

Univerzita Pardubice
Fakulta elektrotechniky a informatiky

Návrh a realizace mobilní aplikace využívající rozšířenou realitu
Bakalářská práce

2021

Petr Dolejš

Univerzita Pardubice
Fakulta elektrotechniky a informatiky
Akademický rok: 2020/2021

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: **Petr Dolejš**
Osobní číslo: **I18115**
Studijní program: **B2646 Informační technologie**
Studijní obor: **Informační technologie**
Téma práce: **Návrh a realizace mobilní aplikace využívající rozšířenou realitu**
Zadávající katedra: **Katedra informačních technologií**

Zásady pro vypracování

Cílem bakalářské práce bude v teoretické části charakterizovat a popsat rozšířenou realitu a analyzovat její využitelnost v mobilních aplikacích, zejména existující API pro rozšířenou realitu (např. ARCore, ARKit). Součástí práce bude i přehled identifikačních a lokačních bezdrátových technologií potřebných pro umístování digitálních objektů v obrazu reálného světa na displeji mobilního zařízení v rozšířené realitě, práce identifikuje jejich výhody a nevýhody.

Druhým cílem v praktické části bakalářské práce bude navrhnout a realizovat mobilní aplikaci (volitelně pro Android/iOS/Windows mobile) či mobilní webovou aplikaci v HTML5. Vlastní aplikace bude využívat identifikačních (např. QR kódy) a lokačních a pohybových technologií (např. GPS/GALILEO a elektronický kompas, gyroskop a akcelerometr) pro uživatelské rozhraní s rozšířenou realitou.

Rozsah pracovní zprávy: **35**
Rozsah grafických prací:
Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam doporučené literatury:

- TACGIN, ZEYNEP. *Virtual and augmented reality: An Educational Handbook*. Cambridge Scholars Publishing: 2020. ISBN 1-5275-4813-9.
- SOOD, RAGHAV. *Pro Android augmented reality*. New York: Apress, c2012, xvi, 329 p. ISBN 143023945x.
- ALAN B. CRAIG. *Understanding augmented reality concepts and applications*. Online-Ausg. Amsterdam: Morgan Kaufmann, 2013. ISBN 9780240824109.
- GREGORY KIPPER, Joseph Rampolla. *Augmented Reality an Emerging Technologies Guide to AR*. 1st ed. Rockland, MA: Elsevier Science, 2012. ISBN 9781597497343.

Vedoucí bakalářské práce: **Ing. Jiří Kysela, Ph.D.**
Katedra informačních technologií

Datum zadání bakalářské práce: **31. října 2020**
Termín odevzdání bakalářské práce: **14. května 2021**

L.S.

Ing. Zdeněk Němec, Ph.D. v.r.
děkan

Ing. Jan Panuš, Ph.D. v.r.
vedoucí katedry

PROHLÁŠENÍ

Prohlašuji:

Práci s názvem Návrh a realizace mobilní aplikace využívající rozšířenou realitu jsem vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Beru na vědomí, že v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů, a směrnicí Univerzity Pardubice č. 7/2019 Pravidla pro odevzdávání, zveřejňování a formální úpravu závěrečných prací, ve znění pozdějších dodatků, bude práce zveřejněna prostřednictvím Digitální knihovny Univerzity Pardubice.

V Pardubicích dne 29. 04. 2021

Petr Dolejš v.r

PODĚKOVÁNÍ

Rád bych touto cestou vyjádřil poděkování Ing. Jiřímu Kyselovi, Ph.D., za jeho cenné rady, doporučení a trpělivost při vedení mé bakalářské práce. Taktéž bych chtěl poděkovat všem, kteří mě podporovali, zejména rodičům a sourozenci.

ANOTACE

Tato bakalářská práce se zaměřuje na charakterizaci rozšířené reality a její využitelnost. V práci jsou představeny vybrané identifikační a lokační bezdrátové technologie. Dále jsou popsány 3D objekty, je podrobně popsán návrh aplikace a samotný vývoj aplikace, která využívá ARCore pro zobrazení objektů ve skutečném světě. Tato aplikace je určena pro zařízení s operačním systémem Android, který je v práci rovněž představen. Závěrem je popsáno testování aplikace.

KLÍČOVÁ SLOVA

rozšířená realita, mobilní aplikace, identifikační a lokační bezdrátové technologie, Android, ARCore

TITLE

Design and implementation of a mobile application using augmented reality

ANNOTATION

This bachelor thesis focuses on the characterization of augmented reality and its use. The work presents selected identification and location wireless technologies. Next, 3D objects are described, the design of the application and the development of applications that require ARCore to display objects in the existing world are described in detail. This application is intended for devices with the Android operating system, which is further presented in the work. Finally, application testing is described.

KEYWORDS

augmented reality, mobile application, identification and location wireless technologies, Android, ARCore

OBSAH

Seznam obrázků	10
Seznam tabulek	10
Úvod	11
1. Rozšířená realita	12
1.1. Historie	12
1.2. AR vs VR	12
1.3. Využitelnost	13
1.3.1 Využití ve sportu	13
1.3.2 Využití v nakupování	14
1.3.3 Využití v herním průmyslu a sociálních sítích	15
1.4. API k rozšířené realitě	16
1.4.1 ARCore	16
1.4.2 ARKit	17
2. Identifikační a lokační bezdrátové technologie	18
2.1. Lokační bezdrátové technologie	18
2.1.1 GPS	18
2.1.2 Galileo	18
2.1.3 BeiDou	19
2.1.4 GLONASS	19
2.1.5 WiFi	20
2.2. Identifikační bezdrátové technologie	20
2.2.1 RFID	20
2.2.2 NFC	21
2.2.3 Čárové kódy	21
2.2.4 QR kód	22
2.2.5 Biometrie	23
3. Návrh aplikace	24
3.1. 3D modely	24
3.2. Operační systém Android	24
3.2.1 Aplikační vrstva	25
3.2.2 Aplikační framework	25
3.2.3 Android Runtime	25
3.2.4 Knihovny	26
3.2.5 Linux Kernel	26

3.3.	Analýza	26
3.3.1.	Funkční požadavky	26
3.3.2.	Nefunkční požadavky	26
3.3.3.	Use case diagram	27
3.3.4.	Use case specifikace	29
3.3.5.	Diagram aktivit	30
3.4.	Uživatelské rozhraní aplikace	31
3.4.1.	Použité layouty	31
3.4.1.1	ConstraintLayout	32
3.4.1.2	FrameLayout	32
3.4.2.	Použité komponenty	32
3.4.2.1	Aktivita	32
3.4.2.2	ArFragment	33
3.4.2.3	TextView	33
3.4.2.4	ProgressBar	33
3.4.2.5	ListView	34
3.4.2.6	Button	34
3.4.2.7	ImageView	34
3.4.3.	Celkový návrh vzhledu	34
4.	Vývoj aplikace	38
4.1.	Použité technologie	38
4.1.1.	Android studio	38
4.1.2.	Java	38
4.1.3.	ARCore	38
4.2.	Struktura aplikace	39
4.2.1.	Balíčky	39
4.2.2.	Třídy	39
4.3.	Rozbor kódu	40
4.3.1	SceneForm/ArCore	40
4.3.2	Manifest	41
4.3.3	Zobrazení modelu	41
4.4.	Uživatelská příručka	44
4.4.1	Hlavní stránka	44
4.4.2	Kvíz	45
4.4.3	Knihovna	45
4.4.4	O aplikaci	46

5. Testování	47
5.1. Testování vývojářem	47
5.2. Testování uživatelem	47
5.3. Výsledky testování	47
Závěr.....	48
Použitá literatura.....	49
Obsah CD.....	58

SEZNAM OBRÁZKŮ

Obrázek 1: Studio Sky Sports. Zdroj [8].	13
Obrázek 2: Aplikace IKEA Place. Zdroj [14].	14
Obrázek 3: Aplikace Pokémon GO. Zdroj [19].	15
Obrázek 4: Filtry v aplikacích Facebook, Instagram a Snapchat. Zdroje [24], [25], [26].	16
Obrázek 5: Čárový kód EAN13. Zdroj [50].	22
Obrázek 6: QR kód. Zdroj: vlastní.	22
Obrázek 7: Android architektura. Zdroj [61].	25
Obrázek 8: Use Case diagram. Zdroj: vlastní.	28
Obrázek 9: Diagram aktivit. Zdroj: vlastní.	31
Obrázek 10: Životní cyklus aktivity. Zdroj [73].	33
Obrázek 11: Návrh úvodní strany. Zdroj: vlastní.	35
Obrázek 12: Návrh stran v kvízu. Zdroj: vlastní.	36
Obrázek 13: Návrh stran v knihovně. Zdroj: vlastní.	37
Obrázek 14: Návrh strany o aplikaci. Zdroj: vlastní.	37
Obrázek 15: Úvodní obrazovka. Zdroj: vlastní.	44
Obrázek 16: Obrazovky v části s kvízem. Zdroj: vlastní.	45
Obrázek 17: Obrazovky v části s knihovnou. Zdroj: vlastní.	46
Obrázek 18: Obrazovka v části O aplikaci. Zdroj: vlastní.	46

SEZNAM TABULEK

Tabulka 1: Funkční požadavky. Zdroj: vlastní.	26
Tabulka 2: Nefunkční požadavky. Zdroj: vlastní.	27
Tabulka 3: Use case specifikace – Zobrazení modelu v kvízu. Zdroj: vlastní.	29

ÚVOD

Jedním z cílů této bakalářské práce je charakterizovat a popsat rozšířenou realitu, její využitelnost a existující API pro práci s ní. Spolu s tím představit identifikační a lokační bezdrátové technologie, které jsou pro práci s rozšířenou realitou nezbytné.

Druhým cílem je navrhnout a realizovat mobilní aplikaci, která bude využívat identifikačních, lokačních a pohybových technologií pro uživatelské rozhraní s uživatelskou realitou.

V teoretické části této práce, bude čtenáři představena historie rozšířené reality, rozdíl mezi rozšířenou a virtuální realitou, její využitelnost v několika odvětvích a API, pomocí kterých můžeme tuto technologii implementovat do mobilních zařízení nebo tabletů. Budou zde také popsány identifikační bezdrátové technologie, mezi které patří GPS nebo Galileo a lokační bezdrátové technologie do kterých řadíme RFID, NFC nebo čárové kódy.

Praktická část bude popisovat návrh samotné aplikace využívající rozšířenou realitu, ve kterém bude popsána analýza aplikace a bude představeno uživatelské rozhraní aplikace. V kapitole popisující vývoj aplikace budou zmíněny použité technologie, bude představena struktura aplikace a popsány vybrané části zdrojového kódu. Také zde bude uživatelská příručka, která bude sloužit pro zjednodušení pohybu v aplikaci. V poslední kapitole budou otestovány jisté aspekty aplikace a vyhodnoceny.

1. ROZŠÍŘENÁ REALITA

1.1. Historie

Rozšířená realita se začala zkoumat mnohem dříve, než si většina lidí myslí. Už v roce 1968 bylo vyvinuto první zařízení, které se dalo přidělat na hlavu. Toto zařízení pro rozšířenou realitu má na svědomí profesor Ivan Sutherland z Harvardu a jmenovalo se The sword of Damocles. Zařízení dokázalo zobrazit virtuální obsah do reálného světa. Tehdy ještě nebyl znám termín rozšířená realita, ten zavedl až Tom Caudell v roce 1990.

V roce 1992 Louis Rosenburs vytvořil Virtual Fixtures, což byl první plnohodnotný systém pro rozšířenou realitu. Tento systém používalo americké letectvo pro bezpečnější výcvik pilotů letadel.

Dále se rozšířená realita dostala do televize, kdy se začal používat při utkáních amerického fotbal systému 1st & Ten pro zobrazení žluté čáry na hrací ploše. O rok později se opět o něco vyvinula rozšířená realita v letectví, kdy NASA vytvořila komponentu, která zobrazovala data o terénu přímo na obrazovce pilota.

Hned z kraje nového století vyvinul Kirokazu Kato softwarovou knihovnu ARToolKit, která měla pomáhat ostatním vývojářům ve vytváření dalších aplikací založených na AR. Tím se vývoj aplikací využívajících rozšířenou realitu zrychlil a vznikaly aplikace jako MARTA, která pomáhala technikům ve firmě Volkswagen při sestavování různých komponent, nebo aplikace IKEA Place, která umožňovala uživatelům prohlédnout si kusy nábytku u nich doma, ještě před samotnou koupí [1]. Microsoft v roce 2016 vyvinul brýle pro AR zvané HoloLens na které navázaly v roce 2019, kdy byly vydány HoloLens 2. Tyto brýle dokáží zobrazit virtuální objekt v reálném světě a pohybovat s nimi pomocí rukou a samotné brýle je možné ovládat pomocí očí [2] nebo hlasu [3].

1.2. AR vs VR

Rozšířená realita (AR) je technologie, která dokáže zobrazit virtuální objekt do reálného světa. Můžeme ho zobrazit například pomocí telefonu, tabletu nebo pomocí brýlí HoloLens. Pokud použijeme telefon nebo tablet, můžeme usadit objekt v podstatě kamkoliv a dívat se na něj

z různých úhlů. Kdežto HoloLens nám umožní umístit plovoucí okna a 3D dekorace kolem nás a pohybovat s nimi.

Virtuální realita (VR) na rozdíl od AR nepřidává do reálného světa virtuální objekt, ale vytváří dojem, že je uživatel úplně na jiném místě [4]. V roce 2020 byl představen nejpokročilejší VR systém pod názvem Oculus Quest 2 [5], pomocí kterého může uživatel například koukat na 360° videa nebo hrát hry, ve kterých se rozhlíží pohybem hlavy [4].

1.3. Využitelnost

1.3.1 Využití ve sportu

Jedna z oblastí využití AR je jednoznačně sport. Ve sportu se používá například pro větší přehled, zobrazení více informací nebo pro reklamní bannery. V této práci jsou uvedeny jen některé z nich.

Jak již bylo zmíněno, poprvé se AR dostalo do televize v souvislosti se sportem v americkém fotbale. Od té doby se zobrazení AR ve sportu posunulo na úplně jinou úroveň. V dnešní době se přímo na hrací plochu zobrazují různé hráčské statistiky, loga jednotlivých týmů nebo tabule se skóre [6].

Stejně jako v americkém fotbale se AR využívá i v ostatních sportech. Například v evropském fotbale se používá pro zobrazení hráčského rozestavení před zápasem, rozebrání akce ve studiu nebo pro reklamní bannery okolo hřiště [7]. V Anglii používá sportovní kanál Sky Sports AR pro zobrazení celého studia [8].

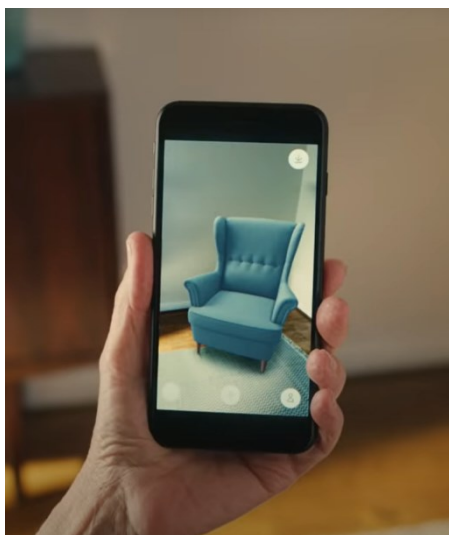


Obrázek 1: Studio Sky Sports. Zdroj [8].

AR se ve sportu využívá zejména pro větší přehled o dané situaci. Kromě již zmíněných sportů se AR také využívá v hokeji nebo závodech formulí 1.

1.3.2 Využití v nakupování

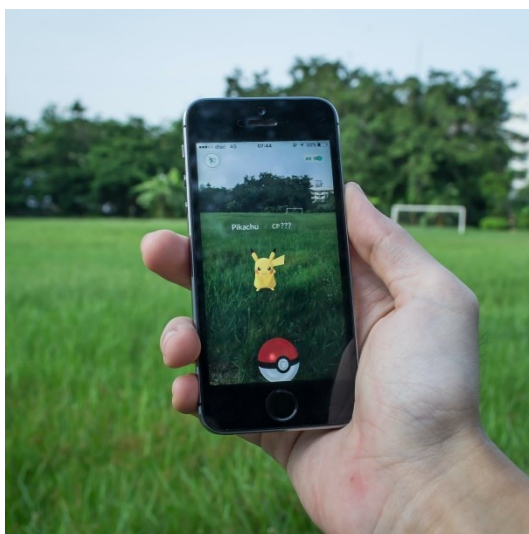
AR se také začalo dostávat do prostředí nakupování. Velkým zástupcem AR v nakupování je aplikace IKEA Place, které bylo spuštěno v roce 2017. Tato aplikace umožňuje vkládat virtuální nábytek přímo do vašeho pokoje, a tím si vyzkoušet, zda se vám nábytek líbí, a zároveň zda se vám vejde do pokoje. Celkově nabízí přes 2000 objektů k zobrazení. Aplikace využívá API ARKit a je dostupná pouze na operační systém iOS [9]. Kromě nákupů nábytku se začalo AR využívat také při nakupování oblečení. Aplikace Wanna kicks umožňuje zobrazit boty na nohou přímo z pohodlí domova [10]. Obdobně fungují sesterské aplikace Wanna nails (nehty) a Wanna watch (hodinky) [11]. Obchodní řetězec Timberland také nabízí využití AR pomocí zrcadla, které k vám (na vás) přidá kusy oblečení, dle vašeho výběru. AR v oblasti nakupování nemusíme využít jen ke zkoušení kusů oblečení. V určitých obchodech je i možnost stažení aplikace, která vám pomůže najít v obchodě zboží, které hledáte [12]. Existují také aplikace na zkoušení brýlí nebo aplikace pro dámy, která jim pomůže se správným výběrem makeupu [13].



Obrázek 2: Aplikace IKEA Place. Zdroj [14].

1.3.3 Využití v herním průmyslu a sociálních sítích

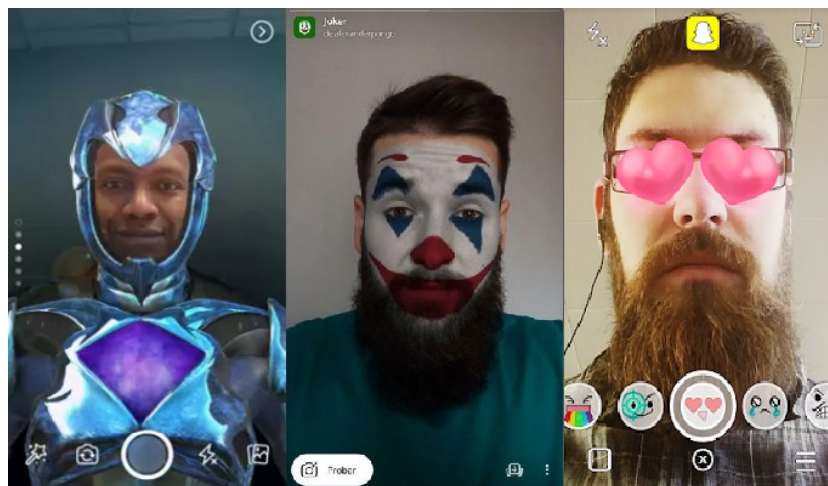
Jedním z největších odvětví, ve kterém se AR rozrostlo je herní průmysl. Existují hry, které díky AR uživatele více vtáhnou do hry, díky tomu, že mohou hrát ve skutečném prostředí, do kterého se přidají herní elementy. Jednou z takových her je Pokémon Go, která byla vydána v roce 2016 [15]. Tato hra získala na velké oblibě (500 milionů stažení během prvních tří měsíců [16]) díky své značce, nostalgii hráčů, ale také díky interakci se skutečným světem. Hra uživateli díky AR zobrazí pokémona ve skutečném světě a následně ho musí chytit [17]. Velmi podobná hra se stejným principem vyšla v roce 2019 s názvem Harry Potter: Wizards Unit [18].



Obrázek 3: Aplikace Pokémon GO.

Další hra, která byla přepracována do světa AR byla klasická hra Angry Birds s názvem Angry Birds AR: Isle of Pigs. Ve hře uživatel namíří fotoaparát na rovnou plochu, kde se zobrazí virtuální objekt, na který poté střílí zelená prasata. Hra na Android využívá API ARCore [20].

AR využívají také velké aplikace jako Facebook, Instagram nebo Snapchat. AR se do těchto aplikací dostala zejména díky aplikaci Snapchat, která je určena pro posílání fotek, a v roce 2015 umožnila uživatelům přidat efekty na obličej v reálném čase [21]. Instagram tuto funkci do aplikace zapracoval až v roce 2017 [22]. V témže roce přišel Facebook s nástrojem AR Spark, která umožňovala vybraným uživatelům navrhnout vlastní filtry na obličej a publikovat je na Facebooku nebo Instagramu. V roce 2019 tuto možnost Facebook povolil všem uživatelům [23].



Obrázek 4: Filtry v aplikacích Facebook, Instagram a Snapchat. Zdroje [24], [25], [26].

1.4.API k rozšířené realitě

API neboli Application Programming Interface je v oblasti informatiky rozhraní, které umožňuje vzájemnou interakci dvou aplikací. API pomocí definovaných metod poskytuje jednoduchý přístup k datům jiné aplikace, s kterými může dále pracovat. Typický příklad je zobrazení Google map nebo počasí v aplikace nebo na internetové stránce [27].

1.4.1 ARCore

ARCore je platforma od společnosti Google. Je určena pro zařízení s operačním systémem Android, které chtějí využívat AR pro své aplikace. Dovoluje vývojářům vytvářet aplikace, které umožňují do skutečného světa, který je zobrazený na telefonu, přidávat 3D objekty [28]. Google staví na zkušenostech z platformy Tango [28], která byla předchůdcem ARCore, ale díky potřebě drahých senzorů byl tento projekt zrušen a Google se zaměřil na ARCore, které postačí fotoaparát chytrého telefonu [29]. Tato platforma se na rozdíl od ARKitu spoléhá výhradně na zadní kameru a integrované pohybové senzory zařízení. Funguje pomocí kamery, která detekuje hlavní body. Díky těmto bodům se rozpozná rovina ve skutečném světě, na kterou můžeme vložit objekt. K možnosti pohybování telefonu okolo zobrazeného objektu, je nutné, aby ARCore měl přístup také k aktuální poloze, orientaci a pohybu mobilního zařízení [30].

1.4.2 ARKit

Platforma ARKit je vyvíjena společností Apple, která je určena pro zařízení s operačním systémem iOS využívající AR. Hlavní výhodou oproti ARCore je, že od roku 2020 využívá metodu LiDAR, která pro měření vzdálenosti objektů skutečného světa používá dobu, za jakou se puls světla vyslaný ze zařízení přesune na objekt a zpět. Každý takový puls generuje jeden bod a pomocí kolekce těchto bodů dojde k vytvoření topografické mapy okolí uživatele. Tato metoda dokáže zmapovat okolí detailněji než ARCore a také dokáže zobrazit objekty rychleji [30].

2. IDENTIFIKAČNÍ A LOKAČNÍ BEZDRÁTOVÉ TECHNOLOGIE

2.1. Lokační bezdrátové technologie

2.1.1 GPS

GPS neboli Global Positioning System je satelitní navigační systém, který dokáže určit přesnou polohu na povrchu Země. Tento systém zahrnuje 24 satelitů, které jsou ve výšce přibližně 20 000 kilometrů, a jsou rozmístěny tak, aby alespoň 4 satelity byly přímo viditelné odkudkoli na světě. Satelity vysílají zprávy, které obsahují aktuální polohu satelitu a čas. Díky těmto informacím, získaným minimálně ze tří (ideálně ze čtyř) satelitů, dokáže GPS přijímač vypočítat aktuální polohu pomocí triangulace [31]. GPS má pokrytí po celém světě a její přesnost je u mobilních zařízení do pěti metrů, přičemž se může zhoršovat v blízkosti budov nebo mostů [32].

2.1.2 Galileo

Galileo je evropský satelitní navigační systém, který má přinést alespoň částečnou nezávislost na navigačních systémech GPS nebo GLONASS. Začal se plánovat již v roce 1999 [33], ale první testovací služby byly systémem poskytnuty až v roce 2016 [34]. Je to jediný ze zde uvedených systémů, u kterého se nejedná o vojenský projekt, ale o civilní, což přináší tu výhodu, že nedojde k vypnutí při vyhocení mezinárodních vztahů. Galileo je zároveň kompatibilní jak s GPS, tak se systémem GLONASS v případě potřeby [35].

Systém by měl nabízet 4 úrovně služeb:

- Open Service (OS) – Zadarmo služba, která poskytuje vysokou přesnost do jednoho metru.
- High Accuracy Service (HAS) – Služba, poskytující další signál, dokáže zašifrovat HAS signál a pomocí toho může kontrolovat přístup do této služby [36].
- Public Regulated Service (PRS) – Služba je omezena pouze pro vládní autorizované uživatele (policie apod.) a citlivé aplikace [37].
- Search and Rescue Service (SAR) – Služba pro rychlou lokalizaci a pomoc lidem v nouzi [38].

2.1.3 BeiDou

BeiDou, podobně jako GPS nebo Galileo, je satelitní navigační systém, tentokrát vyvíjený a provozovaný Čínou. Tento systém je zaměřen na národní bezpečnost země a hospodářské a sociální rozvoje. Vývoj tohoto systému byl rozdělen do tří kroků, přičemž během prvního kroku byly služby poskytnuty po území Číny, v druhém kroku celému Asijsko-Pacifickému regionu a v roce 2020 během třetího kroku celému světu. Čína při vývoji dodržuje 4 zásady:

- Nezávislost – Označuje podporu nezávislé výstavby, vývoje a provozu systému a schopnost samostatně poskytovat služby satelitní navigace globálním uživatelům.
- Otevřenost – Označuje poskytování bezplatné služby, podpora všestrannosti a kvalitní mezinárodní spolupráce.
- Kompatibilita – Označuje vysokou kompatibilitu s jinými navigačními satelitními systémy a podporu mezinárodní spolupráce [39].
- Postupnost – Označuje, že systém bude vyvíjen krok za krokem v souladu s technickým a ekonomickým vývojem v Číně, bude docházet ke zlepšování výkonu a zajištění plynulého přechodu mezi fázemi vývoje [40].

2.1.4 GLONASS

GLONASS je ruská alternativa k americkému satelitnímu systému GPS. Tento systém je vyvíjen a provozován ruským ministerstvem obrany [41]. Aktuálně má na orbitě 24 satelitů ve výšce přibližně 19 000 km. Systém je rozdělen do tří částí:

- Vesmírná část – Tato část se skládá z 24 satelitů umístěných ve vesmíru, pomocí nich uživatelská část dokáže určit polohu.
- Pozemní část – Kontroluje stav satelitů, posuny satelitních hodin a dvakrát denně nahraje navigační data do satelitů.
- Uživatelská část – Skládá se z antén, které přijímají signály a řeší navigační rovnice za účelem získání jejich souřadnic a poskytnutí velmi přesného času.

Aktuálně GLONASS nabízí dvě služby:

- Standardní služba určování polohy – Otevřená bezplatná služba pro uživatele z celého světa.
- Služba přesného určování polohy – Je určena pro vojenské a oprávněné uživatele [42].

2.1.5 WiFi

Lokalizace pomocí WiFi se hodí především ve vnitřních prostorech, kde například GPS nebude mít tak dobrý signál kvůli nepřímému výhledu na satelity. Tato technologie využívá již existující infrastrukturu, která je tvořena pomocí přístupových bodů do WiFi sítí. Díky velkému množství těchto přístupových bodů není v dnešní době problém zjistit polohu zařízení. Mobilní zařízení musí být schopné přístupový bod vyhledat, ale nemusí být schopné se k němu připojit. Pomocí analýzy síly signálu jednotlivých přístupových bodů a znalosti kde se tyto body nacházejí [43] (uloženo v databázi např. od Google [44]), je možné vypočítat kde se zařízení aktuálně nachází. Výhodou lokalizace pomocí WiFi je možnost určení polohy ve vnitřních prostorech, přičemž mezi nevýhody patří, že nemá pokrytí v oblasti, kde se nevyskytují přístupové body a je závislá na databázích třetích stran s přístupovými body [43].

2.2. Identifikační bezdrátové technologie

2.2.1 RFID

RFID neboli Radio-Frequency Identification je technologie, která slouží k bezkontaktní komunikaci na krátkou vzdálenost. K této komunikaci využívá elektromagnetických vln. Tato technologie se skládá ze tří prvků: RFID tag, RFID snímač a RFID anténa. Tag obsahuje kromě antény také integrované obvody a díky tomu z něj snímač dokáže načíst informace. Snímač dále tyto informace z tagu pošle do systému (počítače), kde mohou být uloženy. Tag může být aktivní nebo pasivní.

Typy tagů:

- Pasivní – Využívá elektromagnetických pulsů, které RFID čtečka vyšle a pokud se v okolí nachází tag, využije přijatou energii a čtečce odešle odpověď.
- Aktivní – Aktivní tagy obsahují baterii a jsou neustále napájeny. Tyto tagy periodicky vysílají pulsy, které jsou zachytávány RFID čtečkami a dále zpracovávány.

Typické využití RFID technologie s pasivním tagem je identifikace zboží nebo přístup osob do budov [45].

2.2.2 NFC

Near Field Communication neboli NFC je technologie, která umožňuje rychlou a zabezpečenou výměnu dat na vzdálenost menší než 4 cm. V dnešní době se technologie nachází v řadě chytrých mobilních zařízeních a v praxi ji můžeme použít pro placení mobilním zařízením nebo pro rychlé spárování dvou mobilních zařízení [46].

NFC dokáže pracovat ve třech režimech:

- Čtení/zápis – V tomto režimu se zařízení s NFC (mobilní zařízení) chová jako čtečka NFC tagů, jakmile značku detekuje, může z ní číst data nebo do ní zapisovat data [47].
- Peer-to-peer – Režim, ve kterém si dvě zařízení s NFC dokážou vyměňovat informace mezi sebou. Tento režim využívá technologie Android Beam, která NFC použije pro párování telefonů a následný přenos dat probíhá přes Bluetooth [48].
- Emulace karet – Běžné využití je bezkontaktní platba. NFC umí emulovat kartu a ve spolupráci s Android Pay nebo Apple Pay dokáže bezkontaktně zaplatit pomocí mobilního zařízení [47].

2.2.3 Čárové kódy

Jednodimenzionální čárový kód slouží k rychlé identifikaci produktu. Tvoří ho série černých čar a bílých mezer definované šířky, které je možné načíst pomocí optických skenerů. Tyto kódy se dají rozdělit do čtyř skupin:

- Číselné
 - Obsahují pouze čísla od 0 do 9.
 - Tuto skupinu reprezentují čárové kódy EAN13, EAN8 nebo UPC-A.
- Alfnumerické
 - Obsahují čísla od 0 do 9 a písmena od a do Z.
 - Tuto skupinu reprezentují čárové kódy Code 39 a Code 93.
- ASCII
 - Obsahují všechny ASCII znaky.
 - Tuto skupinu reprezentují čárové kódy Code 128.
- GS1 AI kódovatelná znaková sada 82
 - Obsahuje hodnoty jako alfanumerické a speciální znaky.
 - Tuto skupinu reprezentují čárové kódy GS1 Databar.

Mezi nejpoužívanější v Evropě patří EAN13 nebo EAN8, které se používají pro identifikaci zboží v obchodech. Další použití může být například v lyžařských střediscích nebo v poštovníctví [49].



Obrázek 5: Čárový kód EAN13. Zdroj [50].

2.2.4 QR kód

QR kód je dvojdimenzionální typ čárového kódu, který lze snadno přečíst pomocí mobilního zařízení. Po načtení kódu se informací dočkáme téměř okamžitě, kvůli tomu zkratka QR (Quick Response). Tento kód se uchovává ve čtvercové podobě a lze ho rozdělit do sedmi částí:

- **Poziční značky** – Tři velké čtverce, které slouží k rychlému načtení a správné orientaci kódu.
- **Zarovnávací značky** – Slouží k narovnání QR kódu na zakřiveném povrchu.
- **Časový vzor** – Pomocí těchto řádků skener určí, jak je velká oblast s daty.
- **Informace o verzi** – V této části se definuje, jaká verze QR kódu se používá.
- **Informace o formátování** – Tato část obsahuje informace o toleranci chyb.
- **Data a klíč k opravě chyb** – Data a bloky k opravě chyb sdílí prostor, tyto bloky umožňují poškození až 30 % kódu [51].
- **Tichá zóna** – Tato část je tvořena bílým okolím samotného kódu a zajišťuje, že čtečka nenačte informace, které se netýkají snímaného kódu [52].

Do QR kódu se dá uložit v podstatě cokoliv. Ať už se jedná o obyčejný text, webový odkaz nebo přihlašovací údaje k WiFi síti. Mezi další známé dvojdimenzionální kódy patří Datamatrix nebo Aztec code [53].



Obrázek 6: QR kód. Zdroj: vlastní.

2.2.5 Biometrie

Identifikace pomocí biometrie je založená na rozpoznání jedinečných biometrických charakteristik člověka [54]. Jedna z metod identifikace pomocí biometrie je rozpoznání pomocí obličeje. Tato metoda nevyžaduje žádný kontakt s osobou a je využívána například v nejnovějších iPhonech, kde se pomocí obličeje může uživatel přihlásit, nakupovat nebo potvrzovat platbu [55]. Další velice rozšířenou metodou identifikace pomocí biometrie je rozpoznání otisku prstů. K načtení otisku se používají optické, kapacitní nebo ultrazvukové skenery [56]. Tato metoda se využívá v mobilních zařízeních, zajištění budov nebo k otevření vstupní brány. Mezi další metody patří identifikace duhovky, tvaru ruky nebo žilního vzoru [54].

3. NÁVRH APLIKACE

Aplikace vytvořená v rámci této práce bude využívat AR pro zobrazení 3D modelu a bude tvořena ze tří částí. V první části bude uživatel odpovídat na vygenerovanou otázku na základě zobrazeného modelu. Ve druhé části budou všechny modely, které je možné v aplikaci zobrazit, nabídnuty uživateli, který z nich jeden vybere a ten bude následně zobrazen i s informacemi o něm. V poslední části budou uživateli zobrazeny informace o vývojáři aplikace a také QR kód, který v aplikaci slouží jako referenční obrázek.

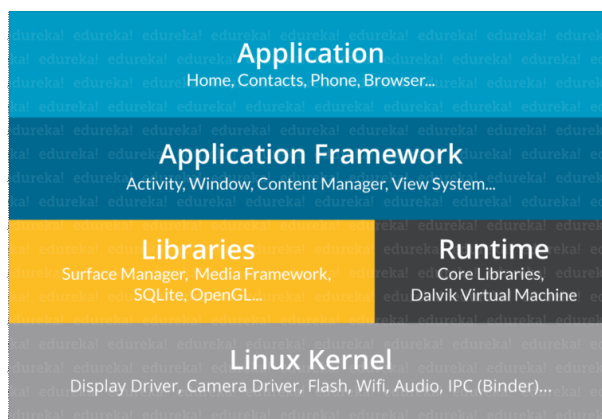
3.1. 3D modely

3D model je matematická reprezentace objektu, který můžeme zobrazit ve třech dimenzích. Tyto modely jsou využívány v mnoha odvětvích jako je virtuální realita, počítačové hry, 3D tisk nebo ve filmovém odvětví. V API ARcore, které využívá vytvořená aplikace v této práci, mohou být použity 3D modely formátu glTF, obj a fbx. Typickým základem těchto modelů je strukturální stavba, která se skládá z polygonů. 3D model můžeme také naskenovat pomocí zařízení podporující technologii LiDAR. Modely se dají animovat, což vyžaduje větší pečlivost při tvorbě, z důvodu možných problémů při neobvyklém pohybu. Barva nebo vyzařování světla bývá definováno pomocí 2D obrázků zvaných mapy. Tento způsob se používá především v počítačových hrách. Další možností je matematická definice barev a dalších charakteristik povrchu pomocí shaderů [57]. Mezi nejznámější programy k tvorbě 3D modelů patří Maya, Houdini nebo Blender [58]

3.2. Operační systém Android

Android je open source, Linuxově založený operační systém, který se dodává především v mobilních zařízeních a tabletech [59]. Tento operační systém je celosvětově nejrozšířenější a na globálním trhu mobilních operačních systémů zastupuje přibližně 70 % [60].

Architektura tohoto operačního systému je rozdělena do pěti základních částí.



Obrázek 7: Android architektura. Zdroj [61]

3.2.1 Aplikační vrstva

Na tuto vrstvu jsou instalovány všechny nativní aplikace a všechny aplikace třetích stran. Tato vrstva běží v době běhu systému pomocí tříd a služeb, které jsou poskytovány vrstvou Aplikační framework [62].

3.2.2 Aplikační framework

Tato vrstva poskytuje třídy a služby které můžeme využít pro vývoj Android aplikací a služby vysoké úrovně. Zahrnuje služby jako jsou lokalizační služby, služba NFC nebo správce oznámení [62].

3.2.3 Android Runtime

Tato vrstva je jedna z nejdůležitějších v Android architektuře. Hlavní účel této vrstvy je zajištění chodu aplikací pomocí základních knihoven a poskytuje základ pro Aplikační framework.

Tato vrstva obsahuje dvě důležité části:

- Základní knihovny – Tyto knihovny umožňují implementovat aplikace, pomocí programovacího jazyka Java [62].
- Virtuální stroj Dalvik – Tento virtuální stroj je využíván pro optimalizaci životnosti baterie, paměti a výkonu. Zajišťuje také konverzi z byte kódu, generovaného Java kompilátorem, do souboru s příponou dex. S tím jsou konvertovány třídy do jednoho .dex souboru pomocí kompilátoru Dex, který běží na virtuálním stroji Dalvik [63].

3.2.4 Knihovny

Vrstva obsahuje základní knihovny jazyka C/C++ a Java knihovny jako jsou SSL, SQLite, Webkit a další, které poskytují podporu pro vývoj operačního systému Android [63].

3.2.5 Linux Kernel

Vrstva Linux Kernel je základní část architektury operačního systému Android. Tato vrstva spravuje všechny ovladače (např. zvukové ovladače, ovladače kamer, ovladače paměti), které jsou hlavně vyžadovány při běhu zařízení. Také poskytuje abstrakční vrstvu mezi hardwarem zařízení a ostatními částmi této architektury. Tato vrstva zajišťuje správu paměti a baterie [63].

3.3. Analýza

3.3.1. Funkční požadavky

Funkční požadavky definují, co by měl vyvíjený systém dělat a určuje je uživatel. Na rozdíl od nefunkčních požadavků, lze tyto požadavky vidět přímo v konečné verzi vyvíjeného systému a pomáhají nám k ověření funkcionality systému [64].

Tabulka 1: Funkční požadavky. Zdroj: vlastní.

ID	Požadavek
F1	System bude identifikovat plochy pro zobrazení modelu
F2	System bude zobrazovat 3D modely
F3	System bude zaznamenávat aktuální skóre
F4	System bude nabízet knihovnu 3D modelů
F5	System bude v knihovně zobrazovat informace o modelu
F6	System bude nabízet otázky a jejich zodpovězení

3.3.2. Nefunkční požadavky

Tyto požadavky neurčují konkrétní chování aplikace, ale definují kvalitu atributu systému (rychlost, zabezpečení) a nijak neovlivňují funkcionality systému. Tyto požadavky jsou

definovány vývojáři nebo technickým vedoucím vyvíjeného systému a pomáhají nám ověřit výkon systému [64].

Tabulka 2: Nefunkční požadavky. Zdroj: vlastní.

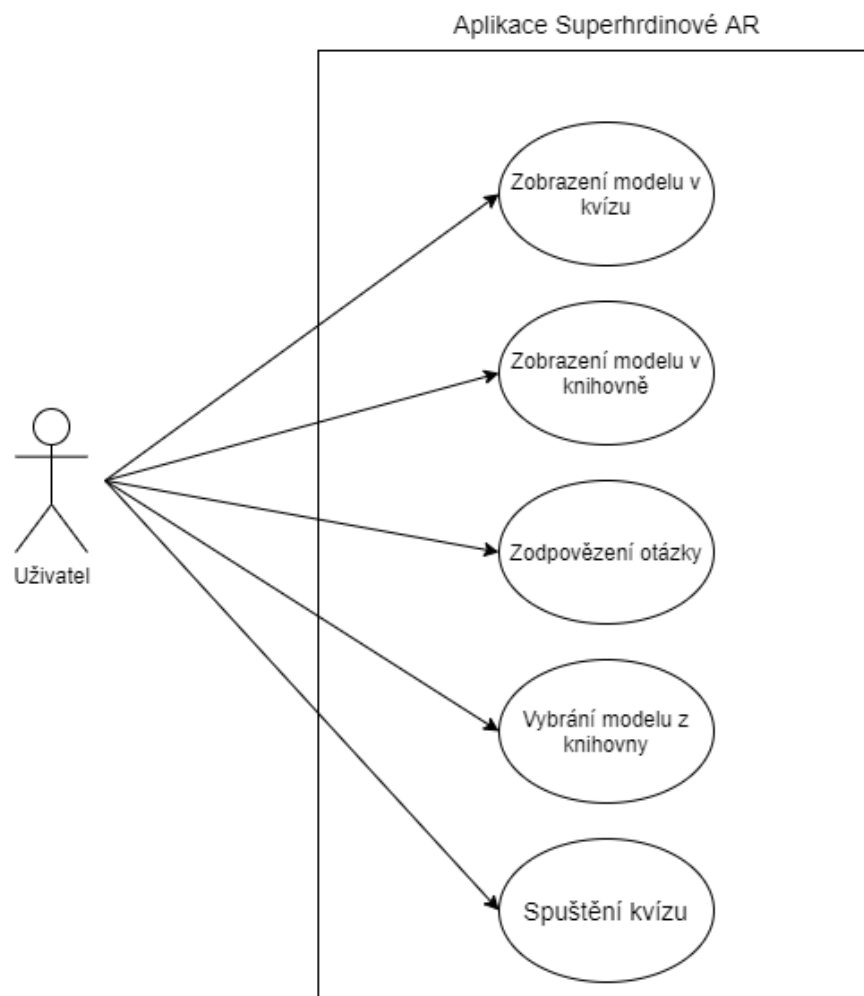
ID	Požadavek
N1	Systém bude naprogramován v programovacím jazyce Java.
N2	Systém bude fungovat na zařízeních s OS Android 7.0 a vyšších.
N3	Systém bude využívat API ARCore pro zobrazení modelů.
N4	Systém bude používat formát modelů glTF.

3.3.3. Use case diagram

Diagram užití neboli Use case diagram je znázornění chování systému z pohledu uživatele. Jeho účelem je popsat funkcionalitu systému. Diagram zobrazuje, co bude daný systém umět, co bude jednotlivým aktérům umožněno dělat, ale neříká, jak to bude dělat.

Diagram se skládá se tří základních částí:

- Případ užití (Use case) – Znázorňuje akci v systému, která vede k dosažení určitého cíle. Je znázorněn oválem s textem uvnitř.
- Aktéři – Aktér je role v systému, která je přiřazena k případům užití pomocí vztahů mezi nimi. Tato role může být například uživatel, server nebo administrátor.
- Vztahy mezi aktéry a případy užití – vztahy mohou být znázorněny šipkami nebo čarami.
 - Relace mezi aktérem a use casem – znázorňuje, který případ užití může aktér vykonat.
 - Relace <<include>> - používá se, pokud je jeden use case využíván v rámci jiného use casu.
 - Relace <<extend>> - znázorňuje dědičnost mezi aktéry nebo use casy [65].



Obrázek 8: Use Case diagram. Zdroj: vlastní.

V zobrazeném use case diagramu lze vidět, že uživatel používající aplikaci může zobrazovat modely, odpovídat na otázky a další.

3.3.4. Use case specifikace

V Use case diagramu je znázorněno, k jakým funkcionalitám mají aktéři přístup, ale již nevíme, jak budou danou funkci provádět. To je doplněno pomocí use case specifikací, které nám dávají více informací o jednotlivých use casech. Tyto specifikace nemají pevnou strukturu, nicméně je můžeme rozdělit do několika částí:

- název případu užití,
- krátký popis,
- aktéři,
- podmínky pro spuštění,
- základní tok,
- alternativní tok,
- podmínky pro ukončení,

příčemž všechny části nejsou povinné. Tabulka 3 ukazuje, jak vypadá Use case specifikace pro případ užití Zobrazení modelu v kvízu [66].

Tabulka 3: Use case specifikace – Zobrazení modelu v kvízu. Zdroj: vlastní.

Případ užití	UC1 – Zobrazení modelu v kvízu
Krátký popis	Use case umožňuje zobrazit model v reálném světě
Aktéři	Uživatel, systém
Podmínky pro spuštění	<ol style="list-style-type: none">1. Uživatel se musí nacházet v části aplikace pro zobrazení objektu2. Musí být dostatečné světelné podmínky
Základní tok	<ol style="list-style-type: none">1. Uživatel se nachází v části aplikace pro zobrazení modelů2. Systém načte komponenty, vygeneruje a zobrazí otázku3. Uživatel hýbe mobilním zařízením okolo markeru, dokud se mu nenačtou plochy4. Systém zaznamená marker5. Uživateli se zobrazí model
Podmínky pro ukončení	Zobrazení modelu na markeru

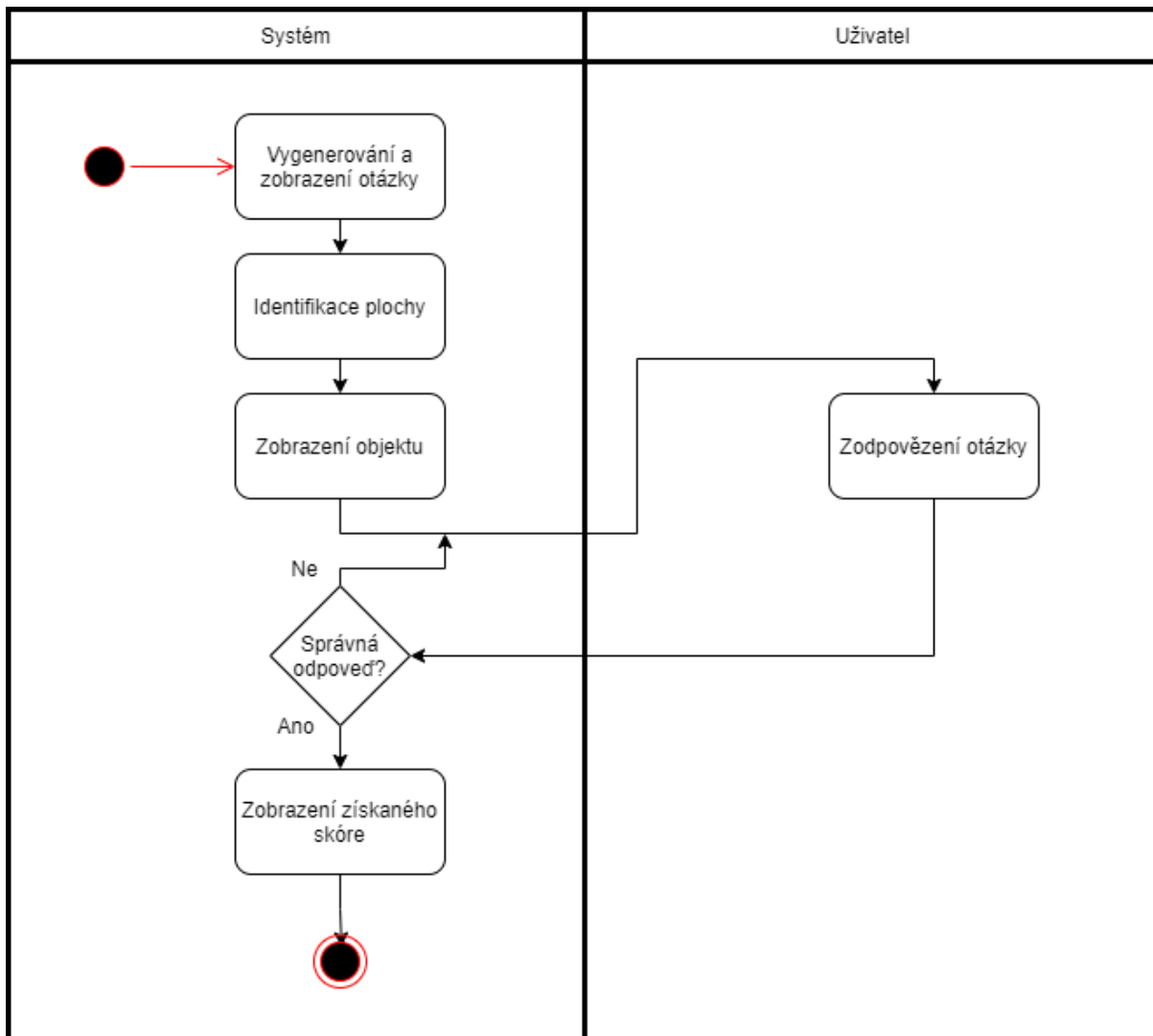
3.3.5. Diagram aktivit

Jedná se o diagram, který znázorňuje posloupnost aktivit, které se budou vykonávat. Pomocí tohoto diagramu můžeme vidět, jak by se měl systém chovat v různých situacích. V diagramu může existovat několik typů uzlů:

- Akční uzly – Znázorňující v rámci aktivity nedělitelné a samostatné jednotky.
- Řídící uzly – Řídí cesty uvnitř aktivity.
 - Uzly rozvětvení a spojení – umožňují provádění procesů paralelně.
- Objektové uzly – V rámci diagramu zastupují objekty.
- Rozhodovací uzly – Uzel má jeden vstup a více výstupů, pomocí kterých se dá diagram větvit.
- Slučovací uzly – Uzel umožňuje sloučit více vstupů do jediného výstupu.

Diagram se také pro větší přehlednost může rozdělit do oddělených bloků tzv. plaveckých drah.

Na obrázku 9 je diagram aktivit, který znázorňuje průběh zodpovězení otázky [67].



Obrázek 9: Diagram aktivit. Zdroj: vlastní.

3.4. Uživatelské rozhraní aplikace

3.4.1. Použité layouty

Struktura uživatelského rozhraní je v aplikacích definována pomocí layoutů. Tyto layouty jsou tvořeny pomocí hierarchie, kterou tvoří objekty View a skupiny objektů ViewGroup. View v aplikaci obvykle představuje něco, co může provádět interakci s uživatelem a ViewGroup definuje strukturu rozložení View objektů, případně dalších ViewGroup. Mezi objekty View patří například Button, TextView nebo ImageView a mezi nejpoužívanější ViewGroup objekty můžeme zařadit LinearLayout, ConstraintLayout nebo RelativeLayout. Samotný layout můžeme deklarovat pomocí jazyka XML, s tím Android Studio nabízí také Layout Editor, kdy

můžeme vytvořit XML dokument pomocí drag-and-drop metody. Druhým způsobem, jak deklarovat layout je pomocí kódu za běhu programu. Zde můžeme vytvářet View a ViewGroup objekty nebo upravovat jejich vlastnosti [68].

3.4.1.1 ConstraintLayout

Tento layout poskytuje velmi flexibilní a přizpůsobivé vytváření více View pro aplikace. Zároveň je nyní výchozím layoutem Android Studia a nabízí mnoho způsobů, jak umístit objekty View nebo ViewGroup [69]. V layoutu můžeme přichytit jeden View k ostatním objektům a tím zajistit správné rozložení v aplikaci. Zároveň je nutné přichytit minimálně jednu horizontální a jednu vertikální stranu [70].

3.4.1.2 FrameLayout

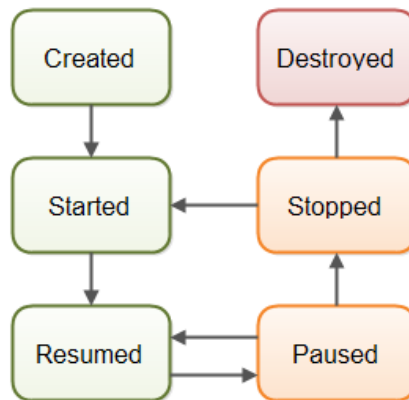
FrameLayout je navržen k blokování oblasti na obrazovce a zobrazení pouze jednoho View. Primárně by měl být použit pro zobrazení jednoho objektu, kvůli obtížnosti umístění více objektů [71]. Více vložených View se skládají jeden na druhého, což může být užitečné pro překrývané uživatelské rozhraní. Nevýhodou tohoto layoutu je, že dokáže umístit View objekty pouze pomocí gravitace, vzhledem k samotného layoutu [72].

3.4.2. Použité komponenty

V této části budou krátce popsány komponenty použité ve vyvíjené aplikaci z vývojového prostředí Android Studio. V aplikaci byly využity základní komponenty jako Button nebo ImageView ale také méně obvyklé jako je ArFragment, který je potřeba pro práci s ARCore. Ještě před představením samotných komponent je nutné definovat, co je to aktivita.

3.4.2.1 Aktivita

Aktivita je v Android aplikaci jedna stránka uživatelského rozhraní. V aplikaci může být jedna nebo více aktivit a při zapnutí aplikace se zobrazí hlavní aktivita, z které se může uživatel přesouvat do jiných aktivit například pomocí Buttonů. Každá aktivita má svůj životní cyklus a sadu metod, které odpovídají stavům životního cyklu [73].



Obrázek 10: Životní cyklus aktivity. Zdroj [73].

3.4.2.2 ArFragment

Fragment je součástí aktivity a můžeme o něm říct, že je to podaktivita. V jedné aktivitě může být více těchto fragmentů a na každý z nich zobrazovat něco jiného [74]. ArFragment zajišťuje kontrolu instalace ARCore nebo povolení přístupu ke kameře [75].

3.4.2.3 TextView

TextView se používá k nastavení a zobrazení textu. Text můžeme nastavit již před startem aplikace nebo až za běhu programu pomocí metody. Nastavený text uživatelé nemohou upravovat, jediná možnost úpravy je ve zdrojovém kódu pomocí metody *setText* [76].

3.4.2.4 ProgressBar

ProgressBar slouží k zobrazení postupu určité operace. Může pracovat v jednom ze dvou režimů:

- Neurčitý pokrok – Tento režim se použije v případě, že se předem neví, jak dlouho operace bude trvat. Režim je nastaven automaticky při vložení komponenty a zobrazuje cyklickou animaci bez jakéhokoli zobrazení pokroku.
- Určitý pokrok – druhou možností je režim, při kterém už předem víme, jak dlouho bude operace trvat a chceme zobrazit pokrok dané operace. Tento režim se hodí například pro zobrazení zbývajících procenta přehrávaného zvukového souboru. Nastavení aktuální hodnoty, kterou chceme zobrazit, nastavíme pomocí atributu *progress* [77].

3.4.2.5 ListView

Díky ListView můžeme zobrazit prvky ve svislém rolovacím seznamu. Tyto prvky se do ListView vloží pomocí objektu Adapter, který dokáže načíst prvky z pole nebo databáze. Komponenta, stejně jako Button, dokáže vyvolat událost po stisknutí prvku v seznamu [78].

3.4.2.6 Button

Komponenta Button neboli tlačítko se může skládat z textu, ikony nebo obojího dohromady. Tato komponenta dokáže zaznamenat, pokud na ni někdo klikl pomocí *onClick* události. Abychom na tuto událost mohli reagovat, je potřeba ještě definovat obslužnou metodu. Díky této komponentě můžeme například otevírat nové aktivity, zavřít aplikaci nebo vyvolat jakoukoliv jinou akci [79].

3.4.2.7 ImageView

ImageView se používá pro zobrazení obrázků v Android aplikacích. Obrázky lze přiřazovat staticky před spuštěním aplikace nebo dynamicky za běhu aplikace [80]. Při dynamickém přiřazení může být obrázek typu Drawable nebo Bitmap. Také může dojít k přiřazení pomocí Resource id nebo URI [81].

3.4.3. Celkový návrh vzhledu

Návrh vzhledu aplikace lze rozdělit do čtyř částí:

- Návrh úvodní strany,
- návrh kvízu,
- návrh knihovny,
- návrh strany o aplikaci.

Při návrhu úvodní strany jde hlavně o to, aby se uživatel dostal do části s kvízy, do knihovny nebo do části o aplikaci. Na základě toho, jsou zde tři Buttony, které ho přesměrují na potřebnou stranu. Také zde bude název aplikace ve formě ImageView.



Obrázek 11: Návrh úvodní strany. Zdroj: vlastní.

Dále je v aplikaci strana, která propojuje úvodní stranu se stranou kvízu, na které se budou nacházet:

- 2 Buttony,
- 1 ImageView,
- 1 ProgressBar.

Na nejdůležitější straně v části s kvízem se bude nacházet:

- 5 Buttonů,
- 1 ListView,
- 3 ImageView,
- 1 TextView,
- 1 arFragment,
- 3 layouty, z nichž 2 budou ConstraintLayout a 1 FrameLayout.

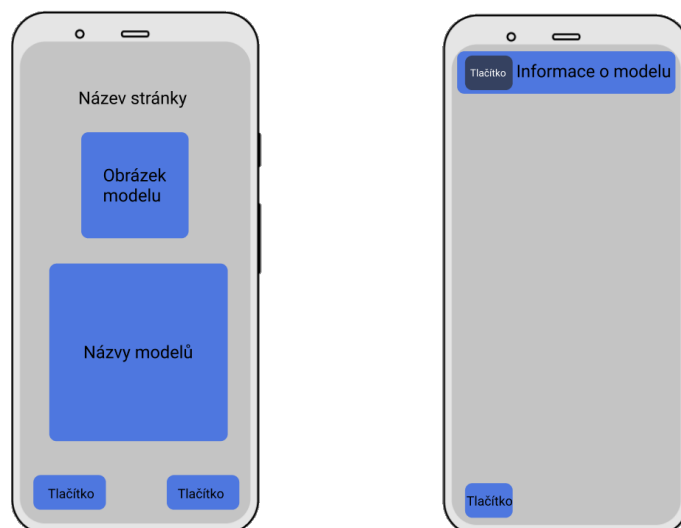


Obrázek 12: Návrh stran v kvízu. Zdroj: vlastní.

Další částí návrhu vzhledu je návrh knihovny. V této části máme opět dvě strany, kde první bude obsahovat:

- 2 Buttony,
- 1 ListView,
- 1 TextView,
- 1 ImageView.

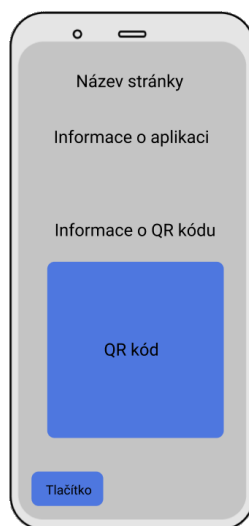
Na druhé straně budou stejně jako v části s kvízem celkem 3 layouty, z nichž 1 bude FrameLayout a 2 budou ConstraintLayout. To doplní 2 Buttony, 1 TextView a 1 arFragment.



Obrázek 13: Návrh stran v knihovně. Zdroj: vlastní.

Na straně o aplikaci se budou nacházet užitečné informace a budou zde použity následující komponenty:

- 3 TextView,
- 1 Button,
- 1 ImageView.



Obrázek 14: Návrh strany o aplikaci. Zdroj: vlastní.

4. VÝVOJ APLIKACE

4.1. Použité technologie

4.1.1. Android studio

Android Studio je vývojové prostředí, které vyvinula společnost Google za pomoci společnosti JetBrains [82]. Toto vývojové prostředí podporuje programovací jazyky Java, Kotlin nebo C++, kdy od roku 2019 je pro programování Android aplikací preferován jazyk Kotlin [83]. Android Studio je možné použít na operačních systémech Windows, macOS, Linux a Chrome OS [84]. Také má vlastní vestavěný emulátor Android zařízení, díky kterému můžeme testovat aplikace bez použití fyzického zařízení s operačním systémem Android. Tento emulátor dokáže simulovat téměř vše, co dokáže fyzické zařízení. Můžeme zde simulovat příchozí hovory, SMS zprávy, simulovat lokaci daného zařízení, simulovat rotaci zařízení a další hardwarové senzory. Testování aplikací pomocí emulátoru je rychlejší než použití fyzického zařízení díky tomu, že se data nemusí přenášet přes USB kabel. V emulátoru jsou k dispozici mobilní zařízení, tablety, hodinky nebo televize, podporující operační systém Android [85]. Vývojové prostředí také nabízí zobrazení 3D modelů.

4.1.2. Java

Java je programovací jazyk, který byl vytvořen v roce 1995 a je vlastněn společností Oracle. Je to objektově orientovaný jazyk, který je bezpečný, výkonný a rychlý. Tento programovací jazyk je jeden z nejznámějších na celém světě, běží na více než 3 miliardách zařízení po celém světě a je používán pro mobilní aplikace, desktopové aplikace, webové aplikace, webové servery, hry, databázové spojení a mnoho dalšího. Java dokáže pracovat na platformách Windows, Mac, Linux nebo Raspberry Pi. Zároveň je velmi blízký jazykům C++ a C#, což usnadňuje přesun z jednoho jazyka na druhý [86].

4.1.3. ARCore

V aplikaci je použita technologie ARCore, která umožňuje zobrazit 3D modely a je blíže popsána v kapitole 1.4.1 ARCore.

4.2. Struktura aplikace

4.2.1. Balíčky

V aplikaci můžeme najít velké množství balíčků. Zde budou uvedeny pouze ty nejdůležitější vzhledem k aplikaci. Vůbec nejdůležitějším balíčkem v aplikaci je balíček s názvem *com.upce.ar*, ve které se nacházejí všechny vytvořené třídy, které jsou potřebné pro správný chod aplikace. Dalším velmi důležitým balíčkem pro naši aplikaci je balíček *raw*, který se nachází ve složce *res* a ve kterém jsou uloženy všechny 3D modely, které se v aplikaci používají. V té samé složce je také balíček *layout*, kde se nachází všechny navržené layouty a také složka *Drawable*, kde můžeme najít vlastní navržené xml soubory, které definují například zaoblení rohů.

4.2.2. Třídy

V aplikaci se celkem nachází 11 tříd, díky kterým je možné zobrazovat objekty a generovat nebo odpovídat na otázky.

- Třída *QuizAr* a *LibraryAR* – Obě tyto třídy zajišťují nasnímání plochy pomocí kamery a následné zobrazení 3D modelu na identifikovaném markeru. K tomu mají implementovány potřebné metody. Třída *QuizAR* má navíc metody pro zobrazení získaného počtu bodů, generování nové otázky, nastavení aktuální otázky nebo pro získání maximálního měřítka aktuálního 3D modelu. Třída *LibraryAR* nabízí kromě metod pro zobrazení 3D modelů také metodu načtení informací o jednotlivých modelech.
- Třídy *Score*, *Model*, *Questions* a *Question*, která je podtřídou třídy *Questions*, pouze uchovávají data, ke kterým můžeme přistupovat pomocí metod typu *get* a nastavovat pomocí metod typu *set*.
- Pomocí třídy *ModelSelection*, můžeme přistoupit ke knihovně 3D modelů a vybrat, který model se nám zobrazí v dalším kroku.
- Třídy *HomeScreen*, *AboutApp* a *QuizWelcome* slouží pouze k možnosti přesunu do jiné aktivity.
- Dále je zde třída *EnumQuestionType*, která definuje možné kategorie, do kterých mohou být jednotlivé otázky přiřazené.

- Poslední je třída *CustomArFragment*, která umožňuje vytvořit databázi obrázků, které má ARCore identifikovat.

4.3. Rozbor kódu

4.3.1 SceneForm/ARCore

V této části bude ukázáno, co všechno je potřeba zajistit předtím, než budeme moct začít používat ARCore, pro zobrazení 3D modelů.

Jelikož samotný ARCore je engine, který pomáhá SDK renderovat objekty, Google vyvinul Sceneform SDK, které dovoluje vývojářům vytvářet Android aplikace využívající AR bez nutnosti využití OpenGL. Sceneform přináší řadu užitečných funkcionalit, například kontrolu oprávnění kamery nebo plugin pro manipulaci s 3D objekty. V Android Studiu se plugin Sceneform vyhledá mezi ostatními pluginy pod názvem *Google Sceneform Tool (Beta)* a následně se přidá. S tím je potřeba přidat do souboru Gradle závislost pro Sceneform UX [75].

```
dependencies {
    implementation "com.google.ar.sceneform.ux:sceneform-ux:1.12.0"
    //Další závislosti
}
```

Sceneform SDK dále vyžaduje Android API level 24 nebo vyšší, což se nastaví opět v souboru Gradle.

```
android {
    defaultConfig {
        applicationId "com.upce.ar"
        minSdkVersion 24
        //Další konfigurace
    }
}
```

Pokud je tato verze menší než 26, musí se rovněž přidat další kompilační nastavení. Toto nastavení je nutné, protože Sceneform knihovny využívají jazykové struktury Javy 8.

```
compileOptions {
    sourceCompatibility JavaVersion.VERSION_1_8
    targetCompatibility JavaVersion.VERSION_1_8
}
```


V tom samém souboru se musí také nastavit přístup k Maven repositáři od Googlu. Přidáním Google repositáře se zajistí i přístup do požadovaného Maven repositáře [75].

```
repositories {  
    google()  
    //ostatní repositáře  
}
```

4.3.2 Manifest

Pro svou funkci vyžaduje ARCore kameru. K povolení přístupu ke kameře se využívá soubor `AndroidManifest.xml`, ve kterém také můžeme nastavit, jestli aplikace vyžaduje ARCore. Na základě toho, se aplikace bude v Google Play zobrazovat jen takovým zařízením, která podporují ARCore. V manifestu se dále nachází metadata, díky kterým se při stažení aplikace stáhne a nainstaluje potřebné služby pro AR. Také se v tomto souboru dají nastavit věci jako ikona aplikace nebo orientace obrazovky [75].

```
//Povolení přístupu ke kameře  
<uses-permission android:name="android.permission.CAMERA" />  
//Požadavek kamery v mobilních zařízeních  
<uses-feature  
    android:name="android.hardware.camera.ar"  
    android:required="true" />  
//Požadavek ARCore v mobilním zařízení  
<meta-data  
    android:name="com.google.ar.core"  
    android:value="required" />
```

4.3.3 Zobrazení modelu

Pro zobrazení 3D modelů, je nejdříve potřeba vytvořit databázi obrázků, které má mobilní zařízení v reálném světě rozpoznat, a na které bude následně zobrazen daný model. K tomu je zde využita třída `AugmentedImageDatabase`, do které můžeme přidat referenční obrázky. V metodě `buildDatabase` můžeme vidět, že třída je inicializována pro aktuální `Session`, je načten referenční obrázek na základě jména aktuálního modelu, je do ní přidán referenční obrázek typu `Bitmap` a předána instancí třídy `Config`, jako databáze referenčních obrázků. Následně je vrácena hodnota `true`, která značí úspěšné provedení této metody.

```

public boolean buildDatabase(Config config, Session sessionIn) {
    //Vytvoření databáze obrázků
    AugmentedImageDatabase augmentedImageDatabase = new
        AugmentedImageDatabase(sessionIn);

    //Načtení obrázku
    Bitmap bitmap = loadImage(model.getQrImage());
    if (bitmap == null) {
        return false;
    }
    //Přidání obrázku do databáze
    augmentedImageDatabase.addImage(model.getModelName(), bitmap);
    //Přidání databáze do konfigurace
    config.setAugmentedImageDatabase(augmentedImageDatabase);
    return true;
}
//Metoda pro načtení obrázku na základě názvu
private Bitmap loadImage(String filename) {
    try {
        InputStream is =
            getApplicationContext().getAssets().open(filename);
        return BitmapFactory.decodeStream(is);
    } catch (IOException e) {
        e.printStackTrace();
    }
    return null;
}
}

```

Po vybudování databáze referenčních obrázků je nutné, aby aplikace dokázala tyto obrázky správně identifikovat. Do listeneru `ArFragmentu` se přidá metoda, která se bude provádět při každém načtení nového snímku. Tato metoda se jmenuje *onUpdate* a zajišťuje správnou identifikaci referenčních obrázků.

```

//Přidání listeneru k ArFragmentu
arCoreFragment.getArSceneView().getScene().addOnUpdateListener(this::
onUpdate);

```

V metodě *onUpdate* se nejdříve načte aktuální snímek kamery, na základě kterého se `ARCore` pokusí rozpoznat referenční obrázky. Následně načteme kolekci obrázků, které `ARCore` dokázal identifikovat jako referenční. Přes tyto obrázky budeme iterovat, ve snaze najít obrázek, který odpovídá obrázku přiřazenému k modelu. Pokud tato situace nastane, program se přesune do metody *placeObject*.

```

public void onUpdate(FrameTime frameTime) {
    //Načtení aktuálního snímku
    Frame frame = arCoreFragment.getArSceneView().getArFrame();
    //Načtení identifikovaných referenčních obrázků
    Collection<AugmentedImage> updateAugmentedImg =
        frame.getUpdatedTrackables(AugmentedImage.class);
    //Iterování přes identifikované obrázky
    for (AugmentedImage image : updateAugmentedImg) {
        if (image.getTrackingState() == TrackingState.TRACKING) {
            //Zjištění, zda identifikovaný obrázek je stejný jako
            //obrázek přiřazený k modelu
            if (image.getName().equals(model.getModelName()) &&
                !model.getIsPlaced()) {
                model.setIsPlaced(true);
                placeObject(arCoreFragment,
                    image.createAnchor(image.getCenterPose()),
                    model.getModel());
            }
        }
    }
}

```

Metoda *placeObject* slouží k vytvoření *Renderable* modelu, na základě číselného identifikátoru modelu. Pokud vytvoření proběhlo úspěšně, tento model se předá metodě *addNodeToScene* jako parametr. V případě neúspěchu se vypíše výjimka.

```

private void placeObject(ArFragment arFragment, Anchor anchor, int
                        object)
{
    ModelRenderable
        .builder()
        .setSource(Objects.requireNonNull(arFragment.getContext()),
                object)
        .build()
        //Volání metody addNodeToScene v případě úspěchu
        .thenAccept(modelRenderable ->
            addNodeToScene(arFragment, anchor, modelRenderable))
        //Výpis výjimky v případě neúspěchu
        .exceptionally(throwable -> {
            Toast.makeText(arFragment.getContext(), "Error:"
                + throwable.getMessage(), Toast.LENGTH_LONG).show();
            return null;
        });
}

```

AddNodeToScene je metoda, která zajišťuje přidání modelu do scény. V této metodě se vytvoří *AnchorNode* a *TransformableNode*, kde první zmíněný se nastaví jako rodič toho druhého. Také se zde nastaví měřítko modelu, ve kterém bude zobrazen a dojde k přidání *AnchorNodu* do scény [87].

```

private void addNodeToScene(ArFragment arFragment, Anchor anchor,
                           ModelRendererable modelRenderable)
{
    AnchorNode anchorNode = new AnchorNode(anchor);
    TransformableNode node = new
        TransformableNode(arFragment.getTransformationSystem());
    node.setRenderable(modelRenderable);
    node.setParent(anchorNode);
    node.getScaleController().setMinScale(0.01f);
    node.getScaleController().setMaxScale(model.getMaxScale());
    //Přidání nodu do scény
    arFragment.getArSceneView().getScene().addChild(anchorNode);
    node.select();
}

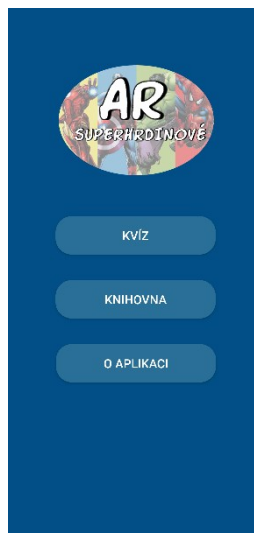
```

4.4. Uživatelská příručka

Tato kapitola slouží pro usnadnění orientace v aplikaci, ukazuje, jak aplikace vypadá a její funkčnost.

4.4.1 Hlavní stránka

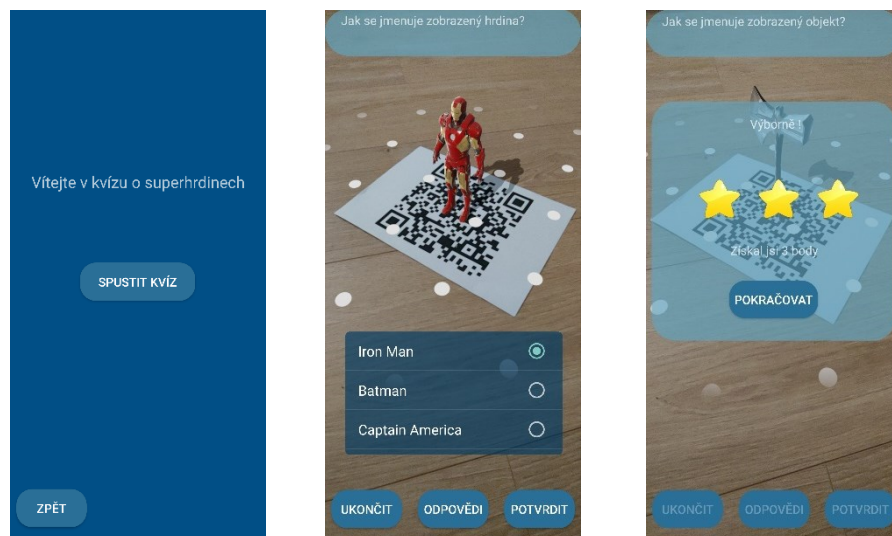
Tato stránka se uživateli zobrazí hned po otevření aplikace a slouží v ní jako rozcestník. Z této stránky může pokračovat pomocí tlačítek do části s kvízem, knihovnou nebo do části o aplikaci.



Obrázek 15: Úvodní obrazovka. Zdroj: vlastní.

4.4.2 Kvíz

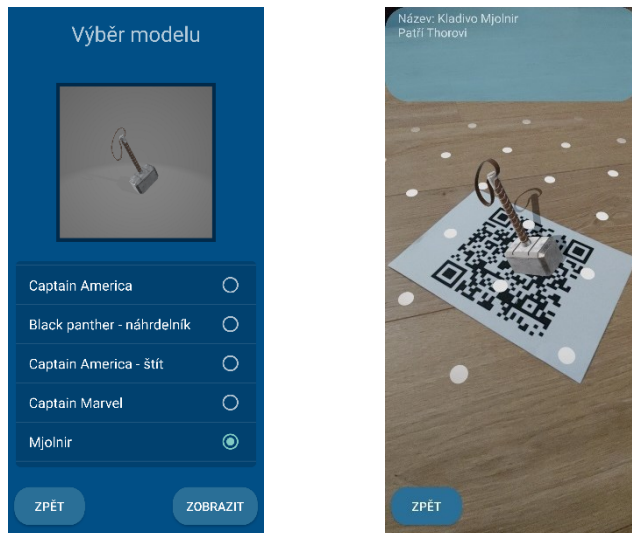
Do této části se uživatel dostane pomocí tlačítka z hlavní stránky a obsahuje dvě části. Do první části se uživatel dostane po stisknutí tlačítka z části Hlavní stránka. Tato část slouží pouze k upozornění, že se uživatel chystá spustit kvíz a po kliknutí na tlačítko Spustit kvíz se dostane do druhé části. Druhá část už slouží k samotnému řešení kvízu, kdy po načtení otázky musí najít správnou odpověď. Kvíz lze kdykoliv opustit nebo dojde k vrácení na hlavní stránku po skončení kvízu.



Obrázek 16: Obrazovky v části s kvízem. Zdroj: vlastní.

4.4.3 Knihovna

Do části s knihovnou se uživatel opět dostane z hlavní stránky a rovněž je rozdělena do dvou částí. V první části jsou uživateli nabídnuty modely, které může zobrazit. Po vybraní je uživatel přesunut do druhé části, která zobrazí vybraný model a vypíše o něm přiřazené informace. Pomocí tlačítek zpět se uživatel může vrátit na hlavní stránku, kde může provádět další akce.



Obrázek 17: Obrazovky v části s knihovnou. Zdroj: vlastní.

4.4.4 O aplikaci

Poslední část obsahuje pouze informace o vývojáři aplikace a QR kód, který slouží jako marker pro zobrazení modelů. Tento kód je zde zobrazen, aby nemusel být k aplikaci přidán žádný jiný soubor a uživatel mohl získat tento kód přímo v aplikaci.



Obrázek 18: Obrazovka v části O aplikaci. Zdroj: vlastní.

5. TESTOVÁNÍ

V této kapitole bude popsáno testování vývojářem z pohledu funkcionality a testování uživatelem z pohledu uživatelské přívětivosti.

5.1. Testování vývojářem

Při testování aplikace vývojářem bylo testováno, zda je aplikace stabilní, jak rychle zvládne identifikování referenčního obrázku a následné zobrazení objektu a zda aplikace funguje tak, jak bylo zamýšleno.

5.2. Testování uživatelem

Aplikace byla dána k otestování uživateli, který nemá blízký vztah k technologickému odvětví. Uživatel dostal mobilní zařízení a referenční obrázek, ale již nedostal žádné informace o tom, co má dělat nebo jak se má aplikace chovat. První a jediný problém nastal, když se uživatel dostal do části, ve které se má zobrazit objekt. V této části uživatel nevěděl, jak použít mobilní zařízení v kombinaci s referenčním obrázkem, avšak po upřesnění, k čemu referenční obrázek slouží nebyl s používáním aplikace žádný problém. Tento jediný problém dávám za vinu tomu, že uživatel nemá téměř žádný vztah k technologickém odvětví a považuji aplikaci za přehlednou a uživatelsky přívětivou i pro tento typ uživatelů.

5.3. Výsledky testování

Výsledky testování považuji za úspěšné, jelikož aplikace je při používání stabilní, nepadá a výrazně se neseká, dále dokáže identifikovat referenční obrázek v rámci jednotek sekund a hned poté zobrazit požadovaný objekt. Také je přehledná a uživatelsky přívětivá i pro uživatele, s výrazně menším vztahem k technologické oblasti.

ZÁVĚR

Tato bakalářská práce se skládá ze dvou částí. Cílem teoretické části bylo představení rozšířené reality, její využitelnosti a přehled identifikačních a lokačních bezdrátových technologií, které jsou potřebné pro umístování digitálních objektů do skutečného světa na displeji mobilního zařízení. Cílem praktické části bylo navrhnout a realizovat aplikaci, která bude využívat identifikačních, lokačních a pohybových technologií pro práci s rozšířenou realitou. Přínosem bakalářské práce je především seznámení čtenáře s technologií rozšířené reality a popis možného postupu, který lze využít při vytváření aplikace, která tuto technologii využívá.

Práce naplňuje cíl teoretické části díky popsání rozšířené reality a její využitelnosti ve sportu, nakupování, herním průmyslu a v sociálních sítích. Dále zde byly popsány vybrané API, díky kterým můžeme AR implementovat do mobilních aplikací. V práci byly také představeny identifikační a lokační technologie, bez kterých by použití rozšířené reality nebylo možné.

V rámci této práce byla navržena a vytvořena aplikace Superhrdinové AR, která je určena pro mobilní zařízení s operačním systémem Android. Přínosem aplikace je přiblížení superhrdinských postav uživateli, za využití rozšířené reality, konkrétně pomocí ARCore. K seznámení uživatele se superhrdinskými postavami slouží v aplikaci kvíz, ve kterém jsou otázky ohledně jednotlivých postav a knihovna, kde jsou všechny postavy spolu s informacemi o nich. Díky realizaci samotné aplikace byl splněn cíl praktické části. V práci je také popsán návrh aplikace, ve kterém byla provedena analýza a popsáno uživatelské rozhraní aplikace. Dále byl představen vývoj aplikace, ve kterém jsou popsány technologie, které byly použity, je popsána struktura celé aplikace a jsou rozebrány důležité části zdrojového kódu. V další kapitole je vysvětleno, jak se v aplikaci orientovat a co vlastně umí hotová verze.

Na závěr je v práci popsáno testování z pohledu vývojáře a z pohledu obyčejného uživatele.

POUŽITÁ LITERATURA

- [1] POETKER, Bridget. A Brief History of Augmented Reality (+Future Trends & Impact). In: *Learning Hub*. [online]. 22. 08. 2019. [cit. 24. 04. 2021].
Dostupné z: <https://learn.g2.com/history-of-augmented-reality>.
- [2] WONG, Raymon. Microsoft Hololens 2 ushers in the next generation of augmented reality. In: *Mashable*. [online]. 24. 02. 2019. [cit. 24. 04. 2021].
Dostupné z: <https://mashable.com/article/microsoft-hololens-2-mwc-2019/?europe=true>.
- [3] SUN Sophia, BJORAN Kristina a JOMORR. Add annotations in Dynamics 365 Remote Assist on HoloLens. In: *Microsoft*. [online]. 10. 12. 2020. [cit. 24. 04. 2021].
Dostupné z: <https://docs.microsoft.com/en-us/dynamics365/mixed-reality/remote-assist/add-annotations-hololens>.
- [4] GREENWALD, Will. Augmented Reality (AR) vs. Virtual Reality (VR): What's the Difference? In: *PCMag*. [online]. 31. 03. 2021. [cit. 24. 04. 2021].
Dostupné z: <https://www.pcmag.com/news/augmented-reality-ar-vs-virtual-reality-vr-whats-the-difference>
- [5] OCULUS. Introducing Oculus Quest 2. In: *Youtube*. [online]. 16. 9. 2020. [cit. 24. 04. 2021].
Dostupné z: <https://www.youtube.com/watch?v=ATVGI9wOJsM>.
- [6] WTVISION. Augmented Reality for Football Broadcasts. In: *Youtube*. [online]. 22. 01. 2019. [cit. 24. 04. 2021].
Dostupné z: https://www.youtube.com/watch?v=9zD_YQBaZEY.
- [7] SWETA, Augmented Reality In Sports: Sprucing Up The Entire Industry. In: *Queppelin*. [online]. 02. 05. 2019. [cit. 24. 04. 2021].
Dostupné z: <https://www.queppelin.com/augmented-reality-in-sports/>.
- [8] SKY CREATIVE. Sky Sports Premier League AR Studio. In: *Vimeo*. [online]. 11. 07. 2019. [cit. 24. 04. 2021]. Dostupné z: <https://vimeo.com/347597299>.
- [9] IKEA Place App. In: *IKEA*. [online]. © 1999 – 2021. [cit. 24. 04. 2021].
Dostupné z: <https://www.ikea.com/ch/en/customer-service/mobile-apps/#6f254e20-867a-11eb-99f0-93776650d90f>.
- [10] O'HEAR, Steve. Wanna Kicks, a new AR app from Wannaby, lets you virtually 'try on' your next pair of kicks. In: *Tech Crunch*. [online]. 31. 01. 2019. [cit. 24. 04. 2021].

- Dostupné z: <https://techcrunch.com/2019/01/30/wanna-kicks-a-new-ar-app-from-wannaby-lets-you-virtually-try-on-your-next-pair-of-kicks/>.
- [11] App Store Preview. In: *Apple*. [online]. © 2021. [cit. 24. 04. 2021]. Dostupné z: <https://apps.apple.com/us/developer/wannaby-inc/id1544517897>.
- [12] KARAMCHANDANI, Harsh. Augmented Reality Shopping is Changing the Way you Buy. In: *WOWSOME*. [online]. © 2020 [cit. 24. 04. 2021]. Dostupné z: <https://www.wowso.me/blog/augmented-reality-shopping-is-changing-the-way-you-buy>
- [13] MCKINNON, Tricia. 10 of the Best Augmented Reality (AR) Shopping Apps to Try Today. In: *Indigo9 Digital*. [online]. 08. 05. 2020. [cit. 24. 04. 2021]. Dostupné z: <https://www.indigo9digital.com/blog/how-six-leading-retailers-use-augmented-reality-apps-to-disrupt-the-shopping-experience>.
- [14] IKEA. Say Hej to IKEA Place. In: *Youtube*. [online]. 12. 09. 2017. [cit. 24. 04. 2021]. Dostupné z: <https://www.youtube.com/watch?v=UudV1VdFtuQ>
- [15] Pokémon GO. In: *Pokémon*. [online]. © 2021. [cit. 24. 04. 2021]. Dostupné z: <https://www.pokemon.com/us/pokemon-video-games/pokemon-go/>.
- [16] IQBAL, Mansoor. Pokémon GO Revenue and Usage Statistics (2020). In: *Business of Apps*. [online]. 08. 03. 2021. [cit. 24. 04. 2021]. Dostupné z: <https://www.businessofapps.com/data/pokemon-go-statistics/>
- [17] FOREMAN, Rich. 4 Reasons Behind Pokémon Go's Wild Success. In: *Startup Grind*. [online]. © 2021. [cit. 24. 04. 2021]. Dostupné z: <https://www.startupgrind.com/blog/4-reasons-behind-pokemon-gos-wild-success/>.
- [18] KOŘOUSKOVÁ, Barbora. Jak a proč (ne)hrát Harry Potter: Wizards Unite, poradíme jak na energii. In: *SvětAndroida.cz*. [online]. 08. 07. 2019. [cit. 24. 04. 2021]. Dostupné z: <https://www.svetandroida.cz/wizards-unite-energie/>
- [19] [Pokémon GO]. In: *The Conversation* [online]. 18. 10. 2016. [cit. 24. 04. 2021]. Dostupné z: <https://theconversation.com/what-went-wrong-with-pokemon-go-three-lessons-from-its-plummeting-player-numbers-67135>
- [20] KÁRNÍK, Jakub. Hra Angry Birds AR: Isle of Pigs vychází na Android. A je sakra zábavná! In: *SvětAndroida.cz*. [online]. 04. 11. 2019. [cit. 24. 04. 2021]. Dostupné z: <https://www.svetandroida.cz/angry-birds-ar-isle-of-pigs-android/>

- [21] KHALID, Amrita a LARSON, Selena. How to get Snapchat's selfie Lenses. In: *Daily Dot*. [online]. 01. 27. 2021. [cit. 24. 04. 2021].
Dostupné z: <https://www.dailydot.com/debug/new-snapchat-selfie-lenses/>
- [22] Introducing Face Filters & More on Instagram. In: *Instagram*. [online].
16. 05. 2017. [cit. 24. 04. 2021].
Dostupné z: <https://about.instagram.com/blog/announcements/introducing-face-filters-and-more-on-instagram>
- [23] The Brief History of Social Media AR Filters. In: *INDE*. [online]. [cit. 24. 04. 2021].
Dostupné z: <https://www.industry.com/blog/the-brief-history-of-social-media-ar-filters>
- [24] In: *WIRED* [online]. 28. 03. 2017. [cit. 24. 04. 2021].
Dostupné z: <https://www.wired.co.uk/article/facebook-filters-like-snapchat>
- [25] In: *Funzen* [online]. © 2021. [cit. 24. 04. 2021].
Dostupné z: <https://www.funzen.net/2019/10/12/how-to-use-the-joker-filter-in-your-instagram-stories/>
- [26] In: *SnapchatGuide* [online]. © 2021. [cit. 24. 04. 2021].
Dostupné z: <https://snapchatguide.blogspot.com/2016/04/snapchat-filters-list-and-how-to-use.html>
- [27] In: *HubSpire* [online]. © 2021. [cit. 24. 04. 2021].
Dostupné z: <https://www.hubspire.com/resources/general/application-programming-interface/>
- [28] TECHTARGET CONTRIBUTOR. ARCore. In: *TechTarget*. [online]. 01. 11. 2017. [cit. 24. 04. 2021].
Dostupné z: <https://whatis.techtarget.com/definition/ARCore>
- [29] WILLIAMS, Robert. Google shuts down Tango AR platform. In: *Marketing Dive*.
18. 12. 2017. [cit. 24. 04. 2021].
Dostupné z: <https://www.marketingdive.com/news/google-shuts-down-tango-ar-platform/513284/>
- [30] JAEHING, Johnathan. Is ARCore or ARKit Better For Creating Augmented Reality Experiences? In: *Make us of*. [online]. 16. 11. 2020. [cit. 24. 04. 2021].
Dostupné z: <https://www.makeuseof.com/is-arcore-or-arkit-better-for-augmented-reality/>
- [31] CHRISTENSSON, Per. GPS Definition. In: *TechTerms*. [online] 2006. [cit. 24. 04. 2021]. Dostupné z: <https://techterms.com/definition/gps>

- [32] GPS Accuracy. In: *gps.gov* [online] 22. 04. 2020. [cit. 24. 04. 2021].
Dostupné z: <https://www.gps.gov/systems/gps/performance/accuracy/>
- [33] OČENÁŠKOVÁ, Adéla. Navigaci GPS vyrostla konkurence. Evropský Galileo najde ztraceného člověka až osmnáctkrát rychleji. In: *Aktuálně.cz* [online].
15. 12. 2016. [cit. 25. 04. 2021].
Dostupné z: <https://zpravy.aktualne.cz/zahranici/galileo/r~a5b34aa0c13a11e6addf0025900fea04/>
- [34] MAJER, Dušan. Galileo dnes ožívá! In: *kosmonautix.cz* [online].
15. 12. 2016. [cit. 25. 04. 2021]. Dostupné z: <https://kosmonautix.cz/2016/12/galileo-dnes-oziva/>
- [35] Galileo is the European global satellite-based navigation systém. In: *European Global Navigation Satellite Systems Agency*. [online]. 17. 02. 2021. [cit. 25. 04. 2021].
Dostupné z: <https://www.gsa.europa.eu/european-gnss/galileo/galileo-european-global-satellite-based-navigation-system>
- [36] Galileo Services. In: *European Global Navigation Satellite Systems Agency* [online].
14. 04. 2020. [cit. 25. 04. 2021].
Dostupné z: <https://www.gsa.europa.eu/galileo/services>
- [37] PRS. In: *European Global Navigation Satellite Systems Agency*. [online].
12. 10. 2018. [cit. 25. 04. 2021]. Dostupné z: <https://www.gsa.europa.eu/security/prs>
- [38] Search and Rescue (SAR) / Galileo Service. In: *European Global Navigation Satellite Systems Agency*. [online]. 01. 04. 2020. [cit. 25. 04. 2021].
Dostupné z: <https://www.gsa.europa.eu/european-gnss/galileo/services/search-and-rescue-sar-galileo-service>
- [39] System. In: BeiDou Navigation Satellite System. [online]. [cit. 25. 04. 2021].
Dostupné z: <http://en.beidou.gov.cn/SYSTEMS/System/>
- [40] CHENGQI, Ran. Development of the BeiDou Navigation Satellite System. In: National Academy of Sciences. [online]. 2012. [cit. 25. 04. 2021].
Dostupné z: <https://www.nap.edu/read/13292/chapter/6>
- [41] GLONASS. In: *International Laser Ranging Service* [online].
18. 10. 2017. [cit. 25. 04. 2021].
Dostupné z: https://ilrs.gsfc.nasa.gov/missions/satellite_missions/current_missions/g137_general.html

- [42] SUBIRANA, Jaume, ZORNOZA, Juan a HERNÁNDEZ-PAJARES, Manuel. *GNSS Data Processing, Vol. I: Fundamentals and Algorithms*. ESA Communications. 2013. ISBN 978-92-9221-886-7
- [43] JONES, Kipp. Using Wi-Fi for Indoor Positioning. In: *Skyhook*. [online]. 31. 03. 2020. [cit. 25. 04. 2021]. Dostupné z: <https://www.skyhook.com/blog/using-wifi-for-indoor-positioning>
- [44] LESONNEC, Nicolas. The power of Wifi geolocation. In: *Medium*. [online]. 15. 04. 2019. [cit. 25. 04. 2021]. Dostupné z: <https://medium.com/@nlesconnec/the-power-of-wifi-geolocation-70d2494b066d>
- [45] ESP Holding. JAK FUNGUJÍ RFID ČTEČKY. In: *WSTUDIO.cz* [online]. © 2011-2020. [cit. 25. 04. 2021]. Dostupné z: <https://esp.cz/cs/blog/funguji-rfid-ctecky>
- [46] Co je NFC? In: *alza.cz* [online]. 06. 03. 2020. [cit. 25. 04. 2021]. Dostupné z: <https://www.alza.cz/co-je-nfc>
- [47] MARWAGA, Ganeshji. Mobile Payments: What is NFC Card Emulation Mode? In: *Ganeshji – a Congenial Perspective* [online]. 07. 08. 2014. [cit. 25. 04. 2021]. Dostupné z: <http://www.gmarwaha.com/blog/2014/08/07/mobile-payments-what-is-nfc-card-emulation-mode/>
- [48] PAVLÍČEK, Michal. Android Beam končí, Google jej odstranil z nového Androidu Q. In: *mobilenet.cz*. [online]. 09. 05. 2019. [cit. 25. 04. 2021]. Dostupné z: <https://mobilenet.cz/clanky/android-beam-konci-google-jej-odstranil-z-noveho-androidu-q-37840>
- [49] JIA, Rachel. The Comprehensive Guide to 1D and 2D Barcodes. In: *Dynamsoft* [online]. 02. 02. 2020. [cit. 24. 04. 2021]. Dostupné z: <https://www.dynamsoft.com/blog/insights/the-comprehensive-guide-to-1d-and-2d-barcodes/>
- [50] [Barcode EAN13]. In: *Barcodes CZ* [online]. © 2021. [cit. 25. 04. 2021]. Dostupné z: <https://barcodes.cz/en/ean-13-barcodes/?lang=en>
- [51] QR Codes 101: a Beginner's Guide. In: *qr-code-generator.com*. [online]. © 2021. [cit. 24. 04. 2021]. Dostupné z: <https://www.qr-code-generator.com/qr-code-marketing/qr-codes-basics/>
- [52] Why are Quiet Zones So Important? In: *Electronic Imaging Materials* [online]. © 2021. [cit. 24. 04. 2021]. Dostupné z: <https://barcode-labels.com/why-are-quiet-zones-so-important/>

- [53] What is a 2D code? In: *Denso Wave* [online]. [cit. 24. 04. 2021].
Dostupné z: <https://www.denso-wave.com/en/adcd/fundamental/2dcode/2dcode/index.html>
- [54] Biometrics: definition, use cases and latest news. In: *Thales Group* [online].
06. 04. 2021. [cit. 25. 04. 2021].
Dostupné z: <https://www.thalesgroup.com/en/markets/digital-identity-and-security/government/inspired/biometrics>
- [55] Používání Face ID na iPhoneu nebo iPadu Pro. In: *Apple* [online].
05. 04. 2021. [cit. 25. 04. 2021]. Dostupné z: <https://support.apple.com/cs-cz/HT208109>
- [56] WINDER, Dave. Hackers Claim ‘Any’ Smartphone Fingerprint Lock Can Be Broken In 20 Minutes. In: *Forbes*. [online]. 02. 11. 2019. [cit. 25. 04. 2021].
Dostupné z: <https://www.forbes.com/sites/daveywinder/2019/11/02/smartphone-security-alert-as-hackers-claim-any-fingerprint-lock-broken-in-20-minutes/?sh=2544cb0c6853>
- [57] TECHTARGET CONTRIBUTOR. 3D Model. In: *TechTarget*. [online].
01. 09. 2016. [cit. 29. 04. 2021].
Dostupné z: <https://whatis.techtarget.com/definition/3D-model>
- [58] JARRATT, Steve a CREATIVE BLOG STAFF. The best 3D modelling software in 2021. In: *Creative Blog*. [online]. 21. 01. 2021. [cit. 25. 04. 2021].
Dostupné z: <https://www.creativebloq.com/features/best-3d-modelling-software>
- [59] KING, Mertel. Is Android Really Open Source? And Does It Even Matter? In: *makeuseof* [online]. 28. 03. 2016. [cit. 25. 04. 2021].
Dostupné z: <https://www.makeuseof.com/tag/android-really-open-source-matter/>
- [60] Mobile Operating System Market Share Worldwide. In: *StatCounter* [online].
© 1999-2021. [cit. 29. 04. 2021]. Dostupné z: <https://gs.statcounter.com/os-market-share/mobile/worldwide>
- [61] *edureka!* [online]. 12. 01. 2021. [cit. 29. 04. 2021].
Dostupné z: <https://www.edureka.co/blog/interview-questions/top-android-interview-questions-for-beginners/>
- [62] Android Architecture. In: *Tutlane* [online]. ©2021. [cit. 29. 04. 2021].
Dostupné z: <https://www.tutlane.com/tutorial/android/android-architecture>

- [63] Android Architecture – 5 Components of Android Architecture.
In: *DataFlair* [online]. © 2021. [cit. 29. 04. 2021]. Dostupné z: <https://dataflair.training/blogs/android-architecture/>
- [64] SINGLA, Chitra. Functional vs Non Functional Requirements.
In: *GeeksForGeeks*. [online]. 29. 04. 2020. [cit. 29. 04. 2021].
Dostupné z: <https://www.geeksforgeeks.org/functional-vs-non-functional-requirements/>
- [65] ČÁPKA, David. Lekce 2 - UML - Use Case Diagram. In: *itnetwork.cz* [online]. ©2021. [cit. 29. 04. 2021]. Dostupné z: <https://www.itnetwork.cz/navrh/uml/uml-use-case-diagram>
- [66] ČÁPKA, David. Lekce 3 - UML - Use Case Specifikace. In: *itnetwork.cz* [online]. © 2021. [cit. 29. 04. 2021]. Dostupné z: <https://www.itnetwork.cz/navrh/uml/uml-use-case-specifikace-diagram>
- [67] REJNKOVÁ, Petra. Diagram aktivit. In: *uml.czweb.org* [online]. © 2009. [cit. 29. 04. 2021]. Dostupné z: http://uml.czweb.org/diagram_aktivit.htm
- [68] Layouts. In: *Google Developers* [online]. 07. 01. 2020 [cit. 29. 04. 2021].
Dostupné z: <https://developer.android.com/guide/topics/ui/declaring-layout>
- [69] KAMAL, Fuad. ConstraintLayout Tutorial for Android: Complex Layouts. In: *Raywenderlich* [online]. 11. 02. 2019. [cit. 25. 04. 2021].
Dostupné z: <https://www.raywenderlich.com/9475-constraintlayout-tutorial-for-android-complex-layouts>
- [70] HARCHANI, Ritik. Always use Constraint Layout for creating Complex layouts. In: *Medium* [online]. 12. 06. 2021. [cit. 25. 04. 2021].
Dostupné z: <https://medium.com/nybles/heres-why-you-should-always-use-constraint-layout-for-creating-complex-layouts-in-android-42802f0a3b58>
- [71] FrameLayout. In: *Google Developers* [online]. 21. 04. 2021. [cit. 25. 04. 2021].
Dostupné z: <https://developer.android.com/reference/android/widget/FrameLayout>
- [72] ARORA, Niharika. A Battle towards Performance- Constraint Layout vs Other Layouts(Part -1). In: *ProAndroidDev* [online]. 01. 10. 2019. [cit. 29. 04. 2021].
Dostupné z: <https://proandroiddev.com/constraintlayout-vs-other-layouts-a-battle-towards-performance-part-1-14d8116e876e>
- [73] JENKOV, Jacob. Android Activity. In: *Jenkov* [online]. 30. 10. 2014. [cit. 29. 04. 2021]. Dostupné z: <http://tutorials.jenkov.com/android/activity.html>

- [74] Android Fragments. In: *JavaTpoint*. [online]. © 2011 – 2018. [cit. 29. 04. 2021].
Dostupné z: <https://www.javatpoint.com/android-fragments>
- [75] JAIN, Ayusch. Build your first Android AR app using ARCore and Sceneform.
In: *Flexiple* [online]. © 2016 – 2020. [cit. 29. 04. 2021].
Dostupné z: <https://flexiple.com/blog/build-your-first-android-ar-app-using-arccore-and-sceneform/>
- [76] Android TextView with Examples. In: *Tutlane* [online]. © 2021. [cit. 29. 04. 2021].
Dostupné z: <https://www.tutlane.com/tutorial/android/android-textview-with-examples>
- [77] ProgressBar. In: *Google Developers* [online]. 21. 04. 2021. [cit. 29. 04. 2021].
Dostupné z: <https://developer.android.com/reference/android/widget/ProgressBar>
- [78] Android List View. In: *tutorialspoint* [online]. © 2021. [cit. 29. 04. 2021].
Dostupné z: https://www.tutorialspoint.com/android/android_list_view
- [79] Buttons. In: *Google Developers* [online]. 18. 11. 2020. [cit. 29. 04. 2021].
Dostupné z: <https://developer.android.com/guide/topics/ui/controls/button>
- [80] Android Programming Unleashed: Laying Controls in Containers.
In: *Informit* [online]. 19. 12. 2012. [cit. 29. 04. 2021].
Dostupné z: <https://www.informit.com/articles/article.aspx?p=1970144&seqNum=5>
- [81] ImageView. In: *Google Developers* [online]. 21. 04. 2021. [cit. 29. 04. 2021].
Dostupné z: <https://developer.android.com/reference/android/widget/ImageView>
- [82] DUCROHET, Xavier, NORBYE, Tor a CHOU, Katherine. Android Studio: An IDE built for Android. In: *Google Developers* [online]. 15. 05. 2013. [cit. 29. 04. 2021].
Dostupné z: <https://android-developers.googleblog.com/2013/05/android-studio-ide-built-for-android.html>
- [83] LARDINOIS, Frederic. Kotlin is now Google's preferred language for Android app development. In: *Tech Crunch* [online]. 07. 05. 2019. [cit. 29. 04. 2021].
Dostupné z: <https://techcrunch.com/2019/05/07/kotlin-is-now-googles-preferred-language-for-android-app-development/>
- [84] Android Studio downloads. In: *Google Developers* [online]. [cit. 29. 04. 2021].
Dostupné z: <https://developer.android.com/studio>
- [85] Run apps on the Android Emulator. In: *Google Developers* [online].
12. 10. 2020. [cit. 29. 04. 2021].
Dostupné z: <https://developer.android.com/studio/run/emulator>
- [86] Java Introduction. In: *W3School* [online]. © 1999 – 2021. [cit. 29. 04. 2021].
Dostupné z: https://www.w3schools.com/java/java_intro.asp

[87] JAIN, Ayusch. How to Build an Augmented Images Application with ARCore.

In: *Medium* [online]. 28. 11. 2018. [cit. 29. 04. 2021].

Dostupné z: <https://medium.com/free-code-camp/how-to-build-an-augmented-images-application-with-arcore-93e417b8579d>

OBSAH CD

Obsah přiloženého CD je následující:

- *elektronická verze práce bp_Dolejs_AR.pdf;*
- *kompletní projekt aplikace Superhrdinové AR obsahující zdrojové kódy;*
- *spustitelná aplikace Superhrdinové AR.*