

Mendelova univerzita v Brně
Provozně ekonomická fakulta

Hra založená na rozšířené realitě pro platformu iOS

Bakalářská práce

Vedoucí práce:
Ing. David Procházka, Ph.D.

Aleš Kocur

Brno 2015

Chtěl bych především poděkovat svým rodičům za jejich důslednou podporu v průběhu celého studia, přítelkyni za shovívavost nad mými hodinami strávenými před počítačem, kolegům z The Funtasty za brainstorming při vymýšlení konceptu a také Ing. Davidu Procházkovi, Ph.D. za příkladné vedení práce a cenné připomínky, rady a nápady.

Čestné prohlášení

Prohlašuji, že jsem tuto práci: **Hra založená na rozšířené realitě pro platformu iOS**

vypracoval samostatně a veškeré použité prameny a informace jsou uvedeny v seznamu použité literatury. Souhlasím, aby moje práce byla zveřejněna v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách ve znění pozdějších předpisů, a v souladu s platnou *Směrnicí o zveřejňování vysokoškolských závěrečných prací*.

Jsem si vědom, že se na moji práci vztahuje zákon č. 121/2000 Sb., autorský zákon, a že Mendelova univerzita v Brně má právo na uzavření licenční smlouvy a užití této práce jako školního díla podle § 60 odst. 1 Autorského zákona.

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

V Brně dne 18. května 2015

.....

Abstrakt

Tato práce se zabývá rozšířenou realitou a jejím využitím ve hrách na platformě iOS. Zkoumá aktuální přístupy her, které rozšířenou realitu využívají. Práce popisuje jejich koncepty interakce a přidaných hodnot oproti běžným hrám (bez rozšířené reality). Dále jsou představeny frameworky umožňující implementaci aplikací založených na rozšířené realitě pro mobilní platformu iOS a jejich srovnání na základě vhodnosti pro tvorbu hry. Poslední částí práce je návrh vlastní hry s důrazem na vyřešení či zamezení nedostatků vyzorovaných na již existujících hrách a jejich konceptech a následná implementace s použitím nejlépe ohodnoceného frameworku.

Abstract

This thesis discusses an augmented reality and its integration in games for iOS platform. It outlines different kinds of approaches used in currently popular augmented reality games, namely their concepts and their benefits over usual games (without usage of the augmented reality). It describes frameworks that can be used to develop applications with an augmented reality for iOS and comparison of their features and suitability to be the foundation of an augmented reality game. The key section describes design of my own game which aims to avoid critical drawbacks found by observing existing games, as well as implementation based on the most suitable framework.

Klíčová slova

rozšířená realita, iOS, Objective-C, Metaio framework

Keywords

augmented reality, iOS, Objective-C, Metaio framework

Obsah

1	Úvod a cíl práce	11
1.1	Úvod	11
1.2	Cíl práce	11
2	Rozšířená realita	12
2.1	Metody vizuálního sledování	12
2.2	Srovnání	14
3	Využití rozšířené reality, koncepty využité ve hrách	15
3.1	Interakce se zařízením	15
3.2	Interakce s herní deskou	16
3.3	Smíšená interakce	17
3.4	Shrnutí	18
4	Framework	19
4.1	Kritéria hodnocení	19
4.2	Přehled frameworků	19
5	Metodika	22
5.1	Koncept hry	22
5.2	Výběr frameworku	22
5.3	Formát 3D modelů	23
5.4	Shrnutí	24
5.5	Modelovací nástroj	24
6	ARCity	25
6.1	Rozbor konceptu	25
6.2	Metaio	26
6.3	Cocoapods	28
6.4	Model	28
6.5	Uživatelské rozhraní	30
6.6	Ukládání dat	35
6.7	Výpočty ukazatelů	35
7	Testování uživatelského rozhraní	38
7.1	Základní přístupy testování	38
7.2	Heuristická analýza ARCity	40
7.3	Zhodnocení	41
8	Závěr	43
8.1	Zhodnocení dosažení cílů	43
8.2	Návrhy na vylepšení	43

9 Přílohy	45
9.1 Datové přílohy	45
9.2 Ukázky z práce	46
10 Reference	49

1 Úvod a cíl práce

1.1 Úvod

Myšlenku rozšířené reality (angl. *Augmented reality*) nastínil již před více než sto lety americký spisovatel *Lyman Frank Baum* (BAUM, 1901), ale až v posledních letech s nástupem mobilních technologií získává na svém potencionálu více než kdy předtím. Mobilní technologie a přenositelná zařízení nám umožňují pomocí senzorů snímat naše okolí a obohacovat ji o uměle vytvořené prvky. Toho se dá využít v celé škále odvětví, jako je zábavní průmysl, medicína, marketing, vzdělávání, navigace, sport a mnoho dalšího. Mobilní zařízení jsou v současné době nejrychleji rostoucím odvětvím (GSMA Mobile Economy, 2015), jejich hardwarové konfigurace zajišťují prostředky pro bezproblémové vykreslování detailních modelů i chod herního enginu. Mnohé firmy vidí v AR a VR (Virtual reality) budoucnost hraní a proto investují do jejich výzkumu (The Guardian, 2015). Tato práce se zabývá aplikací rozšířené reality v zábavním průmyslu, konkrétně hrami.

1.2 Cíl práce

Cílem práce je prozkoumat možnosti her v rozšířené realitě na mobilní platformě iOS a vytvořit hru, která bude rozšířené reality využívat jako obohacujícího prvku pro hráče. Prvním a zároveň velmi důležitým krokem je zvolení správného frameworku. Poté je potřeba důkladně navrhnout herní systém a využití rozšířené reality. Dalším krokem bude připravení grafických podkladů a modelů pro hru a následně samotná implementace. Poslední kroky jsou pak věnovány uživatelskému testování, vyhodnocení a diskuzí nad možnými vylepšeními.

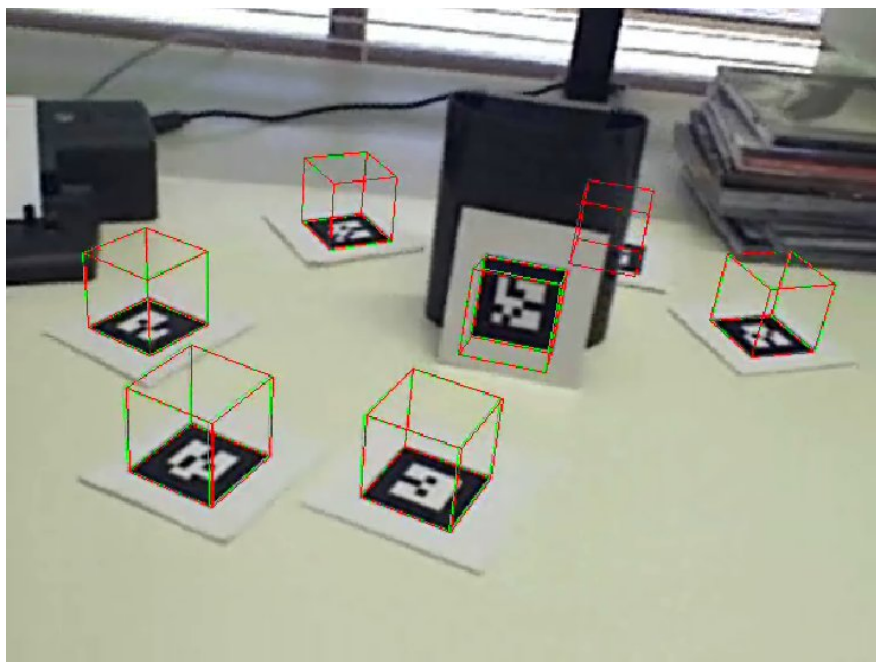
2 Rozšířená realita

Rozšířenou realitou (angl. *Augmented reality*) je označováno zobrazování digitálních objektů (3D modelů, 2D obrazů) v reálném světě. Tohoto efektu lze dosáhnout pomocí tzv. okna, které nám sdružuje reálné prostředí s virtuálním. Může se jednat o jakoukoliv formu displeje, od mobilního telefonu až po sofistikované nástroje, jako jsou speciální brýle (např. *Google Glass*). Digitálním objektům je možné pomocí různých technik analýzy obrazu specifikovat pozici, natočení a velikost. Metody vizuálního sledování lze rozdělit do třech základních kategorií (Klein, 2012).

2.1 Metody vizuálního sledování

Marker tracking

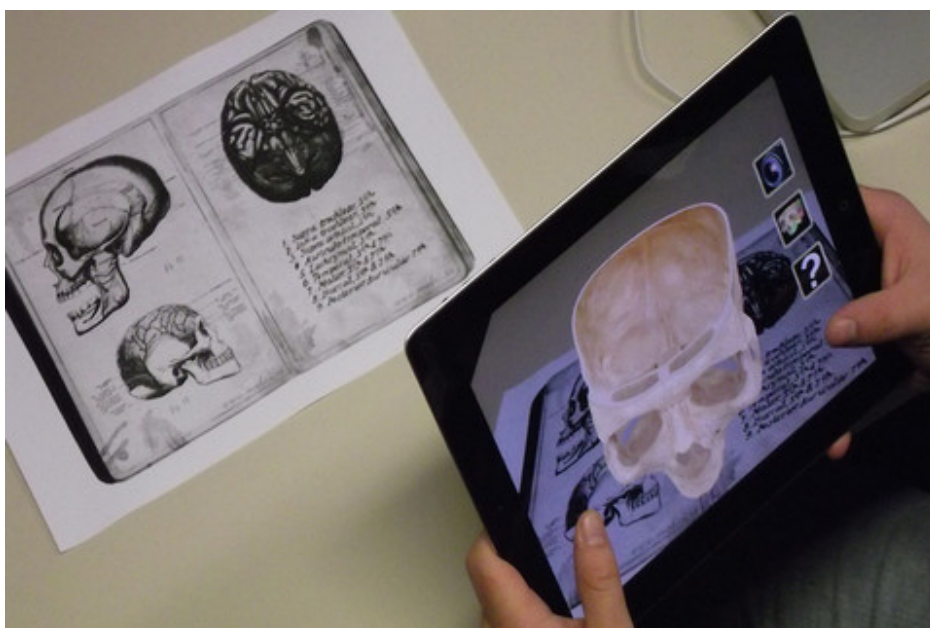
Jedním z nejjednodušších a také nejspolehlivějších rozpoznávacích technik postavení objektů je *Marker Tracking*. Jedná se většinou o černobílé čtvercové *QR kódy*, které uchovávají informaci o svém identifikačním čísle, orámované černým okrajem o pevné velikosti. Na základě přečteného identifikátoru lze přiřazovat jednotlivým markerům různé objekty. Černé rámování slouží k analýze vzdálenosti od markeru, jeho natočení a velikost. Tato metoda je velmi efektivní, analýza takového markeru na zařízení *iPhone 5* s frameworkem *Metaio* v obrazu zabere průměrně 4,3 ms (na základě vlastního měření). Další výhodou této metody je možnost definice velkého množství markerů s různým identifikátorem a na každém identifikátoru pak lze zobrazovat odlišné objekty.



Obrázek 1: Marker tracking. Zdroj: (VTTRearch, 2015)

Markerless tracking

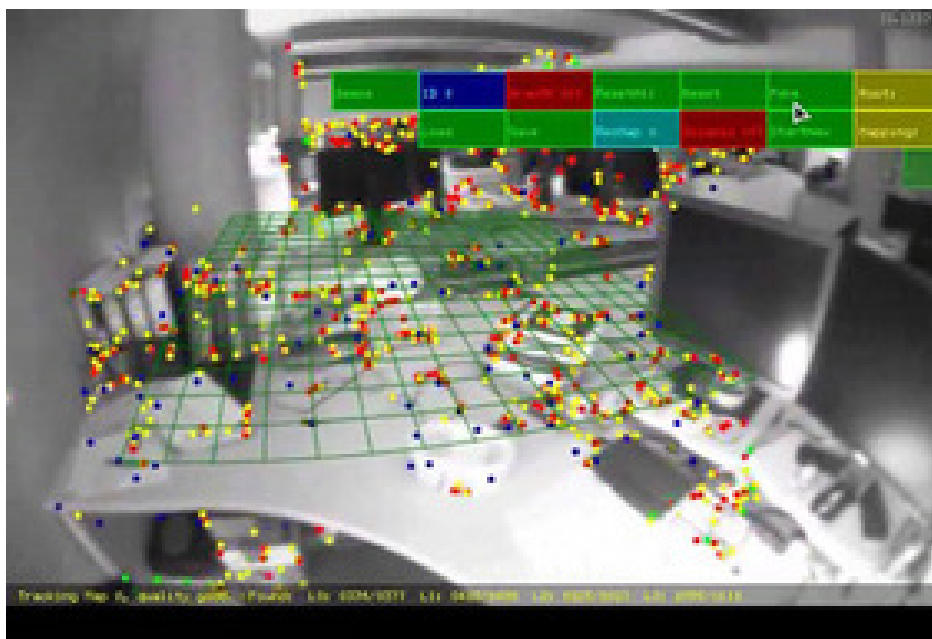
Další možnost jak umisťovat objekty do rozšířené reality je pomocí tzv. *Markerless trackingu*. Jedná se o princip podobný Marker trackingu s tím rozdílem, že namísto markerů jsou použity libovolné obrazy. Při analýze je pak potřeba vlastnit digitální předlohu takového obrazu a vyhledávat jej ve snímané realitě. Na základě porovnání natočení obrazu ve snímané realitě a předlohy je zjištěna aktuální vzdálenost od objektu, velikost a natočení. Z důvodu zrychlení je v průběhu snímání porovnáván vzor z posledního snímku. Tento princip analýzy může být pomalejší, zvláště při špatných světelných podmínkách (Borko Furht, 2011).



Obrázek 2: Markerless tracking. Zdroj: (Wikipedia, 2015)

Sledování neznámých prostředí

Umisťovat virtuální objekty do snímané reality lze i bez známých vzorů, a to pomocí detekce hran na základě pohybu kamery. Nejrozšířenější systémy takovéto analýzy jsou *SLAM* (*Simultaneous localization and mapping*) a jeho vylepšení *PTAM* (*Parallel Tracking and Mapping*). *PTAM* je vyvíjen *Active Vision Laboratory* na *University of Oxford* a od roku 2014 volně dostupný pod licencí *GNU GPLv3*. Tento systém dokáže analyzovat plochy a hrany v obrazu a na základě těchto informací pak vykreslovat správně umístěné a natočené virtuální objekty. Tyto systémy mají uplatnění mimo jiné také při navigaci autopilotovaného vozidla, vesmírných vozů a podobně.



Obrázek 3: SLAM: Simultaneous localization and mapping. Zdroj: (*Department of Engineering Science, University of Oxford, 2008*)

2.2 Srovnání

Z uvedených informací vyplývá, že implementačně nejjednodušším a nejdostupnějším řešením je *marker tracking* za předpokladu potřeby rozpoznání více konkrétních míst v realitě. Mírně složitější, ale zato uživatelsky přívětivější řešení, nabízí *markerless tracking* v podobě možnosti rozpoznávání vzorů a obrázků. Pro situace, kdy nepotřebujeme nacházet konkrétní pozice, ale spíše plochy, je zde řešení typu SLAM, které je však z uvedených typů nejnáročnější na implementaci.

3 Využití rozšířené reality, koncepty využité ve hrách

Rozšířená realita nabývá s vývojem stále výkonnější mobilních telefonů na atraktivitu a rozšiřuje se i v komerční sféře. Mobilní aplikace dokáží interagovat s realitou, např. přehráváním trailerů k filmům nad jeho plakátem, vizualizovat návrhy staveb nad prospekty nebo vizualizovat učební materiál v interaktivních učebnicích. Využití rozšířené reality je nespočetné a s přibývajícím dostupností technologií roste i počet nápadů na její uplatnění. Velké oblibě se těší zejména ve hrách, kde využívá prostředí hráče jako herní plochu a mobilní telefon jako okno do rozšířené reality.

3.1 Interakce se zařízením

Tento koncept je postaven na principu interakce hráče pouze se zařízením. Hráč vůbec nemanipuluje ani neinteraguje s prvky herního pole. Tento koncept je vhodný zejména pro typy her, ve kterých nepotřebujeme pohybovat s objekty a je charakteristický pro mapově (geolokačně) založené hry – objekty jsou umísťovány do reálného světa s koordinátami. Typickým představitelem takového konceptu je hra *Ingress*.

Ingress

Masově multiplayerová online hra vyvíjená startupem *Niantics Labs*, kterou zajišťuje *Google*, dostupná pro *iOS* i *Android*. Celá hra má v pozadí příběh o „Exotické hmotě“ (*Exotic Matter*), která byla objevena vědci z *CERNu*, a je to zárodek mimozemského druhu zvaného Shapers. Osvícení (*The Enlightened*), jedna ze dvou frakcí, věří, že toto je úsvit nového věku. Druhá frakce, Rezistence (*The Resistance*), naopak brojí proti těmto mimozemským silám. Hra spočívá ve vytváření frakčních portálů na různých místech, zejména na městských památkách, veřejných budovách a podobně. Hráči chodí s telefony po městě a vytvářením takovýchto portálů přivlastňují daná území své frakci. Mohou také stavět obranné prvky a bránit tak tyto portály před napadením frakce druhé. Zobrazování rozšířené reality je zjednodušené, pouze černé pozadí s obrysy některých budov. Hra se tedy více než na rozšířenou realitu zaměřuje na příběh a rozšířená realita zde slouží pouze jako část celé hry. Více informací lze získat na oficiálních webových stránkách hry www.ingress.com (Niantics Labs, 2015).



Obrázek 4: Screenshot ze hry Ingress, Zdroj: (Flickr, 2015)

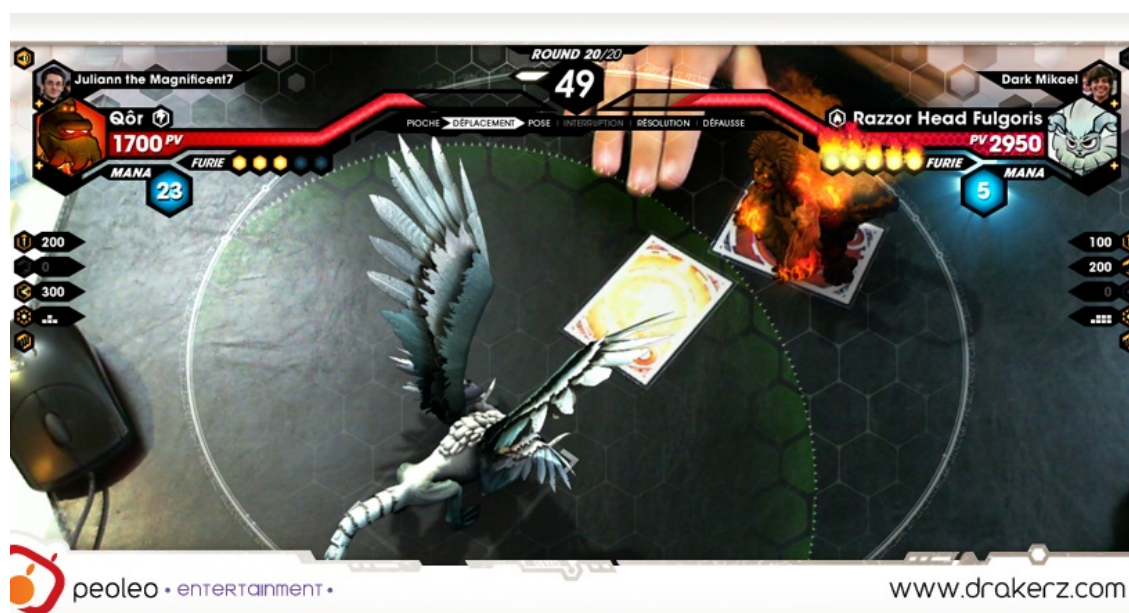
3.2 Interakce s herní deskou

Dalším konceptem je interakce s herní deskou, kdy hráč interaguje pomocí gest nebo stisknutím virtuálních tlačítek na desce. Tento koncept štěpí část interakční a zobrazovací (hráč manipuluje s prvky na desce, ale výsledek vidí pouze přes zařízení) a pro uživatele může být v případech běhu aplikace na mobilním telefonu obzvláště nepříjemný na ovládání, více vhodný je například pro brýle. Z tohoto důvodu je také tento koncept v mobilních hrách téměř nepoužívaný. Mezi hrami využívající tohoto

přístupu jsou povětšinou hry původem deskové či karetní a rozšířenou realitu zde využívají spíše pro zvýšení intenzity zážitku ze hry vizualizacemi procesů, které si jinak člověk musí pouze představovat.

Drakerz Confrontation

Karetní hra vyvíjená od roku 2010 francouzskou firmou *Peoleo* s první alpha verzí vydanou roku 2013 ve Francii (Venture Beat, 2014). Dnes je hra dostupná pouze pro PC a nejsou žádné informace, zda dorazí v budoucnu i na mobilní platformy. Hráči *Drakerz Confrontation* mají karty s různými typy draků s různými vlastnostmi a schopnostmi. Hra využívá rozšířené reality k vizualizaci draků nad kartami položenými na stole. Ke hraní je potřeba vlastnit kameru připojitelnou k PC, na kterém se pak vizualizace zobrazují. Vizualizované jsou jak pohyby draků po herní ploše, tak např. boje i prostoje. Více informací lze nalézt na oficiálních stránkách hry www.drakerz.com (Peoleo, 2015), kde je také možné zakoupit karty nebo hru stáhnout.



Obrázek 5: Screenshot ze hry Drakerz Confrontation. Zdroj: (Game guide, 2015)

3.3 Smíšená interakce

Tento koncept kombinuje předchozí dva uvedené. Hráč v tomto případě primárně používá zařízení, ale má i možnost ovládat herní prvky v reálném prostředí na herní desce. Typickým příkladem smíšené interakce je hra *ARHrrrr!*.

ARHrrrr!

Jednou z populárních her je hra *ARhrrrr!* z dílny *Georgia Tech Augmented Environments Lab*, která využívá vytištěné hrací plochy k vizualizaci části města (*Markerless tracking*), kterou napadají zombie a hráč v roli snipera v helikoptéře se pohybuje nad hrací plochou a střílí. Tento koncept s herní plochou a ovládáním výhradně přes zařízení je doplněn o různé podpůrné funkce ovládané pomocí interakce se snímaným prostředím. Jako příklad lze uvést, že položením bonbónu *Skittles* na herní plochu vznikne nášlapná mina, kterou můžeme kliknutím na ni odjistit (příklad interakce s herní plochou). Různé barvy bonbónů mají navíc různé funkce.



Obrázek 6: Screenshot ze hry ARhrrrr!. Zdroj: (Engadget, 2015)

3.4 Shrnutí

Hry využívají rozdílných konceptů podle úlohy, kterou rozšířená realita ve hře zastává. Hra *Ingress* nám příkladně ukazuje využití konceptu interakce se zařízením, není zde potřeba nijak ovlivňovat hru prostřednictvím reality. Na takovém principu staví povětšinou karetní a deskové hry, jako příklad zde uvedena hra *Drakerz Confrontation*. Tento přístup není moc uplatnitelný na mobilních zařízeních, je zde výhodnější použít statickou kameru. Třetím konceptem je pak kombinace dvou výše zmíněných, který nám dovoluje ovládat hru jak přes zařízení, tak přes realitu. Výhody tohoto konceptu jsou popsány na hře *ARhrrrr!*.

4 Framework

4.1 Kritéria hodnocení

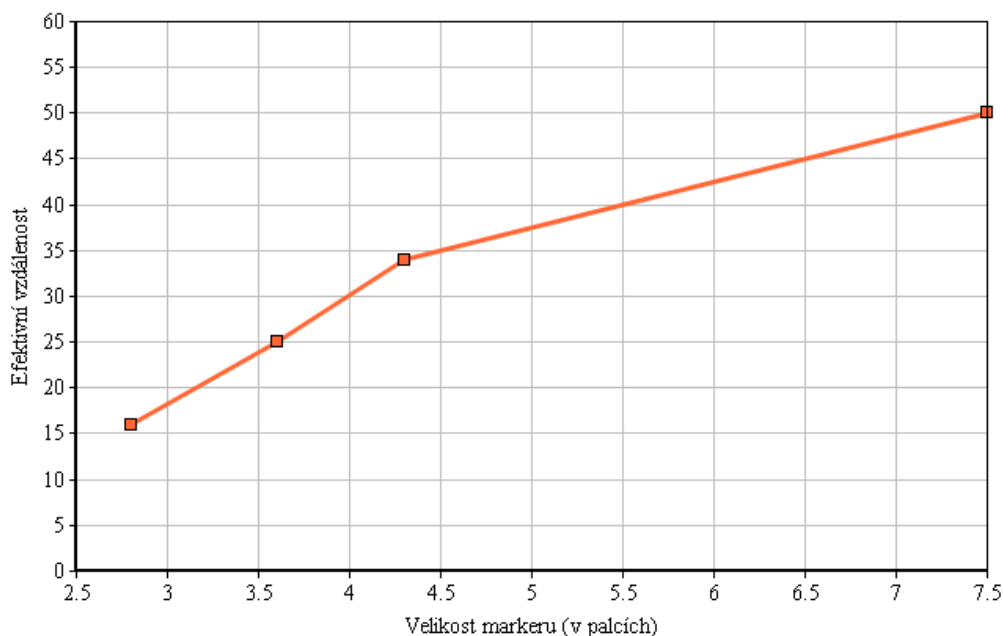
Kritéria hodnocení se odvíjí od náročnosti požadavků na aplikaci, v tomto případě se jedná o hru. Je tedy důležité, aby framework uměl dobře pracovat s 3D modely a provádět animace, aby hra byla zajímavá. Dalším aspektem je rychlost závislá na počtu zobrazovaných objektů a posledním, neméně důležitým faktorem, jsou licenční podmínky pro používání.

4.2 Přehled frameworků

ARToolkit

ARToolkit byl původně vyvinut *Hirokazu Katou* na *Nara Institute of Science and Technology* v roce 1999. Nyní je udržován *The Human Interface Technology Lab* na *University of Washington* jako open source projekt s komerčním licencem od *ARToolWork*.

Chyba sledování závislá na velikosti markeru



Obrázek 7: Chyba sledování závislá na velikosti markeru. *Originál: (HITLab, 2015)*

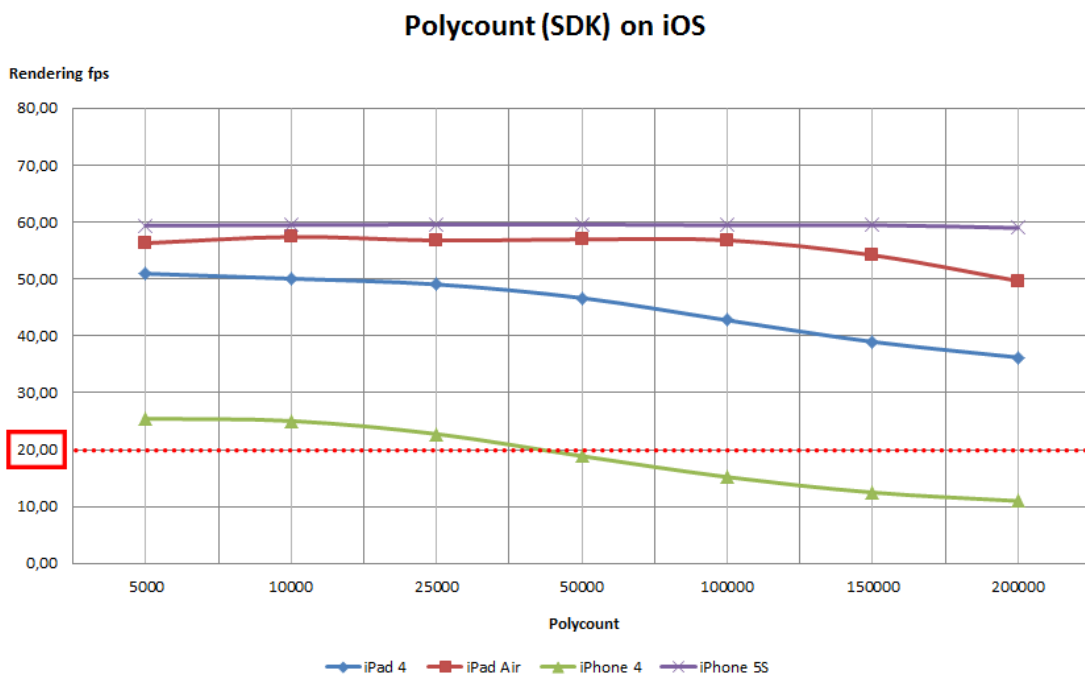
Tento framework je dostupný pro všechny majoritní platformy (*Windows*, *Linux*, *OS X*, *SGI*) s různými porty pro mobilní operační systémy (*iOS*, *Android*).

Mobilní verze tohoto frameworku portují již konkrétní firmy a vydávají jako vlastní produkt, nejsou tedy volně dostupné – nepodařilo se sestavit testovací projekt pro tento framework a benchmarky uvedené v dokumentaci jsou velmi nepřesně specifikované (chybí specifikace hardwaru, na kterém byly tyto měření prováděny).

Mezi hlavní funkce frameworku patří kamerově pozicované a orientované sledování, sledování černých čtverců s možností definice vlastních typů, jednoduchá kalibrace kamer (HITLab, 2015). Podpora vykreslování modelů je pouze nízkoúrovňová, lze renderovat objekty pomocí *OpenGL*. Většinou je využit v projektech jako část komplexního toolkitu pro práci s rozšířenou realitou např. *OSGART* (kombinace *ARToolKit* a *OpenSceneGraph*) (HITLab NZ, 2015) nebo *ARToolKit-Plus* (rozšířená verze *ARToolKitu*, vývoj ukončen roku 2006) (WAGNER, SCHMALSTIEG, 2007).

Metaio

Metaio je multiplatformní framework, který vyvíjí stejnojmenná německá firma od roku 2003. Mezi produkty *Metaio* najdeme framework pro rozšířenou realitu na všechny majoritní platformy a také sadu aplikačních nástrojů specializující se na usnadnění práce i samotný vývoj bez programování. *Metaio* pro *iOS* je dostupné jako *C++* framework. Dokumentace uvádí měření ukazující udržení výkonu vykreslování 60 snímků za vteřinu (FPS) při 200 000 polygonech na *iPhone 5S*.



Obrázek 8: Rychlost vykreslování (v FPS) pro jednotlivé iOS zařízení závislé na počtu vykreslovaných polygonů. Zdroj: (Metaio, 2015)

Také byl vytvořen testovací projekt, jehož jediným úkolem bylo rozeznávat 20 různých markerů. Na tomto projektu pak byly provedeny zkoušky rychlosti nalezení markeru v obraze. Test byl prováděn na zařízení *iPhone 5* (model A1429) firmy *Apple*. Z výsledných hodnot byl vypočítán průměr 4,31 ms.

Metaio SDK dokáže pracovat s modely ve formátech *OBJ*, *MD2* a *FBX* a podporuje také jejich animace. Pod podmínkou vodoznaku a nemožnosti publikovat aplikaci na *AppStore* nabízí volnou licenci na všechny verze svého frameworku.

Qualcomm Vuforia

Jedná se o poměrně nový framework vyvíjený od roku 2011 a nabízí velké množství funkcí. Jako hlavní lze uvést tzv. markerless recognition (možnost vykreslovat objekty nad obrázky), podporu frameworku *Unity* (framework usnadňující vývoj 3D her) nebo takzvané virtuální tlačítka (*Virtual buttons*, tlačítka promítaná do virtuální reality a interakce s nimi). Samotný framework nabízí aplikační rozhraní pro *C++*, *Javu*, *Objective-C* a *.NET*, což významným způsobem usnadňuje portace aplikace na různé platformy. Široké spektrum funkcí se odráží na licenčních podmínkách frameworku, u neplacených licencí je omezen počet rozpoznávaných objektů (pozn. v průběhu tvorby této práce byly změněny licenční podmínky a framework je dostupný zdarma s vodoznakem a nemožností publikovat aplikaci). Velkým mínusem je stejně jako v případě *ARToolKitu* pouze nízkoúrovňová podpora vykreslování objektů pomocí *OpenGL*. Pro podporu zobrazování modelů ve formátech např. *OBJ*, *FBX* – je potřeba využít nějaké další pomocné knihovny (Qualcomm, 2015).

Augmented kit

Framework psaný čistě pro platformu *iOS* a tedy v *Objective-C*. Vyniká jednoduchým API a dobrou dokumentací. Nabízí pouze základní služby vykreslování objektů na markerech či GPS souřadnic a gyroskopu. Framework je vyvíjen teprve od roku 2012 (*Luteg Software Technologies*) a to se projevuje velmi malou vývojářskou základnou. Na druhou stranu má přívětivé licenční podmínky, kdy poskytuje volnou licenci pod podmínkou vodoznaku a nemožnosti aplikaci publikovat na *AppStore*.

5 Metodika

Pro zvolení správných technologií je nutné jako první navrhnout konceptuální model celé hry. Na základě tohoto modelu je pak potřeba zvolit technologie tak, abychom dosáhli daných kritérií. Konkrétně se jedná o volbu frameworku, 3D formátu pro modely a modelovací nástroj.

5.1 Koncept hry

Nosným prvkem celé hry je rozšířená realita. Na základě informací o možných typech trackingu jsem zvolil jako trackingovou metodu *marker tracking* pro její jednoduchou implementaci a možnost rozlišení markerů podle identifikátorů. Základem hry je tedy pole markerů s různými uspořádáními – hráč si může zvolit na jakém typu herní plochy chce hrát. Na tomto poli budou vizualizovány herní modely s možností jejich animací. Pro interakci s objekty rozšířené reality je využito konceptu *Interakce se zařízením*. Hra je simulací stavby a řízení vlastního města a je spustitelná na zařízeních se systémem *iOS 8* s primárním cílením pro zařízení *iPad*.

5.2 Výběr frameworku

Pro výběr frameworku byly na základě herního konceptu stanoveny tyto kritéria:

- Spustitelnost pod iOS
- Marker tracking
- Podpora 3D formátu modelů
- Animovatelnost modelů
- Volná licence / akademická licence
- Dokumentace, tutoriály

Výsledky pro tyto kritéria jednotlivých představených frameworků můžeme vidět v Tabulce 1.

Framework	ARToolKit	Metaio	Qualcomm Vuforia	Augmented kit
Podpora iOS	<i>Částečně</i>	<i>Ano</i>	<i>Ano</i>	<i>Ano</i>
Marker tracking	<i>Ano</i>	<i>Ano</i>	<i>Ano</i>	<i>Ne</i>
3D formát modelů	<i>Ne, pouze OpenGL</i>	<i>OBJ, FBX, MD2</i>	<i>Ne, pouze OpenGL</i>	<i>Ne</i>
Animace	<i>Pouze GLUT</i>	<i>Ano (FBX, MD2)</i>	<i>Pouze GLUT</i>	<i>Ne</i>
Licence	<i>Komerční</i>	<i>Omezená s vodoznakem</i>	<i>Omezená s vodoznakem</i>	<i>Omezená s vodoznakem</i>
Dokumentace/tutoriály	<i>Ano/ne</i>	<i>Ano/ano</i>	<i>Ano/ano</i>	<i>Ano/pouze ukázka</i>

Tabulka 1: Porovnání frameworků

Po porovnání výsledků byl vyhodnocen jako nejvhodnější framework Metaio, který jako jediný splňuje všechna daná kritéria.

5.3 Formát 3D modelů

Základem dobrého návrhu aplikace je správná volba technologií. Mezi ně v tomto případě patří i volba formátu pro 3D modely. Framework Metaio podporuje renderování objektů ve formátech *OBJ*, *MD2* a *FBX*. Kritérii pro výběr formátu jsou podpora exportu z *3Ds Max*, podpora animací, podpora textur.

OBJ

Otevřený formát vyvinutý firmou *Wavefront Technologies* pro přenos 3D objektů. Formát specifikuje pozice každého vertexu v objektu, UV souřadnice textury vertexu a plochy (faces) každého polygonu (Wikipedia, 2015). Obsahuje podporu textur pomocí přiložených materiálů ve formátu *MTL (Material Template Library)*. Formát *OBJ* je standardně textový formát, může se však vyskytovat i v binární podobě. Takovýto binární soubor má pak příponu *.mod*. Formát bohužel nemá podporu animací (Paul Burke, 2015) a tím nevyhovuje v jednom ze tří bodů daných kritérií.

MD2

Formát *MD2* se stal populárním díky firmě *id Software's* (Wikipedia, 2015), která jej použila v *id Tech 2* známý jako *Quake II engine* (M.H. Williams, 2011) pro modely všech postav ve hře *Quake II* a dalších, rovněž postavených na tomto enginu. *MD2* má podporu key-frame animací a textur. *MD2* byl později nahrazen novějším formátem *MD3* (Wikipedia, 2015). Tento formát je nyní již zastaralý a nemá valnou podporu v *3Ds Maxu*, vyjma komerčního pluginu (QTIP, 2015).

FBX

Proprietární formát *FBX* (FilmBoX) je od roku 2006 vyvíjen a spravován firmou *Autodesk* (Autodesk, 2015). Firma pro užívání tohoto formátu vydala *FBX SDK*, který umožňuje zapis, čtení a konverze tohoto formátu (Autodesk, 2015). Formát podporuje animace, textury, a jelikož je vyvíjen stejnou firmou jako *3Ds Max*, má v tomto softwaru i náležitou podporu pro export.

5.4 Shrnutí

Formát	Podpora animací	Podpora textur	Podpora exportu 3Ds Max
OBJ	Ne	Ano	Ano
MD2	Ano	Ano	Pouze komerčním pluginem
FBX	Ano	Ano	Ano

Tabulka 2: Porovnání formátů pro reprezentaci 3D modelů

Na základě daných kritérií byl zvolen za nejvhodnější formát *FBX*. Metaio sice podporuje tento formát, ale využívá k tomu utility zvané MeshConvertor, které je potřeba dodat *FBX* model s případnými UV mapami textur modelu. Výsledným souborem je archiv formátu *ZIP*, který obsahuje informace o daném modelu, přiložené textury, názvy animací a metadata. Tento *ZIP* pak již lze snadno používat v *Metaio SDK*.

5.5 Modelovací nástroj

Jediné kritérium stanovené pro výběr modelovacího nástroje je podpora exportu do *FBX*. Jelikož tuto možnost mají všechny majoritní modelovací nástroje (*Autodesk 3Ds Max*, *Blender*, *Maya*, ...), rozhodl jsem se pro použití *3Ds Maxu* od firmy *Autodesk* pouze na základě předešlých znalostí prostředí tohoto programu. *Autodesk* poskytuje na své produkty akademickou licenci.

6 ARCITY

6.1 Rozbor konceptu

Základní koncept hry spočívá v budování města – budov – a řízení ekonomiky tohoto města – od toho znikl název ARCITY – ovlivňováním ekonomických parametrů (DPH, počet volných pracovních míst, obyvatelné parcely). Hra se je vizualizována na hrací desce. Rozměr hrací desky lze měnit přidáním desky další a tím navýšit rozměry z 5×5 na 5×10 až 5×15 markerů. Větší rozměry nejsou podporovány a to zejména z důvodů hardwarové náročnosti. Každý marker představuje parcelu pro vytvoření takzvané zóny. Základní surovinou pro řízení hry jsou peníze. Základními ukazateli jsou pak spokojenost a počet obyvatel. Za peníze může hráč nechat na prázdných parcelách vystavět specifické zóny. Základní typy zón jsou čtyři – obytná zóna, průmyslová zóna, obchodní zóna a kulturní zóna. Každá z těchto zón může ovlivňovat ukazatele, přinášet suroviny (peníze) nebo naopak je spotřebovávat. Uprostřed města je již od počátku hry postavena radnice, která slouží k možnosti manipulace s daněmi. Cílem hry je postavit město na nejvyšší úrovni.

Na začátku hry je hráči dovoleno stavět pouze základní zóny bez možnosti je vylepšit. Hráč může takovéto vylepšení odemknout získáním vyšší městské úrovně. Pro získání úrovně je potřeba splnit limity dané pro každou úroveň zvlášť, jedná se o kombinaci počtu obyvatel, jejich spokojenosti a městského rozpočtu. Po splnění těchto požadavků je hráči umožněno vylepšovat své zóny a tím dosáhnout větších profitů či většího počtu obyvatel nebo pracovních míst.

Obytná zóna

Slouží k navýšení maximálního počtu obyvatel, které může město ubytovat. Počet obyvatel, kteří ve městě žijí (přistěhují se nebo se odstěhují), je závislý na stavu spokojenosti a počtu pracovních míst. Zvýšením úrovně této zóny lze navýšit její obytnou kapacitu až na trojnásobek té základní.

Průmyslová zóna

Zaměstnává lidi ve městě – vytváří pracovní místa a generuje peníze. Snižuje však kulturu města (ukazatel spokojenosti). Obdobně lze také zvyšovat nabídku pracovních pozic navýšením úrovně zóny a to až na trojnásobek.

Kulturní zóna

Zvyšuje spokojenost obyvatel města, ale musí být dotována z městského rozpočtu. Zvýšením úrovně získáme možnost uspokojení více obyvatel za mírně zvýšenou dotaci.

Obchodní zóna

Stejně jako průmyslová zóna vytváří pracovní místa, ale pouze v menším množství. Nemá však takový vliv na snižování spokojenosti obyvatel. Také generuje malé množství peněz a jedná se o nejdražší zónu. Zvýšením úrovně této zóny zvyšujeme také její příjmy a počet volných pracovních míst.

Městská správa, radnice

Základní zóna, kterou má hráč již postavenou na startu hry a nelze ji zničit. Poskytuje možnost řízení velikosti daně a tím ovlivňovat příjmy a spokojenost obyvatel.

6.2 Metaio

Metaio SDK je knihovna napsaná v jazyce *C++* a pro použití v *Cocoa touch* projektu, který je vyvíjen v jazyce *Objective-C* (nebo také nově ve *Swiftu*), stačí knihovnu přilinkovat do tzv. bundle. Pro použití knihoven psaných v *C++* má *Cocoa touch* speciální formát s příponou *.mm*, kterému se říká *Objective-C++*. Touto příponou dáváme překladači najevo kombinování těchto dvou jazyků v jednom souboru.

Příprava modelů

Pro každou zónu je potřeba vytvořit 3D model. Tyto modely byly vytvářeny ve zmíněném *Autodesk 3Ds Max*, populárním softwaru pro modelování. Základem je vytvoření plochy o velikosti markeru, na kterém se pak stavěly jednotlivé zóny. Po vymodelování a otexturování následuje převod textur do takzvané *UV mapy*. Jedná se o rozložení všech textur objektů ve scéně, v našem případě do takzvaného *flatten* (plošného) rozložení, pro všechny stěny. Takto mapované textury lze vyrenderovat do jpeg souboru a následně nahrát zpět jako jedinou texturu pro celý objekt. V případě modelu selekce je potřeba ještě nadefinovat animaci. V této chvíli je objekt připraven na export do formátu FBX (viz sekce Formát 3D modelů).

Po exportu dostáváme soubor FBX obsahující model a informace o použitých texturách a keyframe animacích v případě modelu selekce. K tomuto souboru připojíme také naši vytvořenou texturu. Tu je potřeba editovat v nějakém rastrovém editoru a vytvořit kopie pro různé barvy střech. Výsledek pak zabalíme *Metaiem* dodávanou konzolovou utilitou *FBXMeshConverterOSX*.

```
1 ./FBXMeshConverterOSX \ Model source/house/level3/shopping-zone.FBX
```

Takto získáme archiv *ZIP* obsahující model, textury a animace.

Nahrávání modelů

Následující kód ukazuje inicializaci 3D modelu a jeho následnou konfiguraci.

```

1 NSString *modelPath = [[NSBundle mainBundle]
2     pathForResource:@"selection" ofType:@"zip"];
3
4 NSAssert(modelPath != nil, @"Cannot find model path!");
5
6 graphicsCore.selection = metaioSDK->createGeometry(
7     [modelPath UTF8String]);
8
9 NSAssert(graphicsCore.selection != nil, @"Cannot load plot geometry!");
10
11 graphicsCore.selection->setScale(metaio::Vector3d(6, 6, 6));
12 graphicsCore.selection->setRotation(metaio::Rotation(M_PI_2, 0, 0));
13 graphicsCore.selection->setCoordinateSystemID(1000);
14 graphicsCore.selection->setVisible(false);
15 graphicsCore.selection->setTransparency(0.4f);
16 graphicsCore.selection->setAnimationSpeed(12.0f);

```

Konkrétně jde o nahrání modelu zobrazovaného jako indikátor označení zóny. Po úspěšném vytvoření geometrie jsou jí nastaveny základní parametry jako zvětšení, rotace či rychlost animace. Asercí jsou ověřeny stavy kritických chyb.

Spouštění animací

Pokud jsme modelu vytvořili v modelovacím programu animaci a řádně ji s modelem vyexportovali a následně přibalili do zip archivu, *Metaio* se postará o její zpracování. Jediné co je potřeba znát, je pojmenování dané animace. Tu pak lze pustit jednoduše jednou metodou jak lze vidět v následujícím kódu.

```

1 self.selection->startAnimation("Take 001", true);

```

Přřazení markeru

Za předpokladu správného nastavení *Metaia* konfiguračním *XML*, můžeme vytvořeným geometriím snadno přiřazovat identifikátor markeru na kterém se má zobrazovat.

```

1 item.object = [self loadGeometryFromPath:[GlobalConfig
2     pathToModelForZoneType:plot.plotZone.type

```

```

3         level:plot.plotZone.level]];
4
5 item.object->setCoordinateSystemID(plot.markerId.intValue);

```

V tomto případě je *item.object* geometrie typu *metaio::IGeometry* * a nastavíme jí coordinační systémový identifikátor na hodnotu podle daného plotu.

Jak vidíme z uvedených příkladů, *Metaio* nám velmi ulehčuje práci s modely. Jedinou věcí, na kterou si musíme při práci dát pozor, je si uvědomit, že pracujeme s C++ instancemi tříd. C++ narozdíl od Objective-C nedisponuje *ARC* (*Automatic reference counting*) a je proto nutné spravovat uvolňování instancí z paměti ručně.

6.3 Cocoapods

Pro usnadnění připojování pomocných frameworků do projektu vznikl open source projekt s názvem *Cocoapods*. Tento tzv. dependency manager, napsaný v jazyce *Ruby*, má za úkol přidávat, odebírat nebo aktualizovat použité externí frameworky. Seznam frameworků je definován textovým souborem s názvem Podfile, jsou v něm definice zdrojů, verzí a targetu projektu, do kterých mají být frameworky přilinkovány. Veškeré frameworky použité v této hře jsou vydány pod svobodnými licencemi (většinou *MIT*).

MagicalRecord

MagicalRecord je framework, který usnadňuje práci s klasickým *Core Data* frameworkem. Zjednodušuje práci s daty zaváděním *Active record* přístupu známého zejména z *Ruby on Rails*. Konkrétně se jedná o udržování defaultních kontextů a jejich vytváření a vytváření, update, mazání a import objektů. Tento framework je dostupný pod *MIT* licenci (Magical Panda Software, 2015).

TFTableDescriptor

TFTableDescriptor slouží jako pomocník při sestavování tabulek. Tabulky lze sestavit pomocí popisu speciálními třídami *TFTableDescriptor*, *TFSectionDescriptor* a *TFFRowDescriptor*. Tento framework je vyvíjen firmou *The Funtasty* pod otevřenou licenci *MIT* (The Funtasty, 2015).

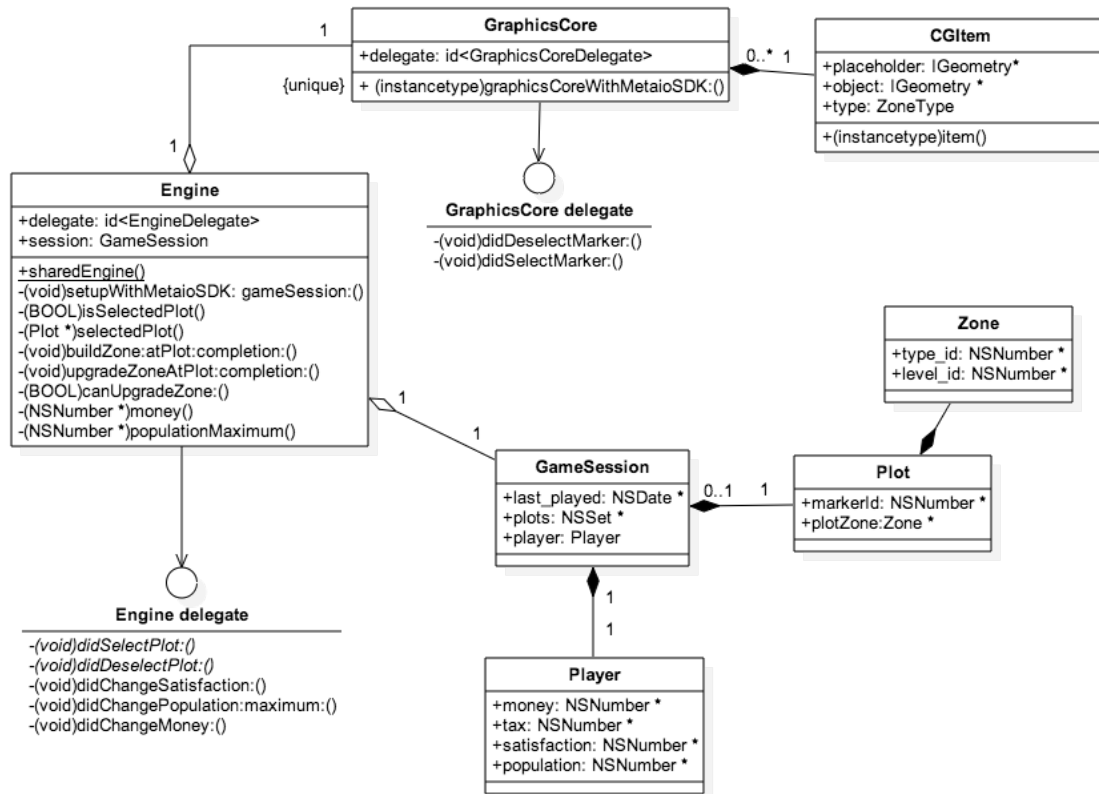
6.4 Model

Pro vývoj aplikací na platformu *iOS* se využívá takzvaného *MVC* (Model-View-Controller) přístupu. Základní model hry se skládá z těchto tříd:

- **Engine** - jádro hry, řídí celou herní logiku, obsluhuje ostatní komponenty modelu

- **GraphicsCore** - grafické jádro hry, zajišťuje vykreslování objektů na správných markerech, deleguje změny stavů, spouští animace a překládá koordináty
- **GraphicsItem** - představuje jeden vykreslovaný objekt v herní scéně, jejich vytváření je řízeno z GraphicsCore
- **MetaioSDK** - třída frameworku Metaio, instance této třídy je držena v GraphicsCore a je pomocí ní řízeno vykreslování objektů ve scéně
- **GameSession** - uchovává informace o stavu hry, obsahuje reference na hráče a herní plochu
- **User** - obsahuje informace o uživateli (peníze, spokojenost, ...)
- **Plot** - představuje herní parcelu, obsahuje informace o zóně a metody pro jejich vytváření a mazání
- **Zone** - jsou vytvářeny Ploty, obsahují informace na základě typu

Jak v modelu jednotlivé třídy figurují je navrženo jednoduchým UML diagramem tříd.



Obrázek 9: Diagram tříd datového modelu. Zdroj: Aleš Kocur

Tento návrh zajišťuje striktní oddělení herní logiky od grafického vykreslování. Pro uchování stavu hry stačí ukládat instance *GameSession* se svými kompozicemi. Uloženou *GameSession* lze pak přiřadit *Enginu*, který dokáže na jejím základě obnovit grafický obraz hry. Grafické jádro monitoruje změny v *GameSession* přiřazené v *Enginu* a na jejich základě provádí změny viditelné části hry.

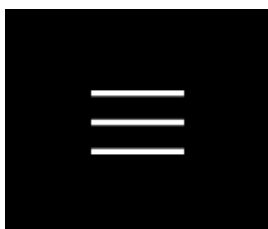
6.5 Uživatelské rozhraní

Rozhraní, jakým uživatelé – hráči – interagují s prvky hry, jsou kritickou částí návrhu. Důležitou roli zde hrají zaběhlé principy daného systému, řekněme konvence, na které jsou uživatelé zvyklí – setkávají se s nimi napříč všemi aplikacemi. V případě operačního systému iOS je jednou z hlavních domén uživatelského rozhraní navigační bar umístěný vždy nahoře po celé šířce displeje.



Obrázek 10: Navigation bar ze hry ARCity. Zdroj: Aleš Kocur

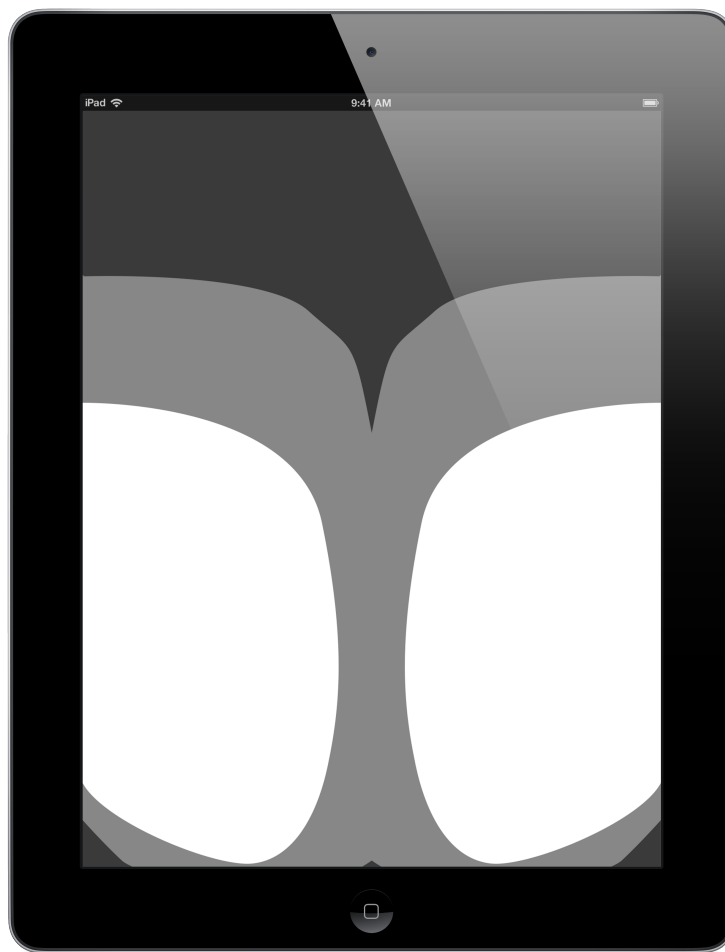
Z toho můžeme usoudit, že uživatel je zvyklý veškeré důležité navigační prvky a statická data (např. titulky) hledat právě na tomto místě, a koncipovat návrh tak, abychom všechna statická data a navigační prvky směřovali na horní část obrazovky. Jelikož je většinou takovýchto navigačních informací velké množství a zobrazování jej na každé obrazovce není z hlediska šetření místa pro samotný obsah příliš dobré, začalo se používat, a dnes je již konvencí, menu reprezentované ikonou se třemi vodorovnými čarami (známé také jako *hamburger menu*).



Obrázek 11: Hamburger menu ze hry ARCity. Zdroj: Aleš Kocur

Dalším důležitým aspektem návrhu rozhraní je rozměr samotného zařízení. Tato hra je primárně cílena na *iPad*. Máme tedy k dispozici plochu čítající 2048×1536 pixelů s úhlopříčkou 12,9 palce. Hlavním obsahem hry je okno pro zobrazování rozšířené reality, je tudíž nutné pro něj vymezit dostatečně velký prostor. Z daného rozlišení můžeme usoudit, že při natočení zařízení do tzv. *portrait* módu, tj. kratší stranou k sobě, výška displaye představuje oněch 2048 pixelů. Při využití části zhruba 500 pixelů na navigaci a ovládání získáme přibližně čtvercovou plochu pro

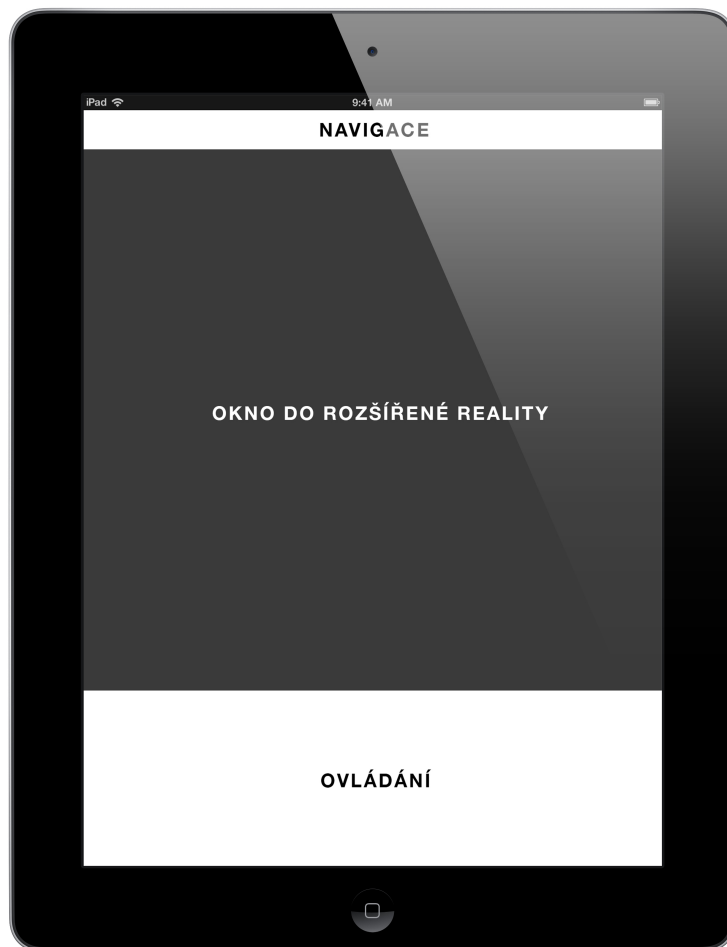
zobrazení okna rozšířené reality. Při takovém držení *iPadu*, oběma rukama zároveň, využívá hráč k ovládní své palce. Dosah a příjemnost takového ovládní ilustruje *Obrázek 12*.



Obrázek 12: Rozdělení dosažitelnosti prvků na iPadu 3. generace v portrait módu na základě vlastního měření. *Zdroj: Aleš Kocur*

Jak můžeme vidět, nejdosažitelnější části displeje se nachází v spodních částí displeje. Na základě takovéto úvahy dojdeme k závěru, že pro snadno dosažitelné ovládací prvky by se mělo ovládní hry vyskytovat někde ve spodní části displeje. Jelikož prostřední část máme vyhrazenou pro okno do rozšířené reality, jediné místo zbývá na spodním okraji displeje.

Složením výše uvedených věcí do jednoho, získáme ideální návrh rozhraní. Horní část, o přibližné velikosti 100 pixelů, vyčleníme pro navigaci a ukazatele, prostřední část, asi 1500 pixelů, pro okno do rozšířené reality a zbylých zhruba 400 px ve spodní části vymezíme na ovládání. Po návrhu konkrétních ovládacích prvků pak můžeme návrh optimalizovat ve prospěch ještě většího okna pro rozšířenou realitu.



Obrázek 13: Výsledný návrh rozložení prvků pro hru ARCity. Zdroj: Aleš Kocur

Z konceptu hry vyplývají dva hlavní ovládací prvky. První umožňující stavbu zón a druhý zobrazení informací o vybrané postavené zóně s možností zónu vylepšit. Z tohoto předpokladu lze vyvodit, za jakých okolností je potřeba toto ovládání zobrazovat. V prvním případě, když uživatel vybere prázdnou parcelu pro stavbu zóny, v druhém pak při výběru nějaké konkrétní zóny. V kterémkoliv jiném případě není ovládací částí potřeba a můžeme ji skrýt ve prospěch zvětšení plochy pro okno do rozšířené reality.

Design

Design je velmi subjektivní součástí návrhu rozhraní a jeho líbivost není nikdy zaručena, protože je hodnocen hodnotitelem na základě jeho vkusu. Nehledě na tuto skutečnost, měl by se design držet několika pravidel:

- Jednoduchost
- Intuitivnost
- Vypovídající o své funkci (tzv. self-explanatory)
- Ohled na různá postižení (např. poruchy barvocitu)

Na základě těchto pravidel byl navrhnout pro veškeré prvky vysoce kontrastní, černobílý design, postavený na typografii použitím bezpatkového písma Helvetica Neue navrženého Maxem Miedingerem v řezech Light a Bold, doplněný o piktogramy vyjadřující povahu jednotlivých zón a statistik.

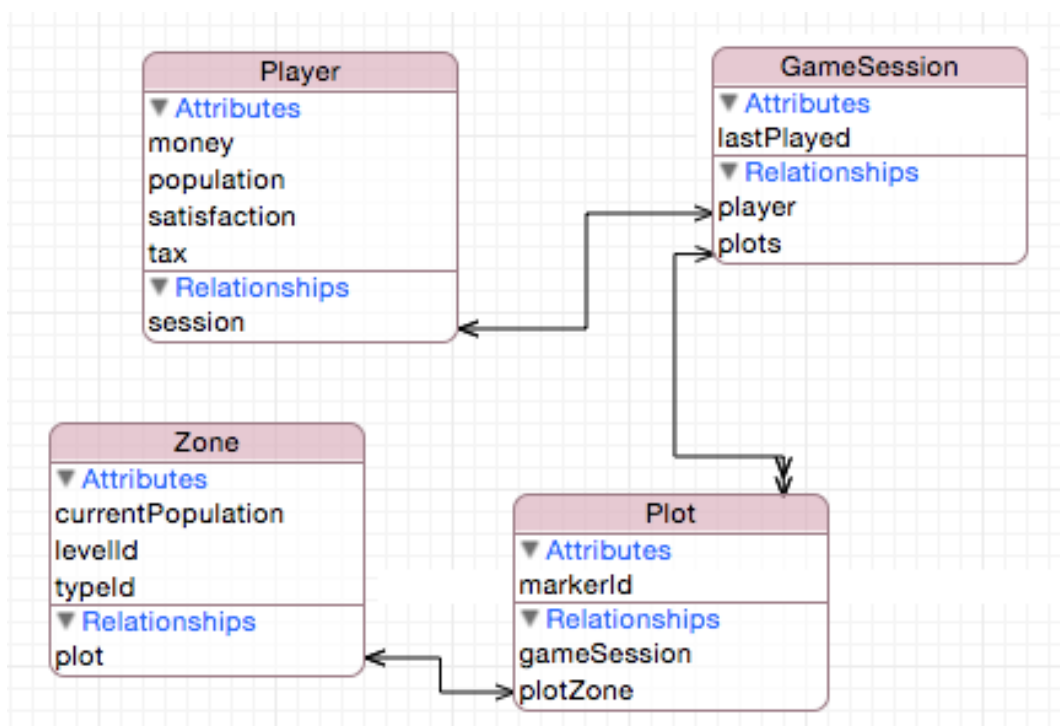


Obrázek 14: Ukázka z grafického návrhu aplikace. *Zdroj: Aleš Kocur*

Snímky obrazovky z hotové hry lze najít v druhé části přílohy.

6.6 Ukládání dat

Hra využívá pro ukládání dat zmíněný MagicalRecord framework, který obaluje systémovou knihovnu zvanou Core Data. Návrh datového modelu vychází z UML diagramu tříd ukládané části – GameSession, Plot, Zone a Player.



Obrázek 15: Datový model hry ARCity. Zdroj: Aleš Kocur

6.7 Výpočty ukazatelů

Hra pracuje s pěti typy ukazatelů stavu hry:

- Peníze
- Populace (aktuální/maximální)
- Spokojenost obyvatel
- Pracovní místa
- Daň

Ukazatel peněz stojí na jednoduchém principu přičítání a odečítání hodnot stanovených jako cena za stavbu/vylepšení/provoz. Stejně tak maximální hodnota populace je součtem postavených obytných zón, respektive jejich kapacit, které se odvíjí od jejich úrovně vylepšení. Změna těchto hodnot probíhá právě při zmíněných

úkonech – stavba, vylepšení, provoz budovy. Na stejném principu funguje i výpočet počtu pracovních míst.

Dalšími typy ukazatelů jsou aktuální stav populace a spokojenost. Tyto hodnoty jsou vypočítávány v pravidelných intervalech a jsou ovlivňovány více aspekty. Rychlost iterace těchto výpočtů stanovuje rychlost hry a na základě několika zkoušení byl jako ideální interval stanoven na 3 sekundy. Obecný princip výpočtu hodnot spočívá v nalezení nové hodnoty na základě parametrů příslušných danému ukazateli. Odečtením aktuálního stavu od této nové, očekávané hodnoty, získáme rozdíl mezi původním stavem a novým stavem, a ten je vynásoben nějakou malou konstantou pohybující se kolem 0,5 a přičten k původnímu stavu. Tímto nám, např. po stavbě zóny přidávající spokojenost, nevystoupají hodnoty skokově, nýbrž parabolicky během několika iterací.

Jako příklad uveďme počáteční stav spokojenosti 20 %. Po stavbě kulturní zóny v první iteraci vypočteme novou hodnotu – 60 %. Od nové hodnoty odečteme původní a získáme rozdíl:

$$0,6 - 0,2 = 0,4$$

Tento rozdíl je vynásoben číslem 0,5.

$$0,4 \times 0,5 = 0,2$$

Přičtením k původnímu stavu dostáváme nový stav po první iteraci – 40 %. Stejným způsobem v dalších iteracích dostaneme nové hodnoty 50 %, 55 %, 57 %, 58 %, 59 % a 60 %. Celkový čas do stabilizace ukazatele lze pak vypočítat podle vzorce:

$$t = h \times d \tag{1}$$

kde t je výsledný čas, h je počet hodnot, resp. iterací, a d je délka trvání iterace v sekundách. Dosazením dostaneme výsledek v podobě délky trvání stabilizace a to 21 sekund.

Konkrétní vzorec pro každý ukazatel je navrhnut na jeho logické návaznosti k ostatním ukazatelům a ovlivňují se tudíž navzájem. Po každé změně ve hře následuje stav postupné stabilizace. Po několika iteracích se ukazatele hry stabilizují na určitou hodnotu. Pravidla pro výpočty zbylých ukazatelů jsou definovány těmito vzorci.

Počet obyvatel

$$p = mp \times s \tag{2}$$

kde p je populace, mp je maximální možná populace a s je spokojenost.

Spokojenost

$$s = \frac{f(o, m) + f(o, sz) + g(dk, d)}{3} \quad (3)$$

kde

$$f(x, y) = \begin{cases} \frac{y}{x} & : x > y \\ 1, 0 & : jinak \end{cases} \quad (4)$$

$$g(x, y) = \begin{cases} 1, 0 & : \frac{y}{x} - 0,5 > 1,0 \\ 0 & : \frac{y}{x} - 0,5 < 0 \\ \frac{y}{x} - 0,5 & : jinak \end{cases} \quad (5)$$

a o je aktuální počet obyvatel, m je počet pracovních míst, sz je hodnota uspokojení z dosud postavených zón, dk je konstanta pro průměrnou daň a d je aktuální daň.

7 Testování uživatelského rozhraní

Jedním z pilířů použitelnosti je uživatelské testování. Mnoho skutečností, které programátor či designér aplikace považují za samozřejmé, nemusí být pro uživatele na první pohled patrné či srozumitelné. Steve Grug, autor knihy *Nenuťte uživatele přemýšlet!: praktický průvodce testováním a opravou chyb použitelnost webu*, přirovnává uživatelské testování k návštěvě cizinců v rodném městě. Prováděli je po svém městě, všimneme si věcí, které jsme doposud nevnímali, protože jsme je brali jako samozřejmost (KRUG, 2010).

7.1 Základní přístupy testování

Existují různé přístupy, jak lze uživatelské testování provádět. Liší se zejména v časové a technologické náročnosti. Vhodný typ testování je nutno zvolit na základě projektu, který má být testován.

Eye tracking

Jedná se o velmi populární metodu snímání uživatelského zacházení s produktem. K testování je potřeba člověka-testera, který bude daný produkt používat, a to nejlépe bez předchozí zkušenosti. Při testování se sledují pohyby očí v poměru k obrazovce a získáváme tím obraz uživatelského zájmu. Více porozované oblasti pak lze odlišit barvami a získáme pro jednotlivé obrazovky zóny největšího soustředění uživatelského zraku. Analýzou těchto dat pak můžeme zlepšit umístění důležitých ovládacích prvků do zón s vysokým indexem soustředění, případně se změnou barevnosti a kontrastu pokusit tyto zóny přemístit. Ke snímání soustředěnosti zraku uživatele-testera je možno použít speciálních brýlí či helem. Toto řešení je nicméně nákladnější na počáteční nákup hardwaru a softwaru a proto vznikly alternativy. Tou je například pro testování mobilních aplikací metoda využití přední kamery pro snímání místa pozorování. Uživatelé-testerů jsou snímáni přímo zařízením během procesu testování a pozorovací mapy jsou generovány přímo pro konkrétní obrazovky, na kterých se právě tester nachází. *Eye tracking* lze samozřejmě využít i k jiným věcem než k uživatelskému testování, důkazem může být například firma *uMoove* vyvíjející SDK pro interakci s aplikacemi pomocí eye trackingu (UMOOVE, 2015).

Heuristic evaluation

Heuristická analýza uživatelského přístupu spočívá v prozkoumání uživatelského rozhraní a kontrole nedostatků na základě zkušeností. Zkušenostmi máme na mysli seznam poznatků, které by neměly být při návrhu rozhraní opomenuty. Výhoda tohoto testování je, že není potřeba uživatele-testera, ale tento test může být vyhodnocen kýmkoliv kompetentním v daném projektu. Zkoumáním lze pak například zjistit, že v některých situacích uživatel ztrácí přehled o aktuálním stavu aplikace,

nebo nemá možnost návratu k přechozím stavům aplikace. Jedním ze zakladatelů tohoto přístupu je Jakob Nielsen, který vytvořil jakési desatero základních principů pro interaktivní design (překlad (LICHNOVSKA KARBEROVA, 2010); originál (NIELSEN, 1995)):

1. Viditelnost stavu systému
2. Spojení mezi systémem a reálným světem
3. Uživatelská kontrola a svoboda
4. Konzistence a standardizace
5. Prevence chyb
6. Rozpoznání místo vzpomínání
7. Flexibilní a efektivní použití
8. Estetický a minimalistický design
9. Pomoc uživatelů poznat, pochopit a vzpamatovat se z chyb
10. Náповěda a návody

Zamyšlením se nad těmito body v kontextu dané aplikace můžeme objevit její nedostatky.

Focus Groups

Hlavní podstatou tohoto testování je doplnění informací o použitelnosti, zejména pak uplatnění některých funkcí. Testování je vedeno formou diskuze ve skupině několika lidí, nejlépe různých postavení, zájmů a pohlaví. Diskuzi vede moderátor a jeho hlavním úkolem je udržovat či směřovat diskuzi k potřebným tématům. Úskalím této metody může být výskyt dominantního člena diskuzní skupiny, který následně strhává názory ostatních (NIELSEN, 1997).

Cognitive walkthrough

Česky tzv. kognitivní průchod, je způsob testování uživatelského rozhraní za pomoci testerů, kteří zkoumají složitost průchodu aplikací s daným účelem. Testerem zde může být jak člověk nezasvěcený, tak i přímo se podílející na vývoji aplikace. Jako příklad průchodu lze uvést registraci, přihlášení a také specifické průchody typu rezervace v rezervačním systému a podobně. Takto se stanoví průchody a cíle pro danou aplikaci. V průchodu je hodnocena jednoduchost (počet akcí uživatele nutných k průchodu), intuitivnost (srozumitelnost informací) a navigace (možnost návratu z nevyžádaných pozic). Nedostatek, který je nalezen při průchodu, je zaznamenán a označen prioritou, podle které je mu pak nutno věnovat, resp. nevěnovat pozornost.

Další metody a shrnutí

Metod testování uživatelského rozhraní je velmi mnoho. Za zmínku stojí dále také *Formal usability inspections*, *Pluralistics walkthroughs*, *Feature inspection*, *Consistency inspection* a *Standard inspection*, informace o nich lze nalézt v rešerši *Usability Inspection Methods* napsané roku 1994 *Jakobem Nielsenem* (NIELSEN, 1994).

Pro testování této aplikace jsem zvolil metodu heuristické analýzy a to hlavně z důvodu možnosti testování mnou samým.

7.2 Heuristická analýza ARCity

Na základě uvedených informací bylo sepsáno 10 bodů odpovídajících desateru základních principů popsanych v sekci *Základní přístupy testování - Metody Heuristic evaluation* a posouzeno v kontextu této aplikace.

Viditelnost stavu systému

Této viditelnosti je dosaženo statickým umístěním ukazatelů hry do horní části obrazovky. Uživatel má v každém daném stavu hry možnost zjistit v jakém stavu se jeho město nachází (počet obyvatel, stav peněz, spokojenost, počet pracovních míst).

Spojení mezi systémem a reálným světem

Veškeré texty jsou psány srozumitelně a výstižně. Některé důležité prvky doplňují pikrogramy pro srozumitelnost na první pohled. Nevyskytují se zde žádné odborné termíny.

Uživatelská kontrola a svoboda

Umístěním tlačítka pro menu v horním levém rohu má uživatel možnost kdykoliv hru pozastavit. Stejně tak při minimalizaci je aplikace automaticky ukládána a její případná terminace nemá následek na poslední stav hry. Možnost odvolání akcí při stavbě města není z očividných důvodů žádoucí.

Konzistence a standardizace

Hra se drží standardních termínů pro zahájení hry (*angl. New game*) či pokračování (*angl. Continue*). Stejně tak zavádí názvy jednotlivých zón a tyto názvy jsou použity napříč celou aplikací při jejich zmínce.

Prevence chyb

Design aplikace dovoluje uživateli provádět pouze platné tahy. Nemůže například nastat situace, kdy při označení již postavené zóny by se uživateli zobrazilo menu

pro stavbu zóny nové, nebo opačně při označení prázdné parcely zobrazení menu s detaily a možností vylepšení.

Rozpoznání místo vzpomínání

Zobrazení menu s akcemi přesně ve chvíli, kdy uživatel označí položku, je pro uživatele snadno rozpoznatelným impulsem, očekávajícím jeho reakci ve smyslu volby. Naopak jeho skrytím, kdy volba není potřeba, je upřednostněno místo pro herní plochu a tím vybízeno k provedení akce na herní desce.

Flexibilní a efektivní použití

Náročnější uživatelé mají možnost jednoduše rozšířit herní plochu o další volné parcely přidáním až dvou dalších herních ploch.

Estetický a minimalistický design

Design je velmi minimalistický, kontrastní – využívá pouze kombinace černé a bílé barvy – a doplněn piktogramy, korespondující s černobílým designem. Snaží se vyjádřit přesnou podstatu každého užitého prvku.

Pomoc uživatelů poznat, pochopit a vzpamatovat se z chyb

Rozsah chyb je natolik eliminován designem aplikace, že prostor zbývá pouze pro neidentifikovatelné chyby a kritické chyby kódu. V prvním případě se jedná o chybu, kdy nejsou načteny textury objektů. Tento stav je bohužel chybovým stavem použitého SDK a to nenabízí žádnou možnost jej ošetřit. V druhém případě se pak chyba projeví pádem aplikace. Všechny pády aplikací vypadají na iOS platformě stejně a uživateli je jasné, že je nutné aplikaci pustit znovu. Těmto chybám lze předejít pouze testováním kódu aplikace. Tento bod lze označit jako bod s nízkou prioritou pro vyřešení.

Nápověda a návody

V menu má člověk přístupem na položku *About and help* možnost nechat si zobrazit základní informace o aplikaci, herní logice a jak s aplikací pracovat.

7.3 Zhodnocení

Na základě heuristické analýzy jsme došli k poznatku, že aplikace vyhovuje všem zmíněným bodům analýzy, pouze u předposledního bodu, týkajícího se chybových stavů, jsme zaznamenali problém s texturami. Ten má však malou prioritu hlavně z důvodu velmi malé četnosti výskytu a také v podstatě nemožnosti opravy tohoto problému.

Pro důkladnější hodnocení bychom mohli provést také *Focus group* metodu pro vylepšení hratelnosti a využít názoru samotných hráčů. Správným řízením směřování diskuze bychom také mohli získat cenné nápady na budoucí vylepšování hry, případné přidávání nových zón a ukazatelů či rozšířit možnosti rozšířené reality o nové funkce.

8 Závěr

8.1 Zhodnocení dosažení cílů

Vody her s rozšířenou realitou jsou stále ještě neprobádané a v posledních letech přibýlo konceptů i vydaných her. Většina jich však stále ztroskotává na náročnosti přípravy k hraní z pohledu samotného hráče – získání markerů nebo podkladů, potřeba webkamery nebo speciálních zobrazovacích zařízení či složité ovládání. Na základě poznatků o aktuálních hrách byla vytvořena hra s cílem eliminovat některé tyto nedostatky, aby byla dostupnější pro běžné hráče.

Jako zařízení byl zvolen *iPad*, odpadá tedy nutnost speciálního zařízení. Hra má jednoduchou, kontrastní, *self-explanatory* grafiku. Jedinou predispozicí hry je nutnost tisku herní desky. Byl splněn cíl eliminovat co nejvíce nedostatků pozorovaných v podobných hrách.

Ze získaných informací a jejich porovnání bylo zjištěno, že při volbě frameworku pro tvorbu hry s využitím rozšířené reality je k dispozici několik variant. Ne všechny jsou ale dobře zpracovány nebo poskytují jenom základní obecnou funkcionalitu. Na základě stanovených kritérií byl vybrán za nevhodnější framework *Metaio*. Kritéria a důvody, které k výběru vedly, jsou shrnuty v kapitole *Frameworky*.

Dále byly porovnány technologie, které lze použít pro tvorbu hry, jako je výběr formátu 3D modelů, modelovací nástroj (sekce *Metodika - Formát 3D modelů* a dál), a také rozebrány některé pomocné nástroje (*ARCity - Cocoapods*).

Na posledních stranách této práce si lze prohlédnout ukázky práce a následné uživatelské testování, ze kterého vyplývá úspěch implementace v kontextu uživatelské přívětivosti, zkoumané heuristickou analýzou.

8.2 Návrhy na vylepšení

Hratelnost vytvořené hry lze ještě zvýšit použitím *markerless trackingu* nebo metodami sledování neznámých prostředí (např. *SLAM*) eliminovat nutnost tisku herního podkladu úplně. Tyto metody jsou však implementačně výrazně náročnější nebo frameworky poskytující takovouto funkcionalitu nejsou zdarma. Hra by však byla hratelná na jakékoliv ploše a stala by se tak uživatelsky více přívětivou.

Při brainstormingu s mými kolegy z firmy *The Funtasty* padlo mnoho dalších nápadů na vylepšení. Jedna z těchto myšlenek mě velmi zaujala, myšlenka multiplayeru. Hráči by hráli hru každý na svém zařízení, které by se na pozadí synchronizovaly. Zajímavé je i pojetí multiplayeru, kooperační mód, ve kterém spoluhráči budují jedno obrovské město, či soutěžní mód, ve kterém se předhánějí na jedné ploše kohož město bude největší. Takovýto multiplayer by však byl hůře řešitelný se zmíněným bezmarkerovým řešením. Pro snadnější implementaci by bylo vhodné mít marker alespoň jeden, který by sloužil jako opěrný synchronizační bod pro hrající zařízení.

Herních vylepšení samotné hry se nabízí také mnoho, třeba více úrovní nebo více zón. Troufám si však říct, že nejlepší podněty k vylepšení by vzešly od samotných hráčů.

9 Přílohy

9.1 Datové přílohy

Na přiloženém CD jsou k dispozici tyto přílohy:

- arcity_workspace - kompletní zdrojové soubory k vytvořené hře
- source - veškeré vlastní podklady a zdrojové texty využité při tvorbě této práce

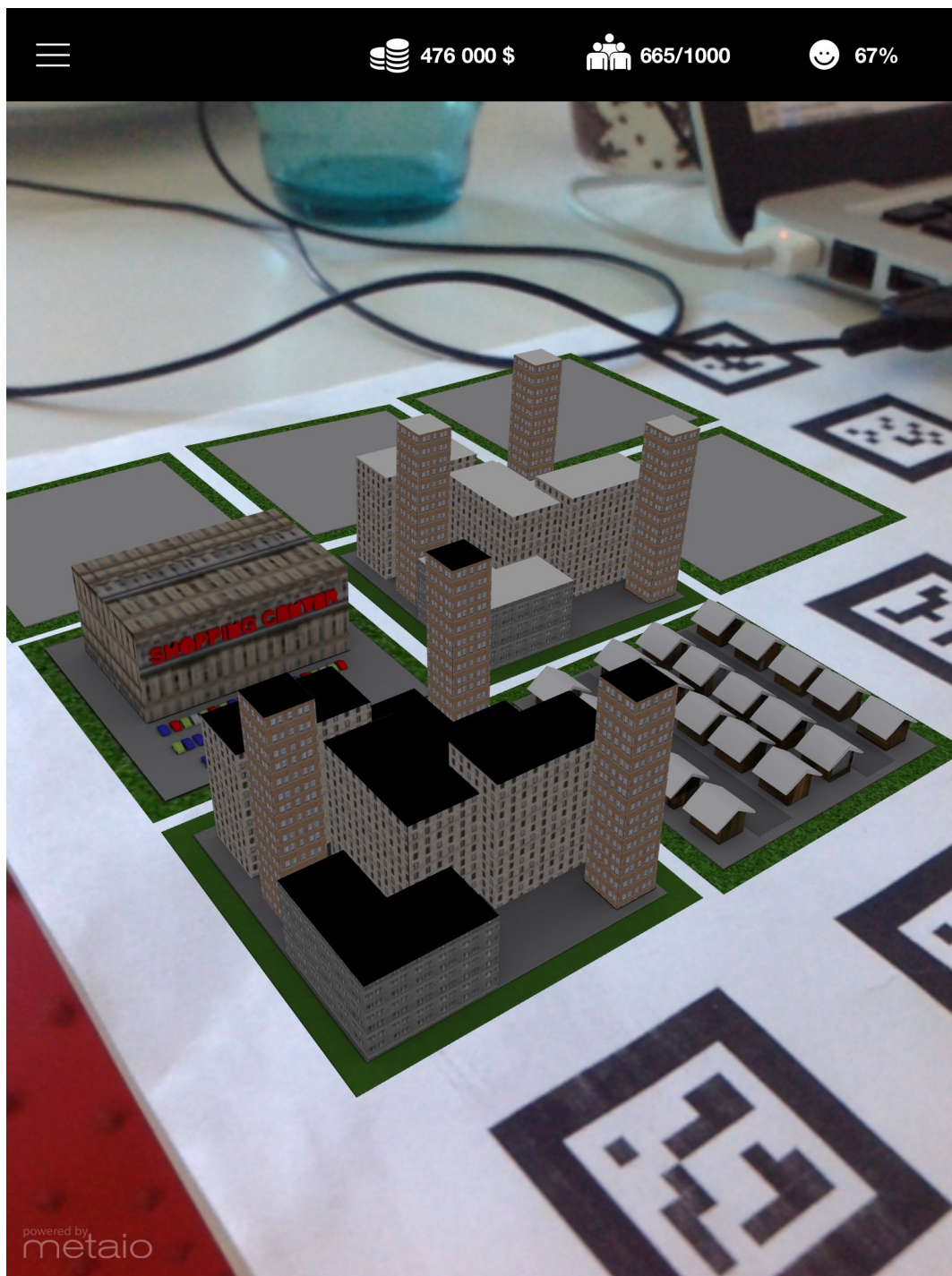
9.2 Ukázky z práce



Obrázek 16: Screenshot ze hry ARCity. Zdroj: Aleš Kocur



Obrázek 17: Hlavní menu ze hry ARCity. Zdroj: Aleš Kocur



Obrázek 18: Screenshot ze hry ARCity. Zdroj: Aleš Kocur

10 Reference

- L. FRANK BAUM *The Master Key: An Electrical Fairy Tale, Founded Upon the Mysteries of Electricity and the Optimism of Its Devotees* BiblioBazaar, 2006, ISBN 978-1426409240 .
- KLEIN, GEORG. *Visual Tracking Methods for Augmented Reality*. [online]. [cit. 2015-03-31]. Dostupné z: <http://www.raeng.org.uk/publications/other/georg-klein-presentation-frontiers-of-engineering> .
- PEOLEO *Drakerz-Confrontation About* [online]. [cit. 2015-03-31]. Dostupné z: <http://www.drakerz.com/qu-est-ce-que-drakerz.html/> .
- VENTURE BEAT *Drakerz-Confrontation augmented-reality card game launches in the U.S.* [online]. [cit. 2015-03-31]. Dostupné z: <http://venturebeat.com/2014/07/01/drazkerz-confrontation-augmented-reality-card-game-launches-in-the-u-s/> .
- NIATICS LABS *Ingress official website* [online]. [cit. 2015-04-01]. Dostupné z: <https://www.ingress.com/> .
- WIKIPEDIA *Wikipedia* [online]. [cit. 2015-04-02]. Dostupné z: http://en.wikipedia.org/wiki/Wavefront_.obj_file .
- PAUL BURKE *Object Files (.obj)* [online]. [cit. 2015-04-02]. Dostupné z: <http://paulbourke.net/dataformats/obj/> .
- M.H. WILLIAMS *id Software Wants To Shorten Dev Cycles* [online]. [cit. 2015-04-02]. Dostupné z: <http://web.archive.org/web/20110827005123/http://www.industrygamers.com/news/id-software-wants-to-shorten-dev-cycles/> .
- WIKIPEDIA *Wikipedia* [online]. [cit. 2015-04-02]. Dostupné z: http://en.wikipedia.org/wiki/MD2_%28file_format%29 .
- WIKIPEDIA *Wikipedia* [online]. [cit. 2015-04-02]. Dostupné z: http://en.wikipedia.org/wiki/Id_Tech_3 .
- QTIP *QTIP* [online]. [cit. 2015-04-02]. Dostupné z: <http://www.qtipplugin.com/> .
- AUTODESK *FBX* [online]. [cit. 2015-04-02]. Dostupné z: <http://www.autodesk.com/products/fbx/overview> .
- AUTODESK *FBX SDK* [online]. [cit. 2015-04-02]. Dostupné z: <http://usa.autodesk.com/adsk/servlet/pc/item?siteID=123112&id=10775847> .
- THE HUMAN INTERFACE TECHNOLOGY LAB, UNIVERSITY OF WASHINGTON *Feature list* [online]. [cit. 2015-04-02]. Dostupné z: <http://www.hitl.washington.edu/artoolkit/documentation/features.htm> .

- THE HUMAN INTERFACE TECHNOLOGY LABORATORY NEW ZEALAND *OSGART website* [online]. [cit. 2015-04-02]. Dostupné z: <https://www.artoolworks.com/community/osgart/> .
- DANIEL WAGNER AND DIETER SCHMALSTIEG *ARToolKitPlus for Pose Tracking on Mobile Devices* In Proc. 12th Computer Vision Winter Workshop (CVWW'07), Sankt Lambrecht, Austria, February 2007 .
- QUALCOMM *3D File Formats* [online]. [cit. 2015-04-02]. Dostupné z: <https://developer.vuforia.com/library/articles/Solution/3D-File-Formats> .
- BORKO FURHT *Markerless Tracking for Augmented Reality: Feature Matching: Image Patches. Handbook of augmented reality* New York: Springer, c2011, 259–260. ISBN 978-1-4614-0063-9. .
- THE GUARDIAN *Samsung creates drone, robotics and virtual reality lab* [online]. [cit. 2015-04-12]. Dostupné z: <http://www.theguardian.com/technology/2015/feb/10/samsung-independent-drone-robotics-virtual-reality-lab> .
- GSMA MOBILE ECONOMY *The Mobile Economy 2015* [online]. [cit. 2015-04-12]. Dostupné z: http://www.gsmamobileeconomy.com/GSMA_Global_Mobile_Economy_Report_2015.pdf .
- MAGICAL PANDA SOFTWARE *MagicalRecord* [online]. [cit. 2015-04-12]. Dostupné z: <https://github.com/magicalpanda/MagicalRecord> .
- THE FUNTASTY *TFTableDescriptor* [online]. [cit. 2015-04-12]. Dostupné z: <https://github.com/thefuntasty/TFTableDescriptor> .
- KRUG, STEVE *Nenuřte uživatelé přemýšlet!: praktický průvodce testováním a opravou chyb použitelnost webu*. Vyd. 1. Brno: Computer Press, 2010, 165 s. ISBN 978-80-251-2923-4. .
- UMOOVE INC. *uMoove website* [online]. [cit. 2015-05-11]. <http://umoove.me/> .
- PAVLA LICHNOVSKÁ A EVA KARBEROVÁ, FILOZOFICKÁ FAKULTA MASARYKOVY UNIVERZITY KABINET INFORMAČNÍCH STUDIÍ A KNIHOVNICTVÍ *Testování a hodnocení rozhraní* [online]. [cit. 2015-05-11]. <http://human-computer-interaction.webnode.cz/testovani-a-hodnoceni-rozhrani/metody-testovani/heuristicka-analyza/> .
- JAKOB NIELSEN *10 Usability Heuristics for User Interface Design* [online]. [cit. 2015-05-11]. <http://www.nngroup.com/articles/ten-usability-heuristics/> .
- JAKOB NIELSEN *The Use and Misuse of Focus Groups* [online]. [cit. 2015-05-11]. <http://www.nngroup.com/articles/focus-groups/> .

- JAKOB NIELSEN *Usability Inspection Methods* [online]. [cit. 2015-05-11]. [http://www.idemployee.id.tue.nl/g.w.m.rauterberg/lecturenotes/0H420/Nielsen\[1994\].pdf](http://www.idemployee.id.tue.nl/g.w.m.rauterberg/lecturenotes/0H420/Nielsen[1994].pdf) .
- YAHOO INC. *Flickr* [online]. [cit. 2015-04-02]. Flickr Obrázek ve formátu: JPEG. Dostupné z: https://farm9.staticflickr.com/8219/8266348091_08fe415355_o.jpg .
- GAME GUIDE *Game guide* [online]. [cit. 2015-04-02]. Game guide Obrázek ve formátu: JPEG. Dostupné z: <http://game-guide.fr/wp-content/uploads/2014/03/drakerz-confrontation-05.jpg> .
- ENGADGET *Engadget* [online]. [cit. 2015-04-02]. Engadget Obrázek ve formátu: JPEG. Dostupné z: <http://www.blogcdn.com/www.engadget.com/media/2009/06/arhrrrr-picrm-eng.jpg> .
- VTTRESEARCH *Marker tracking* [online]. [cit. 2015-04-02]. VTTResearch Obrázek ve formátu: JPEG. Dostupné z: <http://virtual.vtt.fi/virtual/proj2/multimedia/media/images/MultiMarker.jpg> .
- WIKIPEDIA *Wikipedia* [online]. [cit. 2015-04-02]. Wikipedia.org Obrázek ve formátu: JPEG. Dostupné z: http://upload.wikimedia.org/wikipedia/commons/f/f4/App_iSkull,_an_augmented_human_skull.jpg .
- DEPARTMENT OF ENGINEERING SCIENCE, UNIVERSITY OF OXFORD *SLAM* [online]. [cit. 2015-04-02]. Obrázek ve formátu: JPEG. Dostupné z: http://www.robots.ox.ac.uk/lav/Papers/-castle_etal_iswc2008/castle_etal_iswc2008.jpg .
- THE HUMAN INTERFACE TECHNOLOGY LAB, UNIVERSITY OF WASHINGTON *ARToolkit Benchmarks* [online]. [cit. 2015-04-03]. Obrázek ve formátu: JPEG. Dostupné z: <http://www.hitl.washington.edu/artoolkit/documentation/benchmark.htm> .
- METAIO GMBH *Metaio iOS polycount* [online]. [cit. 2015-04-03]. Obrázek ve formátu: JPEG. Dostupné z: <http://dev.metaio.com/sdk/documentation/content-creation/3d-animation/polygon-count/general-guidelines/> .
- DAVIDE 'FOLLETTO' CASALI *Hamburger Icon / Lexicon Nugget 2* [online]. [cit. 2015-04-28]. Obrázek ve formátu: JPEG. Dostupné z: <https://dribbble.com/shots/880056-Hamburger-Icon-Lexicon-Nugget-2> .