosti v realistickém a interaktivním prostředí [21].

2.2.6 Stack AR

Stack AR dostupná na IOS, je mobilní hra, která propojuje budování věží s technologií rozšířené reality. Hráči v rozšířené realitě umísťují bloky na sebe co nejvíce nejpresněji, aby se jim příští blok nezmenšil. Hra končí tak, že hráč nepoloží blok na konstrukci věže [21].

2.2.7 Brickscape

Brickscape dostupná na IOS, je logická hra s možností hraní v AR, ve které má hráč za úkol dostat hlavní kvádr do určité polohy. Pomocí gest (zleva doprava, shora dolů atd.) musí hráč pohnout s ostatními kvádry tak, aby uvolnil cestu hlavnímu kvádru, který se musí dostat do správné polohy. Pro hraní v AR stačí zvolit v nabídce možnost AR a pak puzzle umístit na pohodlné místo [21].



Obrázek 8 Zleva doprava – Kings of Pool, Stack AR, Brickscape [21].

3 VYVÍJENÍ MOBILNÍCH AR APLIKACÍ

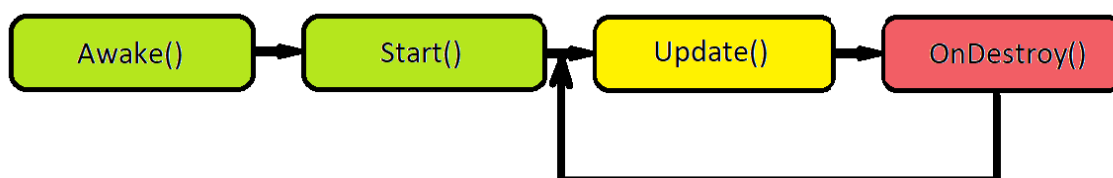
V následujícím textu se zaměříme na vývoj mobilních aplikací s rozšířenou realitou a představíme několik klíčových aspektů spojených s využitím Unity engine, jeho licencemi a Asset Storem.

Neexistuje jediný a definitivní nástroj pro aplikace s rozšířenou realitou. V závislosti na požadavcích aplikace musí vývojář zvážit, jaké nástrojové sady a pluginy chce využít. V sekci (3.1) je výběr 5 nástrojových sad, které lze použít pro vývoj na platformě Android i iOS. Tento konkrétní výběr je založen na pravděpodobnosti použití pro vývoj mobilní AR hry, na dostupnosti dokumentací a na všestrannosti nabízených funkcí. Všechny řečené parametry jsou vyznačeny v tabulce v následující sekci (3.2). Poté se můžete dočíst o doplňujících pluginech k vývoji v AR, které může vývojář použít v případě, že chce doplnit svou aplikaci o další funkcionality.

3.1 Unity

Unity je výkonný multiplatformní herní engine, který umožňuje tvorbu jak 2D, tak i 3D her pro různé platformy jako iOS, Android, macOS, Windows, herní konzole a WebGL v prohlížečích. Jeho popularita spočívá v tom, že usnadňuje vývoj her pro více platforem s minimálním úsilím. Díky tomu je oblíbený zejména mezi začátečníky, kteří mohou využívat bohatou komunitu, návody a podporu na různých fórech.

Pro editaci zdrojového kódu je často používán Visual Studio od firmy Microsoft, který nabízí přímou integraci s Unity. Jazykem programování v Unity je C#. Každá hra vytvořená v Unity obsahuje alespoň jednu scénu, která je tvořena herními objekty. Tyto objekty mohou obsahovat různé komponenty, které ovlivňují jejich chování. Klíčovou komponentou je Transform, která definuje pozici, rotaci a velikost objektů [22].



Obrázek 9 Životní cyklus komponent

Životní cyklus komponent v Unity zobrazen na obrázku 9, je důležitým konceptem, který zahrnuje metody, které jsou volány v určitých fázích životního cyklu objektů a umožňují programátorům ovlivňovat chování svých her. Jedná se o metody:

- `Awake()` - Je volána okamžitě po vytvoření instance objektu a často se využívá pro inicializaci daného objektu.
- `Start()` - Je ideální pro nastavení, které závisí na jiných komponentách a je volána před tím, než se spustí první snímek hry.
- `Update()` - Metoda, která je volána každý snímek a umožňuje aktualizovat stav objektu nebo provádět jiné operace v reálném čase.
- `OnCollisionEnter()` - Je metoda, která je volána, když dochází ke kolizi mezi objekty a je často využívána k detekci a reakci na kolize.
- `OnDestroy()` - Metoda, která je volána těsně před zničením objektu a umožňuje provést úklidové operace nebo uvolnit prostředky.

Další klíčovou vlastností Unity je možnost ukládat herní objekty jako tzv. prefabrikáty (prefabs), které slouží jako šablony pro vytváření dalších objektů se stejnými vlastnostmi. To usnadňuje opakující se vytváření a úpravy objektů ve hře [22].

3.1.1 Asset Store

Asset Store je služba poskytovaná společností Unity, která umožňuje vývojářům distribuovat a stahovat různé nástroje, algoritmy, modely a textury pro použití ve svých projektech. Podobně jako Apple má svůj App Store pro distribuci aplikací, Unity poskytuje Asset Store pro vývojáře, kteří chtějí sdílet své tvorby s ostatními. Vývojáři mohou vytvářet užitečné nástroje, 3D modely s animacemi nebo jiné grafické prvky a nabízet je ke stažení buď zdarma nebo za poplatek. Tyto položky pak mohou být volně používány ve hrách a dalších projektech, pokud není explicitně stanoveno jinak. Asset Store tak poskytuje komunitě vývojářů prostředí pro sdílení a využívání užitečných prostředků pro vývoj her a aplikací ve frameworku Unity [22].

3.1.2 Licence

Licenční model Unity se v průběhu času vyvíjel a aktuálně nabízí různé možnosti pro vývojáře. Základní Personal licence je k dispozici zdarma a umožňuje využití většiny funkcí a služeb Unity, s podmínkou, že roční příjmy nebudou přesahovat 100,000 USD.

Uživatelé této licence jsou povinni zobrazovat logo Unity (splash screen) při spuštění svých her.

Dalšími dostupnými licencemi jsou Plus, Pro a Unity for Enterprise. Licence Plus nabízí všechny výhody Personal licence a navíc umožňuje příjmy do 200,000 USD, slevu v Asset Store, možnost úpravy nebo odstranění Unity splash screen. Licence Pro poskytuje všechny výhody Plus a navíc prémiovou podporu, přičemž neexistuje limit pro příjmy. Pro větší týmy je k dispozici licence Unity for Enterprise, která je uzpůsobena specifickým potřebám společnosti a je dostupná pro týmy o velikosti 21 a více členů [23].

3.2 Nástrojové sady pro vývoj AR mobilních aplikací

Nástrojové sady pro vývoj v AR umožňují vývojářům snadno integrovat AR funkce do svých projektů. Tyto moduly mohou poskytovat funkce pro sledování pohybu zařízení, rozpoznávání obrázku, sledování markerů, renderování AR objektů a lokalizování a mapování prostředí v reálném čase.

V této kapitole bude uvedeno několik dostupných nástrojových sad, které umožňují implementaci rozšířené reality pro platformy Android, IOS a dalších. Následovně bude provedeno jejich vyhodnocení.

3.2.1 ARCore od Googlu

Google v březnu 2018 představil svůj nový nástroj (SDK) ARCore po neúspěšném projektu Tango, který neměl dlouhého trvání, protože byl omezen na zařízení se speciálními senzory, které nebyly součástí chytrých telefonů. Na rozdíl od Tango funguje ARCore na celé řadě telefonů s operačním systémem Android Nougat (7.0) a také na iOS zařízeních podporujících ARKit. ARCore vychází z projektu Tango, ale má méně funkcí, které nezávisí na specializovaných kamerách. Díky tomu je kompatibilní s mnohem větším počtem zařízení. Nabízí podobné technologie jako ARKit od Applu, a to konkrétně sledování pohybu, rozpoznávání prostoru a odhad osvětlení [23].

3.2.1.1 Hlavní funkce

ARCore k fungování využívá hned několik technologií. První z nich je sledování pohybu, které probíhá pomocí kamery a senzoru inerciální měřící jednotky. Kamera snímá charakteristické body scény a senzor IMU sleduje pohyb zařízení. Díky tomu ARCore dokáže přesně určit polohu a orientaci telefonu v prostoru a správně umístit virtuální objekty.

Další důležitou technologií je rozpoznávání prostředí. ARCore k němu využívá tytéž charakteristické body jako k sledování pohybu a dokáže tak detekovat vodorovné plochy. To umožňuje například umístit virtuální nábytek na zem nebo rozložit virtuální mapu na stůl.

Poslední technologií, kterou ARCore využívá, je odhad osvětlení. Pomocí senzorů světla v telefonu dokáže ARCore odhadnout intenzitu okolního světla a v souladu s ním upravit vzhled virtuálních objektů. To je důležité proto, aby virtuální objekty vypadaly realisticky a nenarušovaly dojem z reálného světa [23].

3.2.1.2 Detekce roviny

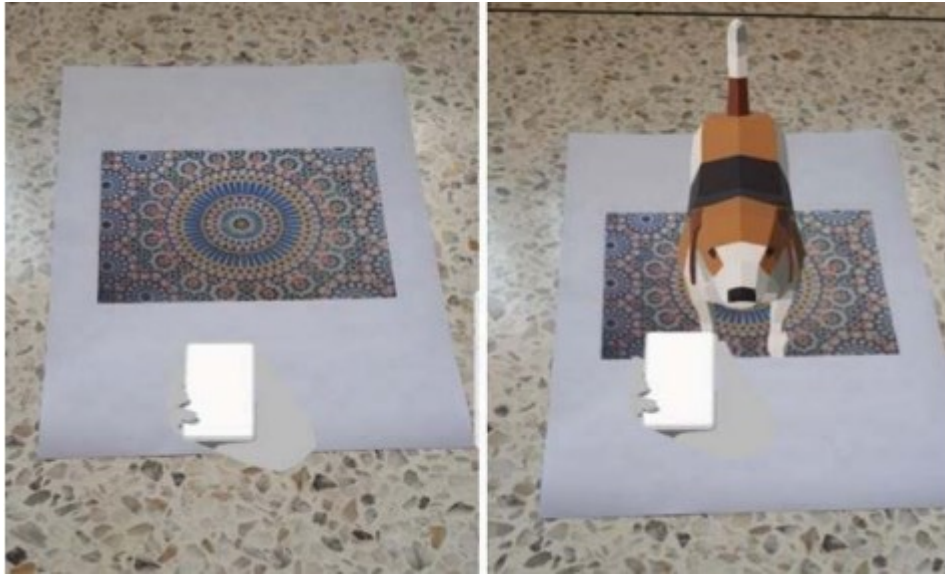
ARCore dokáže analyzovat obraz z kamery a hledat v něm skupiny charakteristických bodů, které pomáhají identifikovat vodorovné a svislé plochy. Dokonce dokáže určit i okraje těchto ploch. Díky tomu můžeme na virtuální podlahu (určenou právě skupinou bodů) umístit 3D objekt, jak je vidět na obrázku 10 [23].



Obrázek 10 Virtuální objekt umístěný na reálnou podlahu pomocí ARCore.

3.2.1.3 Detekce obrázků

Uživatelé také mohou vytvářet rozšířenou realitu pomocí obrázků. Stačí namířit telefon s aplikací na předem uložený obrázek v databázi. Aplikace pak scénu naskenuje a na základě rozpoznání obrázku na něj umístí virtuální 3D objekt. Podobně jako na obrázku 11, kde se na nasnímaném obrázku objeví virtuální objekt. Tímto způsobem lze například oživit dětské obrázkové knížky nebo obohatit turistické průvodce o další informace [23].



Obrázek 11 Objekt umístěn na naskenovaném obrázku pomocí ARCore. Zdroj: autor neznámý, s.5, [23]

3.2.1.4 Detekce obličeje

ARCore nabízí zajímavou funkci s názvem ARCore Augmented Faces. Ta dokáže sledovat geometrii obličeje, včetně polohy špičky nosu, středu hlavy a obou stran čela. Pro co nejpřesnější umístění textur a 3D modelů na rozpoznáný obličej (viz. obrázek 12) využívá ARCore detekci konkrétních oblastí a detailní síť obličeje. Tato síť je vlastně virtuálním obrazem obličeje, složeným z vrcholů, jednotlivých oblastí a středu hlavy uživatele [23].



Obrázek 12 Rozpoznávání obličeje pomocí ARCore. Zdroj: autor neznámý, s.5, [23]

3.2.2 ARKit od Applu

ARKit je sada nástrojů od společnosti Apple, určená pro vývoj aplikací s rozšířenou realitou. Byla uvedena v červnu 2017 a umožňuje vývojářům s aktivním Apple Developer

Accountu vytvářet AR aplikace pro iPhone a iPady se systémem iOS 11 a novějším. Díky tomu mohou uživatelé těchto zařízení zažít virtuální objekty integrované do reálného světa. Celkově je ARKit velice podobný ARCore od Googlu jen s drobnými úpravami [23].

3.2.2.1 Hlavní funkce

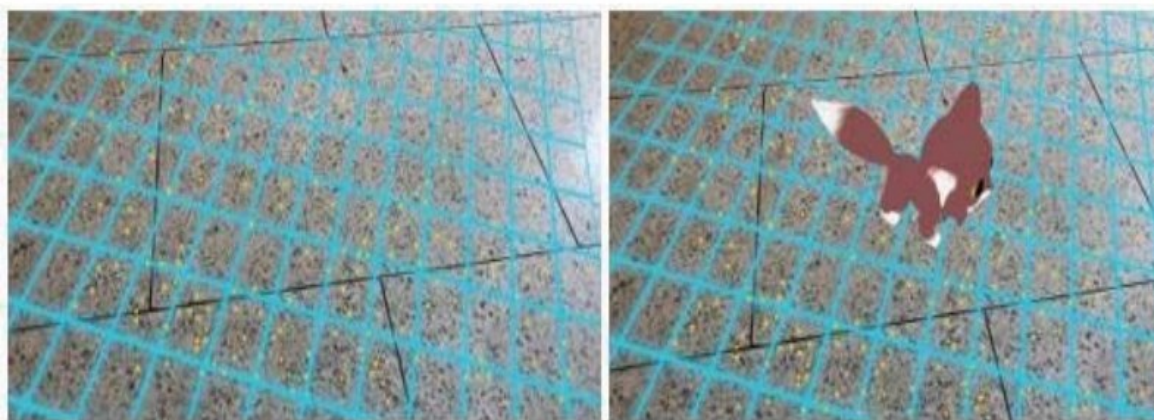
ARKit pro sledování polohy telefonu v reálném čase využívá kombinaci dat z kamery, pohybového senzoru a technologie VIO (vizuálně-inerciální odometr). Díky tomu dokáže ARKit přesně určit polohu a orientaci zařízení v prostoru a umožnit tak plynulé zakomponování virtuálních prvků do reality.

ARKit umožňuje také detekovat vodorovné a svislé plochy v reálném prostředí, což je klíčové pro realistické umístění virtuálních objektů. Navíc dokáže rozpoznávat předem uložené statické obrázky. Po jejich nalezení v okolním prostředí pak umí na tyto obrázky promítat a překrývat virtuálními informacemi (například doplňovat o texty, animace apod.).

ARKit také dokáže například pomocí světelného senzoru odhadnout okolní světlo a na základě toho upravit jas virtuálního objektu pro co největší realističnost. Stejně tak umí na virtuálních kovových předmětech simulovat odrazy okolního prostředí [23].

3.2.2.2 Detekce roviny

ARKit dokáže identifikovat vodorovné a svislé plochy v prostředí díky rozpoznávání a sledování tzv. bodů zájmu. Těmito body mohou být například rohy, linie, změny barvy či tvaru, okraje předmětů a podobně. Pro znázornění detekované plochy aplikace následně vizualizuje její odhadovaný tvar a ohraničí ji modrým obdélníkem, jak je patrné na obrázku 13 [23].



Obrázek 13 Umístění objektu na naskenovanou podlahu pomocí ARKit.

Zdroj: autor neznámý, s.3, [23]

3.2.2.3 Detekce obrázku

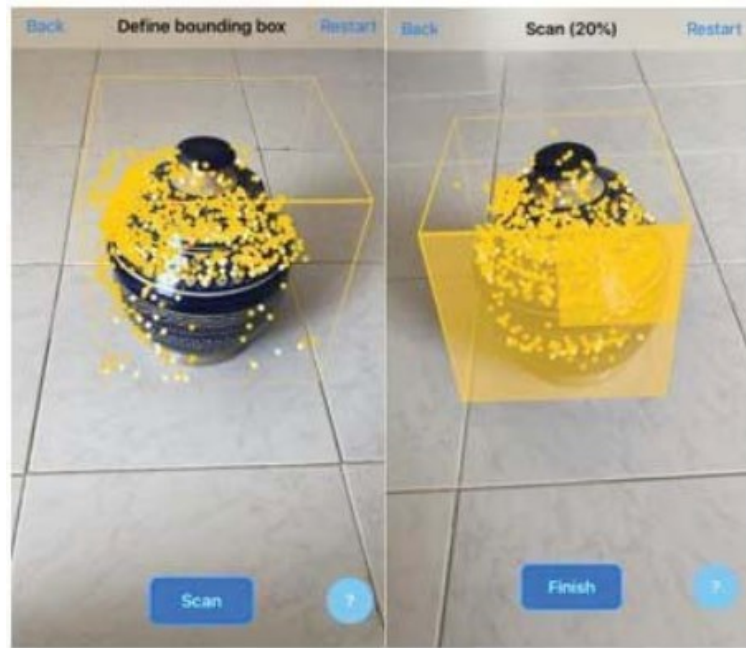
ARKit 2 rozšířil své možnosti o sledování obrázků. Díky tomu dokáže v prostředí detekovat statické i dynamicky se pohybující obrázky. Tyto obrázky musí být předem uloženy v aplikaci. Pomocí analýzy scény pak ARKit daný obrázek rozpozná a na něj umístí virtuální 3D objekt, podobně jako na obrázku 14 [23].



Obrázek 14 Objekt umístěný na obrázek pomocí ARKit. Zdroj: autor neznámý, s.3, [23]

3.2.2.4 Detekce objektů

ARKit 2 také nabízí možnost detekování objektů z reálného světa. 3D objekty se nejdříve naskenují a extrahují se charakteristické body (čím více bodů, tím víc bude sken přesnější), které se dále uloží jako AR objektové rozšíření pomocí ARKit Scanneru, znázorněné na obrázku 15 [23].



Obrázek 15 Skenování objektu pomocí ARKit. Zdroj: autor neznámý, s.3, [23]

3.2.2.5 Detekce obličeje

Detekce obličeje je limitovaná na přístroje s Face ID (Iphone X, Iphone XS a novější modely). ARKit může rozpoznat výrazy obličeje, vzhled, intenzitu světla a geometrii obličeje. Pro práci s ARKitem můžeme použít až 50 polohových koeficientů pro všechny části obličeje, pomocí nich můžeme převést mimiku a pohyb na animované postavy a jiné [23].

3.2.3 Vuforia

Vuforia nabízí stejné funkce jako ARKit a ARCore s možností využití funkcí z obou jmenovaných sad nástrojů a je podporována na Androidu, Linuxu, IOS a macOS. Vuforia také umožňuje rozpoznávání různých typů objektů, prostředí, textu nebo VuMarku (kombinace obrázku a QR kódu). Použitím objektového scanneru lze také skenovat a vytvářet zvolené objekty. Proces rozpoznání se může realizovat pomocí lokální nebo cloudové databáze. Veškeré funkce téhle platformy jsou zdarma k použití, ale obsahují vodoznak Vuforia. Placená verze kromě odstranění vodoznaku nabízí rozpoznání 100,000 objektů, což je 100 krát více než v bezplatné verzi [24].

3.2.4 ARToolKit

ARToolKit je multi-platformní open source knihovna obsahující nástroje k sledování objektů v projektech AR. Jedná se o nástroj počítačového vidění, který snímá obraz

z videokamery a dokáže vypočítat pozici, vzdálenost a rotaci kamery vzhledem k daným objektům. Výsledkem je obraz zobrazující virtuální objekty v reálném prostředí. Jedinou nevýhodou této knihovny je velké množství funkcí, což znamená, že vývojář stráví mnoho času hledáním správné funkce pro vyřešení daného problému [24].

3.2.5 AR Foundation

AR Foundation nabízí vývojáři vytvoření multi-platformní aplikace s rozšířenou realitou v Unity. Pro použití balíčku v unity je potřeba vybrat potřebné AR komponenty a poté je přidat do scény projektu. Při spuštění aplikace na AR zařízení, AR Foundation nastaví a spustí všechny funkce předem nastavené v Unity Enginu pomocí nativního AR SDK. Pro správnou funkci balíčku na platformě, na jaké se aplikace vyvíjí, je nutné nainstalovat jeden z plug-inů poskytovatele, který je na dané platformě podporován. Důvodem je, že balíček AR Foundation sice obsahuje rozhraní pro funkce AR, ale sám o sobě žádné funkce neimplementuje [25].

3.2.6 Vyhodnocení

	ARCore	ARKit	Vuforia	ARToolKit	AR Foundation
Platformy	Android 7.0 Nougat a vyšší	IOS	IOS, Android, Windows	Windows, Mac OS X, Linux, Android, IOS	Android, IOS
Licence	Open-Source	Apple Development Agreement	Open-Source, GNU Lesser GPL v3.0	Open-Source, GNU Lesser GPL v3.0	Open-Source, Unity Companion Licence
Dokumentace	Dostatečná	Dostatečná	Dostatečná	Dostatečná	Dostatečná
Podporované Frameworky	Unity, Flutter, Xamarin, Unreal	Unity	Unity	Unity	Unity
Formáty	.obj, fbx, glTF	.fbx, .obj, glTF	.obj, .v2, x3d	.obj, .v2, .x3d	.obj, .fbx, glTF
Location based AR	Podporuje	Podporuje	Nepodporuje	Nepodporuje	Nepodporuje

Tabulka 1 Porovnání nástrojových sad [23][24][25].

Z důvodu, že cílem této práce je vývoj mobilní hry, pro kterou budu vyžadovat použití mobilní platformy Android s použitím Location based AR, je ideální volbou ARCore rozšířená o AR Foundation pro využití většího množství funkcí Unity.

3.3 Rozšiřující pluginy pro vývoj

Rozšiřující pluginy pro vývoj AR mobilních aplikací představují sadu nástrojů a funkcí, které vývojářům usnadňují integraci pokročilých AR možností do jejich projektů. Tyto pluginy obvykle fungují v tandemu s existujícími AR platformami a nástroji, čímž rozšiřují jejich funkcionalitu a umožňují tvorbu komplexnějších AR zážitků.

3.3.1 Particle Systems

Particle systems se používají k simulování a renderování mnoha obrázků nebo 3D modelů, nazývaných částice. Systém simuluje chování a pohyb všech částic současně, tak, aby se vytvořil co nejpřesnější vizuální efekt. Particle Systems se používají při simulování dynamických objektů a efektů, jako je kouř, oheň, kapaliny nebo počasí [26].

3.3.2 Input System

Unity podporuje vstup z dvou různých systémů. Starší systém se nazývá Input Manager, který je součástí platformy Unity a je nastaven jako výchozí, pokud není nainstalovaný balíček Input system. Balíček Input system je moderní a flexibilní framework, který umožňuje ovládat obsah projektu pomocí široké škály modernějších vstupních zařízení. Pro správný chod balíčku je nutné deaktivovat starší systém Input Manager [26].

3.3.3 ProBuilder

ProBuilder je nástroj, který na rozdíl od Blenderu (externí aplikace pro vytváření geometrie a animací) umožňuje vývojářům přímo v herní scéně snadno a efektivně vytvářet, upravovat a texturovat vlastní geometrii. Nejčastěji se využívá pro navrhování úrovní, prototypování vlastních modelů nebo generování kolizního meshe. Plugin nabízí také API (application programming interface) pro skriptování v C#, pro vytváření vlastních nástrojů a úprav [26].

II. PRAKTICKÁ ČÁST

4 PROCES TVORBY

V první fázi byl zhotoven model testovacího auta a jednoduchý systém pro jeho ovládání pomocí šipek. Následně se ukázalo, že ovládání auta pomocí šipek je velmi nepraktické, a proto byl vytvořen nový ovládací systém používající joystick.

Další část vývoje byla zaměřena na mapování podlahy a následnému umístění auta na tuto mapu. Ze začátku byly kladeny vysoké požadavky na realističnost mapovaného terénu, což vedlo k testování technologií, jako je mapování prostředí od společnosti LightShip, nebo LiDAR. LiDAR se zdál jako nejlepší volba, ale vzhledem k tomu, že většina koncových zařízení technologií LiDAR nedisponuje, bylo nutné se uchýlit k jinému řešení. Jako další bylo do hry implementováno mapování prostředí z vývojářské sady společnosti LightShip. Tato sada nabízí sice skvělé mapování prostředí, ale z hlediska rovné podlahy, na které by auto nemělo problém jezdit byla podlaha velmi nerovná a někdy až autem neprůjezdná. Teď už nezbylo nic jiného než využití komponenty AR plane manager z nástrojové sady AR Foundation. Tenhle komponent sice neobsahuje detekci nerovností, ale díky jednoduché implementaci se ukázalo, že to nebyl tak špatný nástroj.

V posledním kroku bylo potřeba přidat detekování markerů k zobrazení jednotlivých kontrolních bodů trati, které budou reprezentovat brány. Vytvoření kódu pro zobrazování bran na správné pozici markerů nebylo zase tak obtížné. Problém však nastal až při vymýšlení logiky k správnému projíždění trasy, která zabrala delší dobu než jsem předpokládal.

5 POPIS HRY A IMPLEMENTOVANÝCH FUNKCÍ

Kapitola se obecně věnuje vytvořené hře. Popisuje základní principy a poskytuje přehled jejich funkcí a možností pro hráče. Výslednou hru, která je produktem této práce můžete vidět na obrázku 16.



Obrázek 16 Screenshot pořízený ze hry – Vytvořená mapa pomocí naskenovaných markerů

5.1 Koncept hry

Myšlenka hry je založena na hře Mario Kart Live: Home Circuit, ve které mají hráči možnost přenést známou sérii videoher do reálného světa. Tato hra kombinuje rozšířenou realitu s fyzickými závodními autíčky, které se pohybují po reálném prostoru hrací plochy. Každý hráč ovládá své vozidlo pomocí herní konzole Nintendo Switch. Prostřednictvím kamery umístěné na autíčku může hráč na konzoli vidět své auto projíždět tratěmi obohacenými o různé efekty. Trať si hráč poskládá z fyzických branek a markerů které určují směr tratě. Tyto předměty mohou být kdekoli si hráč vybere. Hra nabízí možnost závodit s rodinou nebo přáteli a také vylepšovat své vozidlo prostřednictvím herního pokroku.

Úkolem hráče je projet dráhu v co nejkratším čase, tak aby dojel do cíle před všemi ostatními závodníky. Aby toho docílil, může používat různé vylepšení auta, jako je třeba zrychlení auta nebo pasti, se kterými může chytit jednoho z ostatních závodníků a zpomalit ho na krátký čas. Vylepšení může hráč najít přímo na závodní trati, kde se zobrazují jako otáčející se boxy a aplikují se po jejich najetí.

Má hra funguje podobně jako Mario Kart Live, s tím rozdílem, že auta jsou nahrazeny virtuálními objekty a závodní trať si hráč vytvoří z markerů položených kdekoliv na podlaze. Cílem hry je dostat se do cíle v co nejkratším čase. V následující kapitole je výsledná hra popsána podrobněji.

5.2 Implementované mechaniky, možnosti hry a její popis

K zobrazení dráhy jsou nezbytné předem vytisknuté fyzické markery. Tyhle markery po naskenování fotoaparátem zařízení zobrazí brány, kterými hráč musí projet k dokončení závodu. Markery musí hráč před spuštěním samotné hry rozmístit po podlaze tak, aby tvořily dráhu, kterou si může libovolně navrhnout. Markery jsou vytvořeny tak, aby hráč neměl problém s rozpoznáním v jakém směru se brány zobrazí.

Po spuštění hry se zobrazí menu, ve kterém má hráč možnost vybrání auta. Každé auto má jiné vlastnosti, jako je rychlost, točivost a smyk. V menu má hráč také možnost otevřít nastavení stiskem tlačítka „OPTIONS“, kde si může změnit ovládání z joysticku na ovládání pomocí šipek.

Po stisknutí tlačítka „PLAY“ se zobrazí rozhraní a zapne se zadní kamera na zařízení, která začne analyzovat obrazový vstup, přičemž se identifikují ploché povrchy v záběru, které se hráči vykreslí modrou barvou. Poté, co hráči vyhovuje poloha detekované roviny, může kliknutím prstu na detekovanou rovinu umístit vybrané auto. Tím se vykreslené modré plochy na obrazovce zneviditelní, což přispěje k lepšímu zážitku hráče z rozšířené reality. Dále se hráči v horní části obrazovky zobrazí text, který napovídá k naskenování všech markerů, které předem položil na zem. Při skenování má hráč pořád možnost pohybovat s jednotlivými markery, které aktualizují pozici generovaných brán tak, aby byl spokojený s jejich umístěním. Až je hráč spokojen s tím jak dráha vypadá, může kliknout na tlačítko „Scanned“, které přestane aktualizovat pozici objektů položených na markerech a tak nenastane situace, ve které by se dráha měnila.

Teď už je hráč připraven závodit. Pro začátek závodu musí hráč projet červenou bránou, která signalizuje startovní bod. K odjetí jednoho kola závodu musí hráč projet všemi ostatními zelenými brankami a znovu startovní červenou brankou. Pokud se to hráči podaří a správně projel všemi brankami, připiše se mu odjeté kolo. Až hráč projede celou dráhou 3x, tak se mu na obrazovce zobrazí text, který říká, že se mu úspěšně podařilo dojet závod s časem, za jaký trať dojel a tlačítko k opakování závodu.

Uživatelské rozhraní se skládá z ovládacích tlačítek sloužícím k ovládání auta. Dále se zde nachází ikona nastavení, která po kliknutí zobrazí nastavení, ve kterém si může hráč

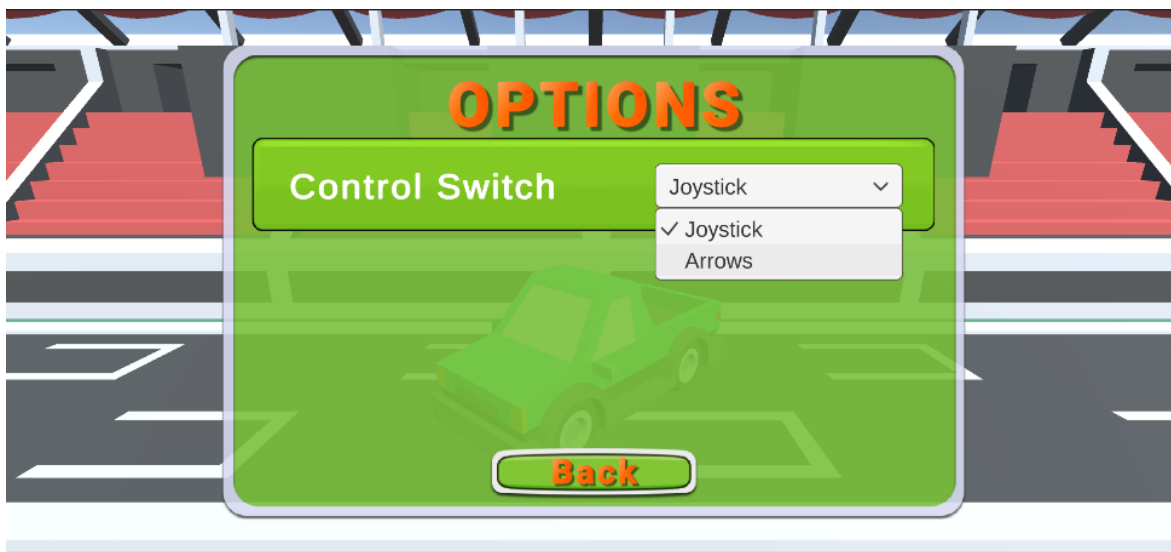
změnit ovládání z joysticku na ovládání šipkami a naopak. Textu s počtem odjetých kol a reálného času stráveného při ježdění tratě od projetí první branky do projetí poslední branky.

5.3 Uživatelské rozhraní

Uživatelské rozhraní je rozděleno do dvou scén *MainMenu* a *GameScene*. Graficky je obrazovka hlavního menu tvořena 3D prostředím závodní dráhy obsahující plovoucí obrazovky s tlačítky sloužící k přesměrování a šipky pro vybrání auta. V herní scéně je potom uživatelské rozhraní zcela odlišné, zaměřené na ovládání auta.

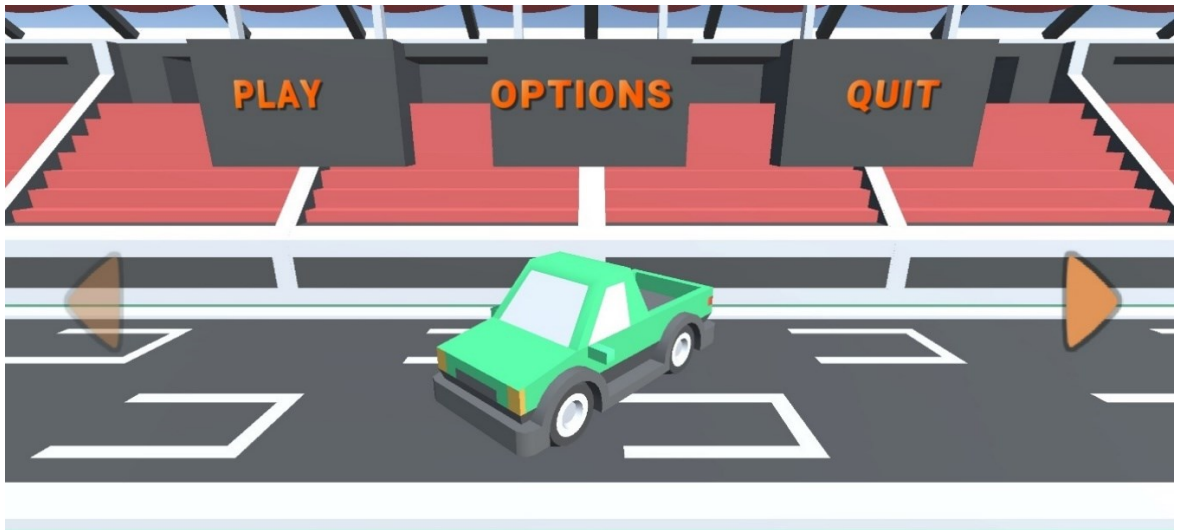
5.3.1 Scéna MainMenu

Tato scéna se hráči zobrazí hned po spuštění hry. V levém horním rohu na se nachází tlačítko „PLAY“, které hráče přesměruje do scény *GameScene*. Uprostřed obrazovky je další tlačítko „OPTIONS“, které zobrazí vyskakovací okno s nastavením. V tomto okně lze zatím jen změnit možnost ovládání auta z šipek na joystick a naopak.



Obrázek 17 Vyskakovací okno s nastavením ve scéně MainMenu

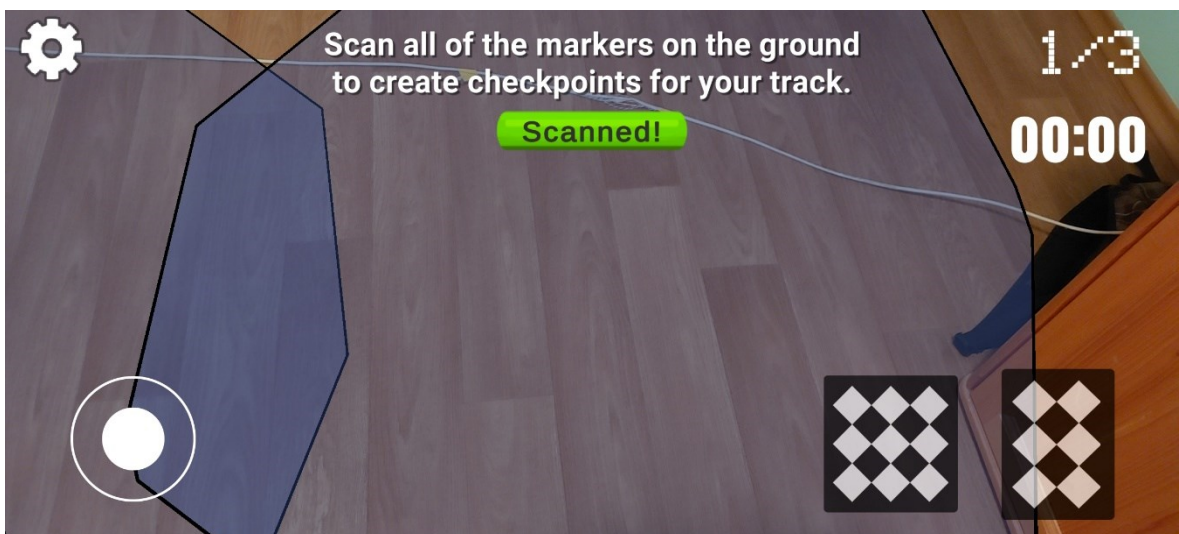
Tlačítkem „QUIT“ v pravém horním rohu může hráč hru jednoduše vypnout. V dolní části obrazovky jsou šipky určené k výběru auta, přičemž hráč potvrdí svůj výběr stisknutím tlačítka „PLAY“.



Obrázek 18 Scéna MainMenu

5.3.2 Scéna GameScene

Tato scéna se spustí po stisknutí tlačítka „PLAY“ ve scéně *MainMenu*. Po spuštění se v dolní části obrazovky zobrazí pedály pro pohyb dopředu a dozadu. Také se zde nachází uložené ovládací rozhraní, které lze upravit v nastavení, jež se aktivuje kliknutím na ikonu nastavení v levém horním rohu. V pravém horním rohu se nachází číslo kola, ve kterém se hráč právě nachází. V pravém horním rohu se nachází číslo kola, ve kterém se hráč právě nachází. Při projetí dalších kol se počet zvyšuje. Pod touto komponentou je časovač, který se spustí kontaktem auta s prvním kontrolním bodem a zastaví projetím posledního kontrolního bodu.



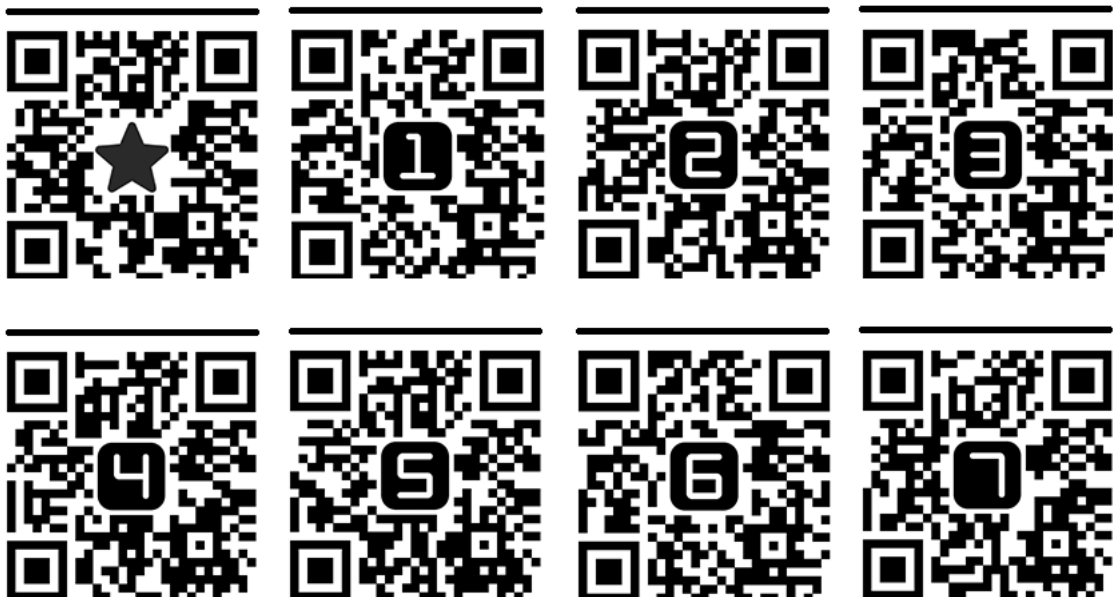
Obrázek 19 Scéna GameScene

5.4 Vytváření mapy

5.4.1 Detekce markerů

Při výběru vhodného typu markerů byla kladena zvláštní pozornost na to, aby byl daný marker bez problémů detekován mobilním zařízením. Během tohoto procesu byl vybírán takový marker, který by byl snadno rozpoznatelný a spolehlivý pro mobilní zařízení, což bylo klíčové pro správnou funkci detekce vytvořené hry. Nakonec byl vybrán marker QR, protože nabízí vysokou přesnost rozpoznání a širokou kompatibilitu s různými typy mobilních zařízení. Markery byly vytvořeny pomocí stránky QR.io [27], která umožňuje zdarma a snadně generovat QR kódy. Při vytváření QR kódů byla doprostřed umístěna buď hvězdička signalizující startovní bránu (červená brána), nebo čísla od jedné do sedmi signalizující záchytný bod (zelená brána). Dále nad marker byla přidána čára, která ukazuje kterým směrem se brána generuje, znázorněno na obrázku 20. Tyto věci byly udělány proto, aby hráč při jejich umístování věděl co každý marker představuje a jakým směrem je umístit.

Markery zatím ve hře nejsou dostupné, protože se hra stále nachází ve fázi vývoje. Nicméně, v pozdější fázi a před publikací hry budou tyto markery ve hře zpřístupněny s možností připojení k tiskárně a jejich vytisknutím.



Obrázek 20 Ukázka vytvořených markerů.

5.4.2 Správné umístění markerů

Markery by měly být umístěny alespoň metr od sebe tak, aby auto mělo dostatek místa pro otáčení. Také by se měly nacházet na rovném povrchu z toho důvodu, že komponenta AR plane manager detekuje jen rovné plochy a na nerovném povrchu by byla nepoužitelná.

6 POPIS JEDNOTLIVÝCH SKRIPTŮ

6.1 Ovládání menu

Tato část textu se zaměří na skripty, které řídí veškeré operace v hlavním menu. Všechny tyto skripty jsou uloženy ve složce „Assets/Scripts/Menu/“.

6.1.1 CarSelection.cs

Tento skript nazvaný „CarSelection“ se stará o výběr auta v herním rozhraní. Jeho úkolem je umožnit hráči procházet seznam dostupných vozidel a vybrat si jedno z nich pro hraní hry. Při startu scény „Menu“ skript inicializuje různé proměnné a nastavuje výchozí auto. Získává referenci na objekt, který obsahuje seznam dostupných aut („carHolderObject“) a zajišťuje, že je vybráno první auto ze seznamu. Dále skript řídí možnost přepínání mezi auty pomocí tlačítek. Pokud hráč použije tlačítko pro předchozí auto nebo následující auto, aktualizuje se index aktuálního auta a zobrazené auto se mění podle tohoto indexu. Kromě toho skript předává informace o aktuálně zvoleném autě skriptu „PlacementOnMesh.cs“, ve kterém se hráčem zvolené auto umístí na zem.

6.1.2 CarAnimation.cs

Tento skript s názvem „CarAnimation“ řídí animaci pohybu auta v menu. Jeho účelem je zajistit plynulý přechod auta mezi pozicí mimo scénu a pozicí ve scéně. Tento způsob zajišťuje, že vybírání aut v menu vypadá dynamicky.

6.2 Ovládání auta

Tato část textu se zaměří na skripty, které se starají o pohyb auta a změnu rozhraní pro jeho ovládání. Všechny tyto skripty jsou uloženy ve složce „Assets/Scripts/CarMovement/“.

6.2.1 CarController.cs

Tento script s názvem „CarController“ je zodpovědný za řízení pohybu auta pomocí šipek a pedálů. Obsahuje různé parametry, které ovlivňují chování auta, jako je rychlost, rychlost zpětného pohybu, točivý moment apod.

Samotný pohyb auta je řízen pomocí několika metod, jako je „Accelerate“ pro zrychlení vpřed, „Reverse“ pro pohyb vzad, „TurnLeft“ a „TurnRight“ pro otáčení doleva a doprava, znázorněné na úryvku kódu 21. Tyto metody upravují rychlost a točivý moment

auta pomocí přidání síly nebo točivého momentu na rigid body auta a jsou volány ve třídě „PlayerInputController“.

```
0 references
void Update()
{
    if (carRigidBody.velocity.magnitude <= minSpeedBeforeIdle)
    {
        carRigidBody.angularVelocity = Vector3.zero;
        AddWheelsSpeed(0);
    }
}

1 reference
public void Accelerate()
{
    carRigidBody.AddForce(transform.forward * speed, ForceMode.Acceleration);
    AddWheelsSpeed(speed);
}

1 reference
public void Reverse()
{
    carRigidBody.AddForce(-transform.forward * reverseSpeed, ForceMode.Acceleration);
    AddWheelsSpeed(-reverseSpeed);
}

1 reference
public void TurnLeft()
{
    if (canApplyTorque())
    {
        carRigidBody.AddRelativeTorque(Vector3.down * torque);
    }
}

1 reference
public void TurnRight()
{
    if (canApplyTorque())
    {
        carRigidBody.AddRelativeTorque(Vector3.up * torque);
    }
}
```

Úryvek kódu 1 Ukázka kódu pro pohyb auta pomocí šipek.

Důležitou součástí skriptu je také metoda Update, která je volána každý snímek hry a zajišťuje, že pokud se auto pohybuje dostatečně pomalu, je jeho úhlová rychlost a rychlost koleček nastavena na nulu, což znamená, že se auto přestává pohybovat.

6.2.2 JoystickCarController.cs

Tento skript řídí pohyb auta pomocí joysticku. Jeho hlavním úkolem je zpracování vstupů z joysticku a převod těchto vstupů na pohyb auta. Hlavní metoda tohoto skriptu je „JoystickMovement“ (úryvek kódu 2), která zpracovává vstupy z joysticku a je volána při každém snímku hry ve třídě „PlayerInputController“. Pokud je joystick dostupný a auto může aplikovat točivý moment (podle nastavené minimální rychlosti), metoda vypočítá směr pohybu z vstupních dat joysticku. Následně se tento směr převádí do prostoru kamery a aplikuje se na rotaci auta pomocí interpolace rotace, což zajistí plynulé otočení auta směrem, kterým se má pohybovat.

```
2 references
public void JoystickMovement(int JoystickRotation)
{
    Awake();
    if (joystick == null)
    {
        return;
    }

    if (canApplyTorque())
    {
        float cameraRotationY = Camera.main.transform.rotation.eulerAngles.y;
        float moveHorizontal = joystick.Horizontal;
        float moveVertical = joystick.Vertical;
        Vector3 movement = new Vector3(moveHorizontal, 0.0f, moveVertical);
        movement = Quaternion.Euler(JoystickRotation, cameraRotationY, JoystickRotation) * movement;
        if (movement != Vector3.zero)
        {
            Quaternion newRotation = Quaternion.LookRotation(movement);
            rb.rotation = Quaternion.Lerp(rb.rotation, newRotation, Time.deltaTime * rotationSpeed);
        }
    }
}
```

Úryvek kódu 2 Ukázka kódu pro pohyb auta pomocí joysticku.

6.2.3 PlayerInputController.cs

Tento skript s názvem „PlayerInputController“ je klíčovým prvkem pro ovládání pohybu hráčů ve hře. Jeho hlavním účelem je zachytávání vstupů od hráče a při jejich zachycení volání metod ze tříd „CarController“ a „CarControllerJoystick“, názorně zobrazeno na úryvku kódu 3. Skript dále obsahuje metody Bind, které slouží k propojení s hlavní řídicí třídou auta („CarController“) a třídou pro ovládání auta pomocí joysticku („CarControllerJoystick“). Tímto způsobem může „PlayerInputController“ předávat akce ovládání pohybu auta těmto skriptům. V metodě „FixedUpdate“ jsou vyhodnocovány stisky kláves a na základě těchto stisků jsou volány příslušné metody pro pohyb auta („Accelerate“, „Reverse“, „TurnLeft“, „TurnRight“). Pokud je aktivní ovládání joystickem, předávají se akce ovládání pohybu auta skriptu „CarControllerJoystick“.

```
0 references
void FixedUpdate()
{
    if (carController == null)
    {
        return;
    }
    if (accelerate) // Podmínka bude pravda, pokud je tlačítko pro zrychlení stisknuto
    {
        carControllerJoystick.JoystickMovement(0);
        carController.Accelerate();
    }
    if (reverse) // Podmínka bude pravda, pokud je tlačítko pro couvání stisknuto
    {
        carControllerJoystick.JoystickMovement(180);
        carController.Reverse();
    }
    if (turnLeft) // Podmínka bude pravda, pokud je tlačítko pro otočení doleva stisknuto
    {
        carController.TurnLeft();
    }
    if (turnRight) // Podmínka bude pravda, pokud je tlačítko pro otočení doprava stisknuto
    {
        carController.TurnRight();
    }
}
```

Úryvek kódu 3 Ukázka kódu pro volání metod ze tříd „CarController“ a „CarControllerJoystick“.

6.2.4 HideShowUI.cs a Options.cs

Skript „Options“ zobrazuje nastavení, kde umožňuje hráči volit mezi dvěma různými způsoby ovládání pohybu auta. Obsahuje dropdown menu, které umožňuje hráči vybrat mezi joystickem a šípkami používanými pro ovládání virtuálního auta. Při startu hry na základě předchozího výběru nastaví viditelnost joysticku nebo šipek a aktualizuje hodnoty dropdown menu podle aktuálního výběru. Metoda „HandleDropDown“ se volá při změně výběru v dropdown menu a podle nového výběru aktualizuje viditelnost ovládacích prvků a ukládá nový výběr do paměti.

Skript „HideShowUI“ dědí z třídy Options a zavolá se například vždy, když je kliknuto na tlačítko „OPTIONS“ v menu, nebo když je kliknuto na tlačítko „BACK“ v nastavení. Ve zkratce řídí skrytí a zobrazení uživatelského rozhraní v menu nebo ve hře. Obsahuje metodu „HideShowUIPanel“, která po zavolání buď skryje nebo zobrazí UI panel.

6.3 Vytváření mapy a umístění auta na detekovanou rovinu

Tato část textu se zaměří na skripty, které se starají o vytváření mapy a umístění auta na detekovanou rovinu. Všechny tyto skripty se nachází ve složce „Assets/Scripts/MapLogic/“.

6.3.1 ScanCheckpoints.cs

Skript „ScanCheckpoints“ má za úkol zakázat ARTrackedImageManager, poté co hráč naskenuje všechny markery a stiskne tlačítko „Scanned“. ARTrackedImageManager je komponenta, která spravuje sledování obrázků v prostoru AR. Tím že se zakáže je zajištěno to, že se pozice bran již nebude nadále měnit.

6.3.2 ImageTracking.cs

Skript „ImageTracking“ je určen k řízení sledování a umístování bran na základě detekce markerů. Při inicializaci skriptu se nastaví reference na ARTrackedImageManager. Dále se vytvoří instance bran, které mohou být umístěny na detekované markery. Tyto objekty jsou vytvořeny jako instance prefabů a jsou uloženy do slovníku podle jejich názvů. Skript také obsahuje metodu „ImageChanged“, která se volá při každé změně ve sledování markeru a zpracovává přidané, aktualizované a odebrané markery (úryvek kódu 4).

```
2 references
private void ImageChanged(ARTrackedImagesChangedEventArgs eventArgs)
{
    foreach (ARTrackedImage trackedImage in eventArgs.added)
    {
        UpdateImage(trackedImage);
    }
    foreach (ARTrackedImage trackedImage in eventArgs.updated)
    {
        UpdateImage(trackedImage);
    }
    foreach (ARTrackedImage trackedImage in eventArgs.removed)
    {
        spawnedObjects[trackedImage.name].SetActive(false);
        CheckpointController.placedObjectNames.Remove(trackedImage.name);
    }
}

2 references
private void UpdateImage(ARTrackedImage trackedImage)
{
    string name = trackedImage.referenceImage.name;
    Vector3 position = trackedImage.transform.position;
    GameObject prefab = spawnedObjects[name];
    prefab.transform.position = position;

    Quaternion rotation = Quaternion.Euler(0, trackedImage.transform.rotation.eulerAngles.y, 0);
    prefab.transform.rotation = rotation;

    prefab.SetActive(true);

    if (!CheckpointController.placedObjectNames.ContainsKey(name))
    {
        CheckpointController.placedObjectNames.Add(name, false);
    }
}
```

Úryvek kódu 4 Znárodně volání metody „ImageChanged“, při přidání, aktualizování nebo odstranění markeru.

6.3.3 CheckpointController.cs

Skript „CheckpointController“ se stará o detekci dotyku checkpointu s autem, přičítání kol, zobrazování nápověd hráči a stopek, které měří čas strávený odjetím závodu. O detekci dotyku auta s checkpointem, se stará funkce „OnTriggerEnter“, což je funkce z knihovny Unity, která se spustí při kolizi Collideru objektu na kterém se script nachází, s jakýmkoli jiným colliderem. V mém případě se sráží collider checkpointů (checkpoint je objekt vložený v objektu brány) s colliderem auta. Pokud dojde ke srážce těchto dvou objektů, funkce zkontroluje, zda byl první zasažen checkpoint červené branky. Pokud ne, bude čekat, dokud k tomu nedojde. Při jeho projetí se spustí stopky a akce se opakuje se všemi ostatními checkpointy. Až je celá trať projetá auto musí znovu projet checkpointem červené branky. Když se to stane, přičte se odjeté kolo a celá akce počítání checkpointů se

opakuje dokud hráč neobjede celou trať 3x. Při projetí celého závodu funkce zobrazí výsledný čas projetí dráhy, celý tento proces je znázorněn na úryvku kódu 5.

```
0 references
private void OnTriggerEnter(Collider other)
{
    GameObject object1 = gameObject;
    string parentObjectName = object1.transform.parent.name;
    if (parentObjectName == "Start" && placedObjectNames[parentObjectName] == false)
    {
        warning.SetActive(false);
        placedObjectNames[parentObjectName] = true;
        if (!raceStarted)
        {
            raceStarted = true;
            raceStartTime = Time.time;
        }
    }
    else if (placedObjectNames.ContainsKey(parentObjectName) && placedObjectNames["Start"] == true)
    {
        if (placedObjectNames[parentObjectName] == false)
        {
            placedObjectNames[parentObjectName] = true;
        }
        else
        {
            bool allTrue = true;
            foreach (bool value in placedObjectNames.Values)
            {
                if (!value)
                {
                    allTrue = false;
                    break;
                }
            }
            if (allTrue)
            {
                if (parentObjectName == "Start" && laps < 3)
                {
                    laps += 1;
                    lapsText.text = laps + "/3";
                    ResetPlacedObjectNames();
                }
                else if (parentObjectName == "Start" && laps == 3)
                {
                    warningText.text = "You have completed the race in " + timerText.text;
                    warningText.color = Color.green;
                    raceStarted = false;
                    warning.SetActive(true);
                    StartAgainGameObject.SetActive(true);
                }
            }
        }
    }
}
else
{
    warningText.text = "You need to go through the first gate!";
    warningText.color = Color.red;
    warning.SetActive(true);
}
}
```

Úryvek kódu 5 Ukázka logiky záchytných bodů.

6.3.4 PlacementOnMesh.cs

Hlavním účelem skriptu „PlacementOnMesh“ je umožnit hráči umístit vybrané vozidlo na AR plochu a přizpůsobit jeho škálování a rotaci. Ve funkci "Awake" se inicializuje materiál pro umísťovaný objekt podle aktuálního indexu materiálu. Ve funkci "Start" se vybere vozidlo podle proměnné "chosenCar" z předchozí scény a přiřadí se do proměnné "chosenCarObject". V "Update" funkci jsou obsaženy podmínky pro vstupy z myši pro testování přímo v enginu Unity nebo dotyku obrazovky mobilního zařízení a vytvoření paprsku z kamery směrem k ploše AR. Pokud nebyl objekt již umístěn a paprsek koliduje s nějakým objektem v prostoru, provede se umístění vybraného vozidla na tuto pozici.

7 TESTOVÁNÍ

K testování vytvořené hry byla přizvána autorova rodina, tak aby poskytla zpětnou vazbu z různých perspektiv. Testování se zúčastnili otec, matka a sestra autora, kteří mají rozdílné úrovně zkušeností s videohrami. Před testováním byla hra nainstalována na mobilní telefon Samsung Galaxy a54 s operačním systémem Android a osm markerů rozmístěno po zemi ve vzdálenosti jednoho metru od sebe.

Testování probíhalo tak, že autor předal mobilní telefon s hrou jednomu z členů rodiny, který ji hrál poprvé, a neposkytl mu žádné instrukce, aby zjistil, jak si poradí s prostředím hry. To mělo zajistit, že jeho reakce na hru bude spontánní.

Jako první se do testování zapojila autorova matka. Narazila na problém s rozpoznáváním markerů, jelikož nebylo zcela jasné, kdy kamera marker detekuje a zobrazí bránu. Také nikdy předtím nehrála žádnou videohru, což jí způsobovalo potíže s ovládním auta pomocí joysticku. Po přepnutí ovládním na šipky uvedla, že toto ovládním je matoucí, zvláště, když je kapota auta otočena jiným směrem než hráč. Všechna 3 kola projela za 5 minut a 10 sekund. Na závěr dodala, že to pro ni byla zcela nová zkušenost a že nikdy předtím nic podobného nezažila.

Dalším účastníkem byl otec autora, který si hned při prohlížení menu všiml absence popisu vlastností vozidel. Dále mu vadilo, že celá navigace hry je v angličtině, což mu nevyhovovalo. Všechna 3 kola tratě projel za 2 minuty a 2 sekundy. Hru zhodnotil kladně a ocenil inovativní přístup, jelikož hra vyžaduje od hráče fyzickou aktivitu a ne jen pasivní sedění u obrazovky jako většina her.

Poslední účastnicí byla sestra autora, které stejně jako matce vadilo, že není jasné, kdy se při skenování zobrazí na markeru brána. S angličtinou ani s ovládním auta však neměla žádný problém. Dokázala celou trať dokončit za pouhých 46 sekund.

7.1 Poznatky z testování

Z testování vyplívá, že se hra stále nenachází ve stádiu, ve kterém by mohla být publikována na platformu Google Play. Obsahuje totiž drobné chyby, které by neměla mít při vydání.

8 POUŽITÉ ASSETY

Při vytváření hry jsem využil několik zdarma dostupných assetů z internetu a z Unity Asset Store. Díky tomuto rozhodnutí jsem mohl lépe soustředit svou pozornost na samotnou implementaci hry, aniž bych musel vynakládat čas na tvorbu vlastních 3D modelů, což by přesahovalo rozsah této práce.

Jedním z využitých assetů je Car Kit [28], nabízející velké množství low poly autíček, které se přesně hodí do stylu mé hry.

Další využitý balíček assetů je Racing Kit [28], ze kterého jsem využil několik modelů k vytvoření hlavního menu.

Posledním z využitých assetů jsou Environment Track Lowpoly: Cartoon Props Mobile Free [29], jehož brány se ve hře zobrazí po naskenování jednotlivých markerů.

ZÁVĚR

Cílem této bakalářské práce byla implementace hry v enginu Unity za použití rozšířené reality. Před samotným začátkem vývoje bylo nezbytné podrobně se seznámit s rozšířenou realitou a prozkoumat její základy a principy fungování. Tato fáze zahrnovala studium technologií a konceptů spojených s AR, aby bylo možné efektivně využít potenciál této technologie při tvorbě hry.

Dalším krokem bylo seznámení se s multiplatformním herním enginem Unity, který představuje klíčový nástroj pro vývoj her. Bylo zásadní porozumět základním principům práce a vývoje her na této platformě, neboť Unity poskytuje širokou škálu nástrojů a funkcí pro tvorbu her od základních až po pokročilé. Důkladné pochopení tohoto enginu bylo klíčové pro úspěšnou implementaci rozšířené reality do projektu. Poté bylo důležité se zaměřit na vyhledání nástrojů, se kterými lze efektivně vytvářet hry s využitím rozšířené reality. Tento proces vyžadoval detailní studium dokumentace, průzkum dostupných pluginů a experimentování s různými technikami implementace.

Výsledným produktem této práce je inovativní závodní hra, která využívá technologii rozšířené reality. Ve hře má hráč možnost postavení dráhy na kterékoliv ploché zemi. Ve hře má hráč taky na výběr ze dvou možností ovládání auta. Různé modely využití v této hře byly staženy z Asset Store a KEENEY.nl. Výsledná hra funguje na všech zařízeních Android s podporou ARCore.

Nyní hra přešla do fáze průběžného testování, ve které zatím pomáhají jen rodinní příslušníci autora. Na základě jejich interakce s hrou budou opraveny stávající chyby a nadále budou probíhat úpravy a vylepšení tak, aby byla hra co nejvíce uživatelsky přívětivá. V pokročilejším stádiu vývoje bude hra publikována na platformě Google Play, kde bude dostupná pro širokou veřejnost a dále upravována podle požadavků uživatelů.

Od samého začátku vývoje byl kód psaný tak, aby byla možnost dalšího rozšíření kódu, což by umožnilo snadné přidávání dalšího obsahu do hry. V budoucnu je v plánu přidání většího výběru aut, skokánek nebo překážek, různých vylepšení aut a hlavně přidání režimu pro více hráčů propojených pomocí internetové sítě. Dále by bylo potřeba hru celkově zlepšit z grafické části a to uživatelské rozhraní a modely tratě. V poslední řadě by bylo vhodné hru publikovat i na platformu IOS pro získání většího počtu uživatelů.

SEZNAM POUŽITÉ LITERATURY

- [1] STATISTA. *Augmented reality (AR) - statistics & facts*. Online. ALSOP, Thomas. Statista.com. 2024, last modified 10 January 2024. Dostupné z: <https://www.statista.com/topics/3286/augmented-reality-ar/#topicOverview>. [cit. 2024-02-15].
- [2] S. L. Kim, H. J. Suk, J. H. Kang, J. M. Jung, T. H. Laine and J. Westlin et al. Using Unity 3D to facilitate mobile augmented reality game development. Online. *2014 IEEE World Forum on Internet of Things (WF-IoT)*. 2014, roč. 2014, č. 1, s. 21-25. Dostupné z: IEEE, <https://doi.org/10.1109/WF-IoT.2014.6803110>. [cit. 2024-02-15].
- [3] P. Milgram, H. Takemura, A. Utsumi, and F. Kishino, "Augmented reality: A class of displays on the reality-virtuality continuum," in *Telemanipulator and telepresence technologies*, vol. 2351. International Society for Optics and Photonics, 1995, pp. 282–293.
- [4] H. Ardiny and E. Khanmirza, "The Role of AR and VR Technologies in Education Developments: Opportunities and Challenges," *2018 6th RSI International Conference on Robotics and Mechatronics (ICRoM)*, Tehran, Iran, 2018, pp. 482-487, doi: 10.1109/ICRoM.2018.8657615.
- [5] IVANOVA, A. V. VR & AR TECHNOLOGIES: OPPORTUNITIES AND APPLICATION OBSTACLES. Online. Strategic decisions and risk management. 2018, roč. 2018, č. 3, s. 88-107. Dostupné z: <https://doi.org/https://doi.org/10.17747/2078-8886-2018-3-88-107>. [cit. 2024-02-15].
- [6] CHEN, Y.; WANG, Q.; CHEN, H.; SONG, X.; TANG, H. et al. An overview of augmented reality technology. Online. *Journal of Physics: Conference Series*. 2019, roč. 2019, conf č. 1237, s. 1-5. Dostupné z: <https://doi.org/10.1088/1742-6596/1237/2/022082>. [cit. 2024-02-20].
- [7] ALKHAMISI, Abrar Omar a Muhammad Mostafa MONOWAR. Rise of Augmented Reality: Current and Future Application Areas. *International Journal of Internet and Distributed Systems* [online]. 2013, **01**(04), 25-34 [cit. 2024-05-07]. ISSN 2327-7157. Dostupné z: doi:10.4236/ijids.2013.14005
- [8] GRUBERT, Jens. Augmented Reality for Android Application Development. [online] 2013. ISBN 1782168559. [cit. 2016-05-14]. Available at: <http://site.ebrary.com/>

- [9] Towards Indistinguishable Augmented Reality: ... *ACM Computing Surveys* [online]. 2021, **2021**(54(6)), 1-36 [cit. 2024-02-27]. Dostupné z: doi:10.1145/3453157
- [10] CHENG, Jack C. P., Keyu CHEN a Weiwei CHEN. *Comparison of marker-based AR and markerless AR: A case study on indoor decoration system* [online]. Vol. 2. Heraklion, Greece: Lean & Computing in Construction Congress (LC3), 2017 [cit. 2024-02-28]. Dostupné z: doi:10.24928/JC3-2017/0231.
- [11] KHAN, Akif, Shah KHUSRO, Azhar RAUF a Saeed MAHFOOZ. Rebirth of augmented reality-enhancing reality via smartphones. *Bahria University Journal of Information & Communication Technologies* [online]. Bahria Univ. J. Inf. Commun. Technol, 2015, **2015**(8), 110-121 [cit. 2024-02-28]. ISSN 1999-4974. Dostupné z: Článek je dostupný v databázi Researchgate
- [12] DONG, Xuanyi, Liang ZHENG, Fan MA, Yi YANG a Deyu MENG. Few-Example Object Detection with Model Communication. *IEEE Transactions on Pattern Analysis and Machine Intelligence* [online]. 2019, 2019-7-1, **41**(7), 1641-1654 [cit. 2024-05-06]. ISSN 0162-8828. Dostupné z: doi:10.1109/TPAMI.2018.2844853
- [13] STEJSKAL, Bc. Ondřej. Rozšířená realita a její využití v současné době [online]. Brno, 2016 [cit. 2024-04-12]. Dostupné z: https://is.muni.cz/th/x7r8p/MAGISTERSKA_DIPLOMOVA_PRACE_-_STEJSKAL__383921_DOC_final.pdf. Magisterská diplomová práce. Masarykova univerzita. Vedoucí práce Mgr. Tomáš Staudek, Ph.D.
- [14] VANARAMA. 4 Ways Augmented Reality Will Revolutionise the Automotive Industry. VANARAMA. *Vanarama* [online]. c2023 [cit. 2024-04-12]. Dostupné z: <https://www.vanarama.com/blog/cars/4-ways-augmented-reality-will-revolutionise-the-automotive-industry>
- [15] ARCHANGELSKAYA, Anastasia, Ivan GERASIMOV, Mariam Al SARDAR a Anna ABRAMOVA. *City AR: Augmented reality navigation in the smart cities infrastructure* [online]. Cyprus: IEEE, 2022 [cit. 2024-04-12]. ISBN 978-1-6654-8562-3. ISSN 2687-8852. Dostupné z: doi:10.1109/ISC255366.2022.9921884
- [16] GOOGLE LLC IPA. Google Translator. *Google Play* [online]. 2012 [cit. 2024-05-02]. Dostupné z: https://play.google.com/store/search?q=google%20translator&c=apps&hl=en_US

- [17] GOOGLE LLC IPA. Snapchat. *Google Play* [online]. 2012 [cit. 2024-05-02]. Dostupné z: https://play.google.com/store/apps/details?id=com.snapchat.android&hl=en_US
- [18] CHONG, Y, DK SETHI, CHY LOH a F LATEEF. Going Forward with Pokemon Go. *J Emerg Trauma Shock*. [online]. 2018, 2018(11(4), 243-246 [cit. 2024-02-25]. Dostupné z: doi:10.4103/JETS.JETS_87_17
- [19] RUIZ-ARIZA, Alberto, Sebastián LÓPEZ-SERRANO, Sara SUÁREZ-MANZANO a Emilio J. MARTÍNEZ-LÓPEZ. The Educational Use of the ‘Harry Potter: Wizards Unite’ Augmented Reality Application. In: *Augmented Reality in Education* [online]. 1. Springer, Cham, 2020, s. 247–259 [cit. 2024-02-26]. ISBN 978-3-030-42156-4. Dostupné z: https://link.springer.com/chapter/10.1007/978-3-030-42156-4_14#Abs1
- [20] BIGAS, Jiří. VORTEX MEDIA S.R.O. Hra Harry Potter: Wizards Unite od Nianticu končí. *VORTEX* [online]. 2019 [cit. 2024-02-27]. Dostupné z: <https://www.vortex.cz/hra-harry-potter-wizards-unite-od-nianticu-konci/>
- [21] APPLE INC. *App Store* [online]. [2008] [cit. 2024-04-09]. Dostupné z: <https://www.apple.com/app-store/>
- [22] *Cross-platform application development using unity game engine* [online]. Vol. 3. Púne: IJARCSMS, 2015 [cit. 2024-05-07]. ISSN 2321-7782. Dostupné z: https://www.researchgate.net/profile/Pratik-Patil-11/publication/312591645_Cross-platform_Application_Development_using_Unity_Game_Engine/links/5884d39faca272b7b44a847f/Cross-platform-Application-Development-using-Unity-Game-Engine.pdf
- [23] UNITY TECHNOLOGIES. What subscription tiers are available? UNITY TECHNOLOGIES. *Unity Support* [online]. 2005 [cit. 2024-05-07]. Dostupné z: <https://support.unity.com/hc/en-us/articles/208610336-What-subscription-tiers-are-available>
- [24] OUFQIR, Z., A. EL ABDERRAHMANI a K. SATORI. *ARKit and ARCore in serve to augmented reality* [online]. 2020 International Conference on Intelligent Systems and Computer Vision (ISCV): IEEE, 2020 [cit. 2024-04-02]. ISBN 978-1-7281-8041-0. Dostupné z: doi:10.1109/ISCV49265.2020.9204243

- [25] LINOWES, Jonathan; BABILINSKI, Krystian. *Augmented reality for developers: Build practical augmented reality applications with unity, ARCore, ARKit, and Vuforia*. Packt Publishing Ltd, 2017. [cit. 2024-04-03]. Dostupné z: <https://books.google.cz/>
- [26] UNITY TECHNOLOGIES. *Unity User Manual* [online]. c2024, 12.3.2024 [cit. 2024-04-03]. Dostupné z: <https://docs.unity3d.com/>
- [27] DENSO WAVE INCORPORATED. *QR.io* [online]. 1976 [cit. 2024-05-09]. Dostupné z: <https://qr.io/>
- [28] KENNEY [online]. 2010, 2024 [cit. 2024-05-08]. Dostupné z: <https://kenney.nl/>
- [29] UNITY TECHNOLOGIES. Environment Track Lowpoly: Cartoon Props Mobile Free. UNITY TECHNOLOGIES. *Asset Store* [online]. 2005 [cit. 2024-05-08]. Dostupné z: <https://assetstore.unity.com/packages/3d/props/environment-track-lowpoly-cartoon-props-mobile-free-211494>

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

AR	Rozšířená realita
VR	Virtuální realita
OST	Optická průhledová technologie
VST	Technologie průhledného videa
SDK	Software Development Kit (Vývojový balík softwaru)
QR	Quick Response (Rychlá odpověď)
ID	Identifikace
IOS	Operační systém pro mobilní zařízení od společnosti Apple
GPS	Globální polohový systém
PvP	Hráč proti hráči
AI	Umělá inteligence
IMU	Inerciální Měřící Jednotka
USD	Americký dolar
VIO	Vizuálně-Inerciální odometr
Fbx	Formát souboru používaný pro 3D modely
glTF	Formát souboru pro grafickou výměnu
GPL	General Public Licence (Obecná veřejná licence)
URL	Uniform Resource Locator (Adresa webové stránky)

SEZNAM OBRÁZKŮ

Obrázek 1 Model hybridní reality [3]	12
Obrázek 2 Nejčastěji způsoby zobrazení AR. Zdroj: autor neznámý, s. 3, [9].....	14
Obrázek 3 Ukázka různých 2D-Barcode markerů používaných v průmyslu	15
Obrázek 4 Chytání pokémonů ve hře Pokemon GO [21]	19
Obrázek 5 Ukázka prostředí hry Harry Potter: Wizzards Unite [19].	20
Obrázek 6 Ar Basketball [21]	21
Obrázek 7 Virtuální dinosaur v reálném prostředí ve hře Jurassic World Alive [21].	21
Obrázek 8 Zleva doprava – Kings of Pool, Stack AR, Brickscape [21].	22
Obrázek 9 Životní cyklus komponent	23
Obrázek 10 Virtuální objekt umístěný na reálnou podlahu pomocí ARCore.	26
Obrázek 11 Objekt umístěn na naskenovaném obrázku pomocí ARCore. Zdroj: autor neznámý, s.5, [23]	27
Obrázek 12 Rozpoznávání obličeje pomocí ARCore. Zdroj: autor neznámý, s.5, [23]	27
Obrázek 13 Umístění objektu na naskenovanou podlahu pomocí ARKit.	28
Obrázek 14 Objekt umístěný na obrázek pomocí ARKit. Zdroj: autor neznámý, s.3, [23] ..	29
Obrázek 15 Skenování objektu pomocí ARKit. Zdroj: autor neznámý, s.3, [23]	30
Obrázek 16 Screenshot pořízený ze hry – Vytvořená mapa pomocí naskenovaných markerů	35
Obrázek 17 Vyskakovací okno s nastavením ve scéně MainMenu	37
Obrázek 18 Scéna MainMenu	38
Obrázek 19 Scéna GameScene	38
Obrázek 20 Ukázka vytvořených markerů.	39

SEZNAM ÚRYVKŮ KÓDU

Úryvek kódu 1 Ukázka kódu pro pohyb auta pomocí šipek.....	42
Úryvek kódu 2 Ukázka kódu pro pohyb auta pomocí joysticku.....	43
Úryvek kódu 3 Ukázka kódu pro volání metod ze tříd „CarController“ a „CarControllerJoystick“.....	44
Úryvek kódu 4 Znázorněné volání metody „ImageChanged“, při přidání, aktualizování nebo odstranění markeru.....	46
Úryvek kódu 5 Ukázka logiky záchytných bodů.....	47

SEZNAM TABULEK

Tabulka 1 Porovnání nástrojových sad [23][24][25].....	31
--	----

SEZNAM PŘÍLOH

Příloha P I: Unity projekt ve formátu .zip

Příloha P II: Markery používané ke generování mapy ve formátu .zip

PŘÍLOHA P I: UNITY PROJEKT VE FORMÁTU .ZIP

Soubor s názvem „UnityGame.zip“ obsahuje celý projekt spustitelný v enginu Unity, včetně zdrojového kódu.

PŘÍLOHA P II: MARKERY POUŽÍVANÉ KE GENEROVÁNÍ MAPY VE FORMÁTU .ZIP

Soubor s názvem „Markers.zip“ obsahuje markery používané ke generování bran.