

# **Manipulace s 3D modely v mobilních aplikacích**

Manipulations with 3D Models in Mobile Applications

Bc. Iva Dorazínová

---

Diplomová práce  
2014



Univerzita Tomáše Bati ve Zlíně  
Fakulta aplikované informatiky

---

Univerzita Tomáše Bati ve Zlíně

Fakulta aplikované informatiky

akademický rok: 2013/2014

## ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: Bc. Iva Dorazínová

Osobní číslo: A12415

Studijní program: N3902 Inženýrská informatika

Studijní obor: Počítačové a komunikační systémy

Forma studia: prezenční

Téma práce: Manipulace s 3D modely v mobilních aplikacích

Téma anglicky: Manipulations with 3D Models in Mobile Applications

Zásady pro vypracování:

1. Prostudujte vývojové frameworky pro manipulaci s 3D objekty v prostředí mobilních aplikací.
2. Otestujte a srovnajte dostupné frameworky umožňující manipulaci s 3D modely a jejich import.
3. Na základě srovnání vyberte vhodného kandidáta pro implementaci do mobilní aplikace na platformě Android.
4. Vytvořte ukázkovou mobilní aplikaci, která bude demonstrovat práci s 3D modely.



Rozsah diplomové práce:

Rozsah příloh:

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

Seznam odborné literatury:

1. DIMARZIO, J. Programujeme hry pro Android 4. Brno: Computer Press, 2012. ISBN 978-80-251-3754-3.
2. VÁVRŮ, Jiří a Miroslav UJBÁNYAI. Programujeme pro Android. 2., rozš. vyd. Praha: Grada, 2013. Průvodce (Grada). ISBN 978-80-247-4863-4.
3. HERODEK, Martin. Android jednoduše. 1. vyd. Brno: Computer Press, 128 s. ISBN 978-80-251-4118-2.
4. ALLEN, Grant. Android 4: průvodce programováním mobilních aplikací. 1. vyd. Překlad Jakub Mužík. Brno: Computer Press, 2013. ISBN 978-80-251-3782-6.
5. POKORNÝ, Pavel. Blender: naučte se 3D grafiku. 2., aktualiz. a rozš. vyd. Praha: BEN – technická literatura, 2009. ISBN 978-80-7300-244-2.
6. Pixel: 2D/3D grafika a animace, DTP, digitální foto, video a zvuk/. Praha: Atlantida Publishing,. ISSN 1211-5401.
7. KOMATINENI, Satya, Dave MACLEAN a Eric FRANCHOMME. Pro Android 4. New York: Distributed to the book trade worldwide by Springer Science Business Media, xxv, 991 p. ISBN 978-143-0239-307.

Vedoucí diplomové práce:

**Ing. Radek Vala**

Ústav informatiky a umělé inteligence

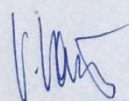
Datum zadání diplomové práce:

**7. února 2014**

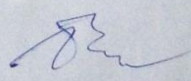
Termín odevzdání diplomové práce:

**27. května 2014**

Ve Zlíně dne 7. února 2014

  
prof. Ing. Vladimír Vašek, CSc.  
*děkan*



  
prof. Ing. Karel Vlček, CSc.  
*ředitel ústavu*

## **ABSTRAKT**

Cílem této práce je prostudovat dostupné frameworky pro manipulaci s 3D modely v prostředí mobilních aplikací. Dále tyto frameworky otestovat a srovnat jejich manipulaci a import objektů. Na základě srovnání vybrat vhodného kandidáta pro implementaci do mobilní aplikace na platformě Android. Výsledkem této práce je ukázková aplikace, která demonstruje práci s 3D modely.

Klíčová slova: Java, 3D model, Android, framework, Open-Source, Blender, Min3D, Rajawali, jPCT-AE, OpenGL

## **ABSTRACT**

The aim of this diploma thesis is to study the available frameworks for manipulating 3D models in an environment of mobile applications. Furthermore, these frameworks to test and compare the model manipulation and import. Based on comparison to choose a suitable candidate for implementation in mobile applications on the Android platform. The result of this diploma thesis is a sample application that demonstrates how to work with 3D models.

Keywords : Java, 3D model, Android, Framework, Open-Source, Blender, Min3D, Rajawali, jPCT-AE, OpenGL

Ráda bych poděkovala svému vedoucímu práce, panu Ing. Radku Valovi, za jeho věcné připomínky, ochotu, vstřícnost a cenné rady.

Také bych chtěla poděkovat své rodině a příteli za podporu.

*„Non schoale sed vitae discimus. Neučíme se pro školu, ale pro život.“*

*Seneca*

**Prohlašuji, že**

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

**Prohlašuji,**

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

Ve Zlíně

.....  
podpis diplomanta

**OBSAH**

<b>ÚVOD.....</b>	<b>9</b>
<b>I TEORETICKÁ ČÁST.....</b>	<b>10</b>
<b>1 ECLIPSE.....</b>	<b>11</b>
1.1 HISTORIE.....	11
1.2 ANDROID SDK.....	11
<b>2 ANDROID.....</b>	<b>14</b>
2.1 PLATFORMA ANDROID .....	14
2.2 ANDROID ARCHITEKTURA .....	14
2.3 JEDNOTLIVÉ VERZE .....	16
<b>3 ŽIVOTNÍ CYKLUS AKTIVITY.....</b>	<b>19</b>
<b>4 GENYMOTION .....</b>	<b>20</b>
<b>5 GRAFICKÉ NÁSTROJE .....</b>	<b>21</b>
5.1 JAK SE TVOŘÍ 3D MODEL .....	21
5.2 KOMERČNÍ 3D PROGRAMY .....	22
5.2.1 Autodesk 3ds Max.....	22
5.2.2 Autodesk Maya .....	23
5.2.3 Autodesk Softimage .....	23
5.2.4 Zbrush .....	23
5.2.5 Poser.....	23
5.2.6 Lightwave.....	24
5.2.7 Cinema 4D .....	24
5.3 NEKOMERČNÍ 3D PROGRAMY .....	24
5.3.1 Blender .....	25
5.3.2 Google SketchUp .....	25
5.4 ONLINE 3D APLIKACE .....	26
<b>6 VÝVOJOVÉ FRAMEWORKY.....</b>	<b>28</b>
6.1 MIN3D .....	28
6.1.1 Klonování animovaných 3D objektů .....	28
6.1.2 Použití akcelometru.....	29
6.1.3 Načtení souboru *.obj .....	29
6.2 RAJAWALI .....	31
6.3 JPCT-AE.....	31
6.4 OSTATNÍ FRAMEWORKY .....	33
6.4.1 Dwarf FW.....	33
6.4.2 libGDX.....	33
<b>II PRAKTICKÁ ČÁST .....</b>	<b>35</b>
<b>7 SROVNÁNÍ DOSTUPNÝCH FRAMEWORKŮ .....</b>	<b>36</b>

7.1	TEST Č. 1 - 3D MODEL OPICE .....	37
7.2	TEST Č. 2 – 3D MODEL DRAKU.....	39
7.3	TEST Č. 3 – 3D MODEL AUTA .....	41
7.4	TEST Č. 4 – 3D MODEL TYRANNOSAURUS REX .....	43
7.5	VÝSLEDNÉ SROVNÁNÍ .....	44
<b>8</b>	<b>3D MODEL VIEWER .....</b>	<b>45</b>
8.1	MAINACTIVITY.JAVA .....	45
8.1.1	activity_main.xml.....	47
8.1.2	file_dialog_main.xml, file_dialog_row.xml .....	48
8.1.3	CustomModel.java .....	48
8.1.4	DataHolder.java .....	49
8.1.5	GridViewCustomAdapter.java.....	49
8.1.6	FileDialog.java .....	49
8.2	RENDERACTIVITY.JAVA .....	49
8.2.1	menu.xml.....	51
8.2.2	model_miniaature.xml .....	51
8.2.3	ColorPickerDialog.java .....	52
8.2.4	Výsledná aplikace .....	52
	<b>ZÁVĚR .....</b>	<b>54</b>
	<b>ZÁVĚR V ANGLIČTINĚ.....</b>	<b>55</b>
	<b>SEZNAM POUŽITÉ LITERATURY.....</b>	<b>56</b>
	<b>SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK.....</b>	<b>61</b>
	<b>SEZNAM OBRÁZKŮ .....</b>	<b>62</b>
	<b>SEZNAM TABULEK.....</b>	<b>63</b>
	<b>SEZNAM PŘÍLOH.....</b>	<b>64</b>



## ÚVOD

3D grafika má ve světě moderních technologií své významné místo. Čím dál více objektů je pro lepší názornost široké veřejnosti vizualizováno a tak není divu, že se tato technologie dostává i do mobilních zařízení.

Implementace 3D modelů do mobilních telefonů a tabletů má velký potenciál. 3D modelů se může využít např. při propagaci, reklamě nebo při prosté vizualizaci nějakého objektu. Příkladem využití může být například nově vybudované nákupní centrum, které si pro své zvědavé návštěvníky vytvoří 3D aplikaci objektu nákupní galerie. Uživatel si může objekt otočit, přiblížit, popř. se podívat dovnitř areálu. Nemusí jít jen o nákupní středisko, ale i třeba o nově vybudovanou posilovnu, kavárnu, zámek či jiný objekt. Velmi často se vizualizují také domy a bytové jednotky. Dalším využitím může být například propagace výrobku (prakticky jakéhokoliv na internetu). Vhodným příkladem může být mobilní aplikace nabízející zákaznickou vizualizaci a konfigurator automobilu (např. 3D konfigurator zn. ŠKODA). Uživatel si může vybrat jednoduchým poklikem model auta, výbavu, motor, barvu auta a kol, typ kol, interiér a sedačky, mimořádnou výbavu, pojištění a tento navolený vůz si znovu prohlédnou, vytisknout a poslat emailem. Samozřejmostí je ceník, který je uváděn u každé položky zvlášť.

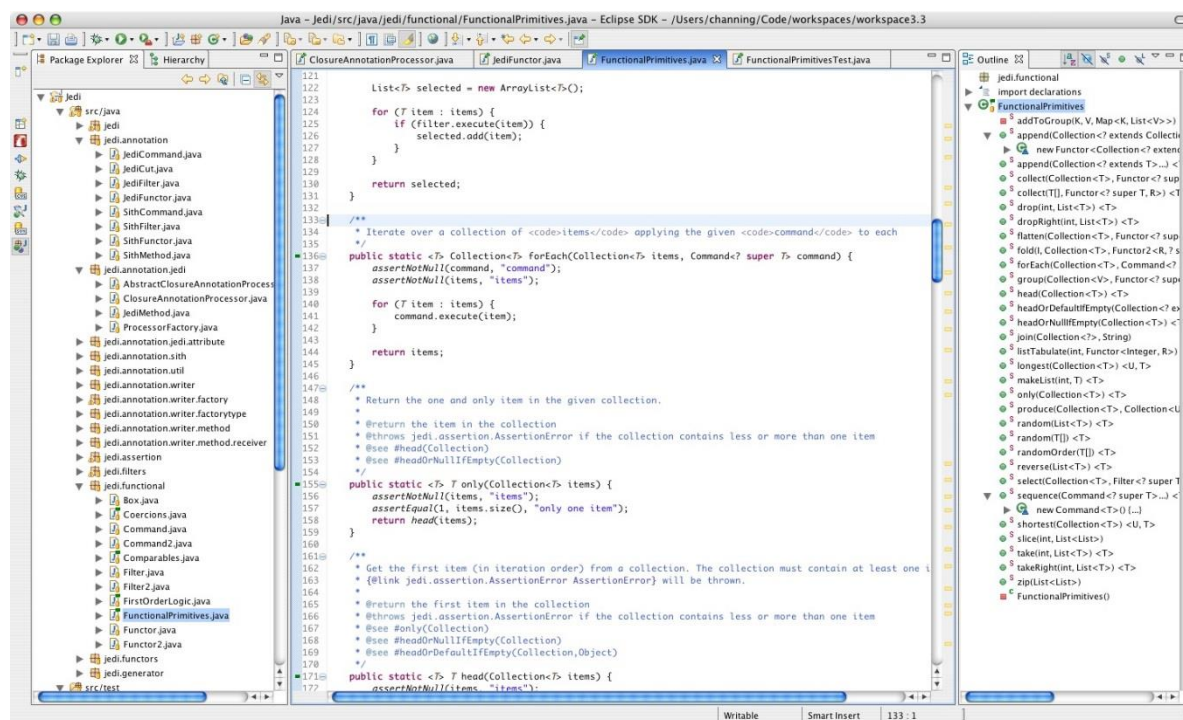
Práce Manipulace s 3D objekty v mobilních aplikacích představuje způsob, jakým lze importovat 3D modely do mobilních aplikací. Výsledná aplikace pak umožní zobrazení, otáčení, přibližování, oddálení a krátkou animaci 3D objektu na mobilním zařízení. Je zde uveden i výčet programů, ve kterých je možné takový 3D model vytvořit. Následně jsou popsány dostupné frameworky, které s nimi umožňují manipulaci a z těchto frameworků jsou vybrány 3, které jsou nejpřijatelnější, a z nich je vybrán ten, který v testu dopadl nejlépe.

## **I. TEORETICKÁ ČÁST**

## 1 ECLIPSE

Eclipse je multiplatformní vývojové prostředí, ze zkratky *Integrated Development Environment (IDE)*, kde se veškerý kód se píše v tomto prostředí.

Android vyžaduje Eclipse IDE a doporučuje se Eclipse Kepler 4.3.x pro Java EE. [28]



Obrázek 1. Ukázka vývojového prostředí Eclipse [29]

### 1.1 Historie

Vývojové prostředí Eclipse vzniklo v roce 2001. Počáteční Eclipse.org tvořilo IBM, Red Hat, SuSE, MERANT, QNX, Borland, Software Systems, Rational Software, TogetherSoft a Webgain. Koncem roku 2003 se Eclipse rozrostl o více než 80 členů. [25]

### 1.2 Android SDK

Vývojové nástroje pro OS Android je kolekce souborů a doplňků, které pracují ruku v ruce s Eclipse IDE k vytváření specifického vývojového nástroje pro Android. [28]

Android SDK se skládá z modulových balíčků, které je možné si stáhnout pomocí správce Android SDK (Android SDK Manager). Například, když jsou aktualizovány nástroje SDK nebo vyjde nová verze Androidu, je možné si rychle stáhnout nové nástroje pomocí SDK správce do svého prostředí. [30]

Zde v níže uvedené tabulce jsou dostupné balíčky pro Android SDK. Tabulka popisuje většinu dostupných balíčků, a kde se nacházejí. [30]

Tabulka 1. Dostupné balíčky pro Android SDK [30]

<i>Balíček</i>	<i>Popis</i>	<i>Umístění</i>
<i>SDK Tools</i>	Nástroje pro ladění a testování	<sdk>/tools/
<i>SDK Platform - tools</i>	Obsahuje platformě závislé nástroje pro vývoj a ladění aplikace	<sdk>/platform-tools/
<i>Documentation</i>	Nejnovější dokumentace pro Android APIs	<sdk>/docs/
<i>SDK Platform</i>	Obsahuje android.jar s kompletní Android knihovnou	<sdk>/platforms/<android-version>/
<i>System Images</i>	Obrazy systému (ARM a x86)	<sdk>/platforms/<android-version>/
<i>Sources for Android SDK</i>	Kopie zdrojového kódu, který je vhodný pro krokování kódu při ladění	<sdk>/sources/
<i>Samples for SDK</i>	Sbírka vzorových aplikací, které demonstrují různé platformy API	<sdk>/platforms/<android-version>/samples/
<i>Google APIs</i>	SDK add-on – je možné testovat aplikace	<sdk>/add-ons/
<i>Android Support</i>	Podpora Androidu	<sdk>/extras/android/support/
<i>Google Play Billing</i>	Obsahuje statické knihovny a příklady, které umožňují integrovat vyúčtování služeb v aplikaci s GP	<sdk>/extras/google/

<i>Google Play Licensing</i>	Obsahuje statické knihovny a příklady, které umožní provádět ověření licence pro app při distribuci GP	<code>&lt;sdk&gt;/extras/google/</code>
------------------------------	--	---

## 2 ANDROID

Operační systém Android, je v současné době nejrozšířenější mobilní platformou, kterou využívá stovky milionů uživatelů dnes již ve 190 zemích světa. [31,32]

Faktem zůstává, že Android není populární jen ve světě mobilních zařízení, ale začíná se už objevovat i v osobních počítačích, konkrétně u modelu all-in-one. [32]

Tomu, že má Android velkou budoucnost odpovídá i skutečnost, že se rok od roku snižují počty osobních počítačů a notebooků a naopak se zvyšuje prodej tabletů a mírně i mobilních telefonů. Tablety by podle Gartnera měly růst o 47% a předpokládá se prodej menších tabletů, které se se blíží mobilním telefonům. [32]

### 2.1 Platforma Android

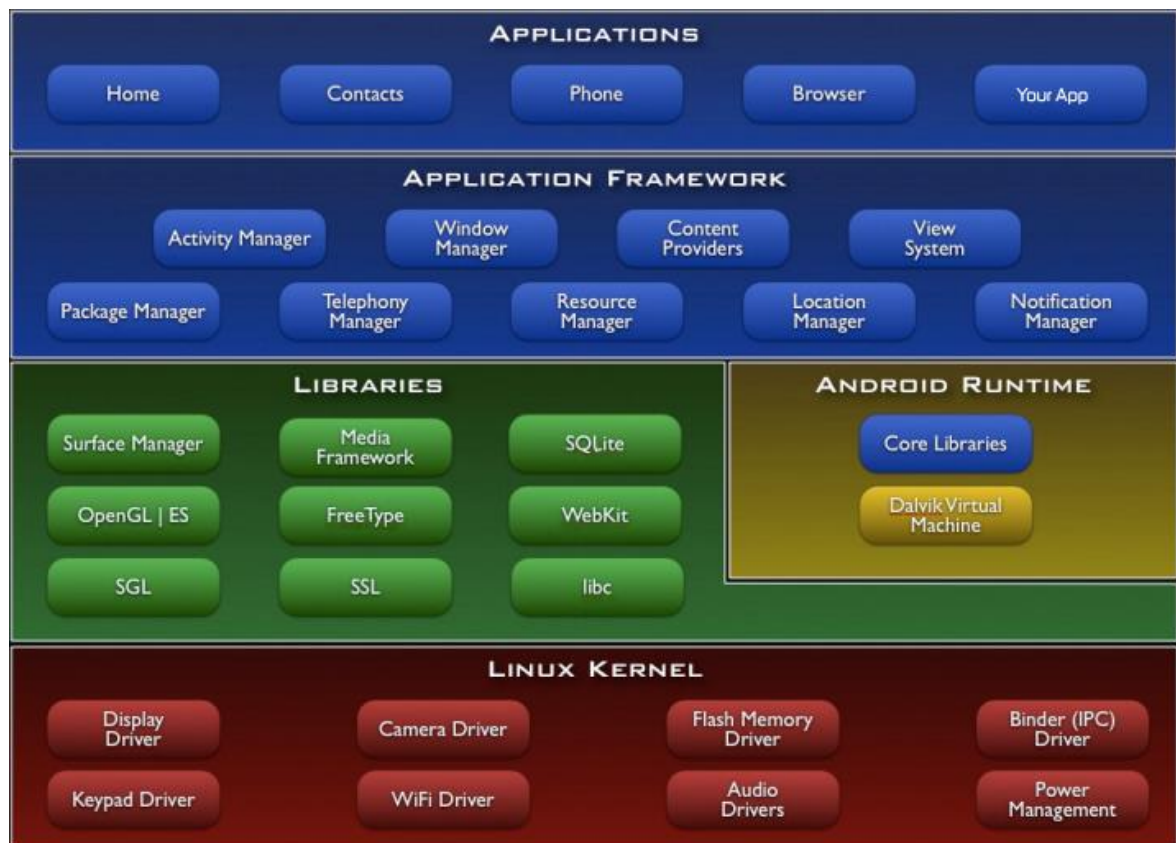
Platforma Android spatřila světlo světa 5. listopadu roku 2007. Oznámila to společnost Google. Android je postavený na Linuxovém jádře, původně určený pro PDA, chytré telefony, navigace a jiná mobilní zařízení. Google tak předal celou platformu i všechny zdrojové kódy nově založenému sdružení firem Open Handset Alliance, která poté Android uvolnila jako open source. [36]

OS Android má spoustu výhod. Hlavním cílem této platformy je, že je maximálně otevřená vývojářům, průmyslu i uživatelům. Uživatelé si mohou bez svolení instalovat jakékoliv aplikace. Ani mobilní operátoři ani výrobci telefonů nepotřebují k nasazení Androidu na své zařízení žádné svolení od Googlu či z Open Handset Alliance. [36]

### 2.2 Android architektura

Níže uvedený obrázek Obr. X., popisuje architekturu Androidu. Ta se skládá z jednotlivých vrstev - Linuxového jádra (*Linux Kernel*), knihoven (*Libraries*), Android Runtime, aplikačního frameworku (*Application Framework*) a z aplikací (*Applications*). [37]





Obrázek 2. Android architektura [37]

## Linux Kernel

Linuxové jádro je základní vrstvou a působí jako abstraktní vrstva mezi hardwarem a dalšími softwarovými vrstvami. Linuxové jádro s hardwarem komunikuje a obsahuje všechny základní hardwarové ovladače. Tyto ovladače řídí a komunikují s hardwarem. [37]

## Libraries

Další vrstvou jsou nativní knihovny. Knihovny umožňují zařízení zpracovávat různé typy dat. Jsou napsány v jazyce C nebo C++ a jsou specifické pro konkrétní hardware.

Za zmínku určitě stojí SQLite, databázový engine, který slouží pro ukládání dat. Dalším je WebKit, vykreslovací jádro k zobrazení HTML obsahu a OpenGL, které se používá k zobrazení 2D nebo 3D grafického obsahu. [37]

## Android Runtime

Android Runtime se skládá ze dvou částí. První část je jádro knihoven v Javě (Core Java Libraries), které jsou odlišné od standardních knihoven Java SE a ME. Tyto knihovny jsou rozšířeny, ale pořád vychází z Java SE knihoven. [37]

Druhou částí je Dalvik Virtual Machine, DVM, což je typ Java Virtual Machine v Androidu. Princip je takový, že každá android aplikace běží ve svém vlastním procesu se svými proměnnými DVM. DVM na rozdíl od JVM nespustí třídy souborů (.class), místo toho běží .dex soubory. Aplikace jsou tedy spouštěny v izolovaném prostředí (Sandbox) a každá aplikace musí mít přidělena práva k systémovým prostředkům. Např. zápis na SD kartu, čtení kontaktů... Každá tato aplikace je přidělena část paměti a tím, že běží v izolovaném prostředí je zajištěna vysoká bezpečnost aplikací. DVM je vyvíjena společností Google. [37]

### **Application Framework**

Jedná se o určité bloky, se kterými aplikace přímo interagují. Tyto programy řídí základní funkce telefonu jako například správa hlasového volání, řízení zdrojů atd. Mezi nejvýznamnější z nich patří např. Activity Manager – řídí životní cyklus aktivity aplikací, Content Providers – řídí sdílení dat mezi aplikacemi, Telephony Manager – řídí hlasové hovory, Location Manager – správce polohy díky GPS a Resource Manager – správce zdrojů. [37]

### **Applications**

Aplikace jsou nejvyšší vrstvou v architektuře Androidu. Mezi nejznámější patří např. správce kontaktů (Contact manager), SMS client, webový prohlížeč (Web Browser) a vytáčení hovorů (Dialer). [37]

## **2.3 Jednotlivé verze**

### **1.5 Cupcake, 30. duben 2009**

První verzí byla 1.5 Cupcake, dortík, která s sebou přinesla vylepšené funkce kopírovat a vkládat, vylepšenou softwarovou klávesnici, nové widgety, nahrávání videí a bluetooth – A2DP (protokol pro přenos hudby). [33]

### **1.6 Donut, 15. září 2009**

Druhá verze, která přišla za necelého půlroku na to, s sebou přinesla možnost hromadného mazání fotografií, nový Android Market, vylepšenou aplikaci fotoaparátu, překládání textu na hlas, vyhledávání, podpora rozlišení WVGA a některé vizuální změny (např. zesvětlená notifikační lišta apod.). [33]

**2.0, 2.1 Eclair, 26. říjen 2009**

Verze Eclair s sebou přinesla novější vzhled, podpora animovaných tapet, nový prohlížeč, podpora HTML 5, podpora led na zadní straně telefonu zařízení, optimalizace HW a další. [33]

**2.2 Froyo, 20. květen 2010**

Další verzi byla Froyo, která už podporovala Adobe Flash 10.1, možnost vytvoření přípojného bodu k WiFi, aplikace již bylo možné instalovat na paměťovou kartu a nový noční režim a režim v autě. [33]

**2.3, 2.4 Gingerbread, 6. prosinec 2010**

K poměrně zásadním vizuálním změnám došlo ve verzi Gingerbread, došlo také k výrazné aktualizaci softwarové klávesnici, je zde nová verze Google maps, vylepšené funkce kopírování a vložení, podpora nových senzorů, protokolu SIP a NFC a mnohé další. [33]

**3.0, 3.1, 3.2 Honeycomb, 22. únor 2011**

Design, optimalizovaný pro tablety s sebou přinesla verze Honeycomb. Ta však nefungovala na mobilních telefonech. Přinesla s sebou také nový prohlížeč, vylepšený multitasking a nové funkce USB Host. [33]

**4.0, 4.0.4, Ice Cream Sandwich, 19. října 2011**

Nový launcher s sebou nesla verze Ice Cream Sandwich téhož roku. Uživatel si již mohl zpříjemnit focení také panoramatickým focením. Vylepšení se také dočkalo správce kontaktů, Android beam, nový ukazatel přenesených dat a opět další vizuální změna oproti předchozí verzi. [33]

**4.1, 4.2 Jelly Bean, 27. červen 2012**

Jelly Bean, jedna z nejpopulárnějších verzí, měla vylepšenou notificační lištu, Project Butter, Google Now, vylepšenou aplikaci fotoaparátu, podporu multi account. A mnohé další. [33]

#### **4.3 Jelly Bean 24. červenec 2013**

Jelly Bean 4.3 pak přináší Bluetooth Low Energy, Open GL ES 3.0, vylepšený číselník a klávesnice, nově zpracovaná grafika, určení polohy přes WiFi, Bluetooth, opět vylepšená notificační lišta, prohlížeč, Google Play a vylepšená funkce Daydream u fotografií. [33,34]

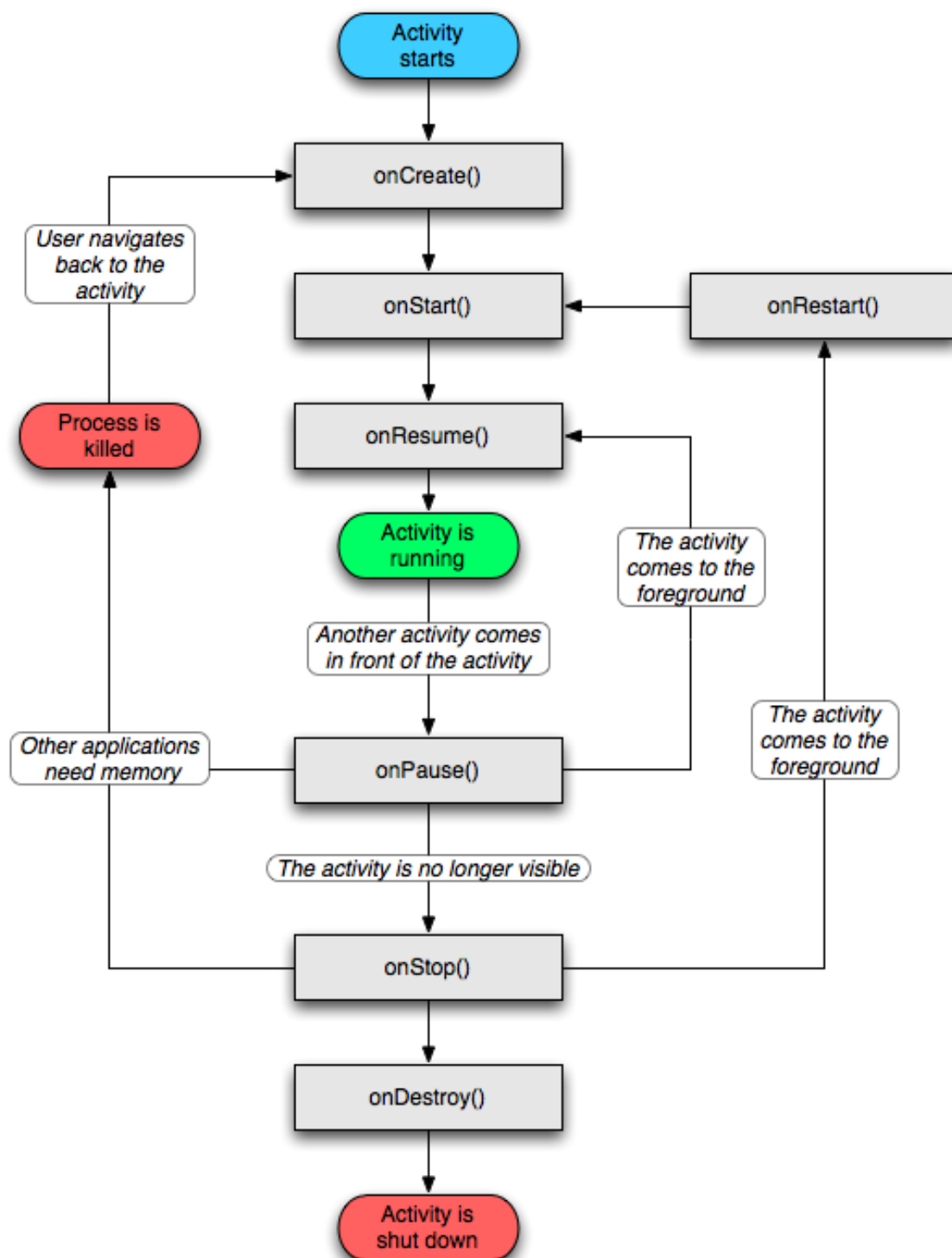
#### **4.4 KitKat, 3. září 2013**

Aktuální verze KitKat s sebou přinesla drobné změny úvodní konfigurace telefonu, nový launcher zvaný Google Experience a jeho částečně průhledná nabídka s tlačítky a oznamovací oblast. Další novinku zaznamenala aplikace Telefon, která po spuštění zobrazí často volané a oblíbené kontakty. Google Cloud Print je nyní nativní součástí Androidu, takže je zde možnost tisku přes položku Tisk. Je možné tak nastavit barvu tisku, orientaci papíru, počet kopií, velikost papíru, oblast apod. Změny se dočkalo také hlasové vyhledávání. Další zásadní změnou je přesun budíků na samostatnou záložku. Také novou možností je nahrávat obrazovku ze zařízení. A poslední takhle zásadní změnou je integrace SMS do Hangouts, čili sdružení všech konverzací do jedné aplikace. A další drobnosti, jako např. nové systémové ikony, odemykací obrazovka během přehrávání hudby, defaultně vypnuté widgety na obrazovce, vylepšený kancelářský balík QuickOffice a mnohé další. [35]

#### **5.0 Key Lime Pie**

Nejnovější verzí Androidu je Key Lime Pie 5.0 a očekává se v červnu 2014. Chystanými novinkami se nechme překvapit. [38]

### 3 ŽIVOTNÍ CYKLUS AKTIVITY



Obrázek 3. Životní cyklus aktivity v Androidu [55]

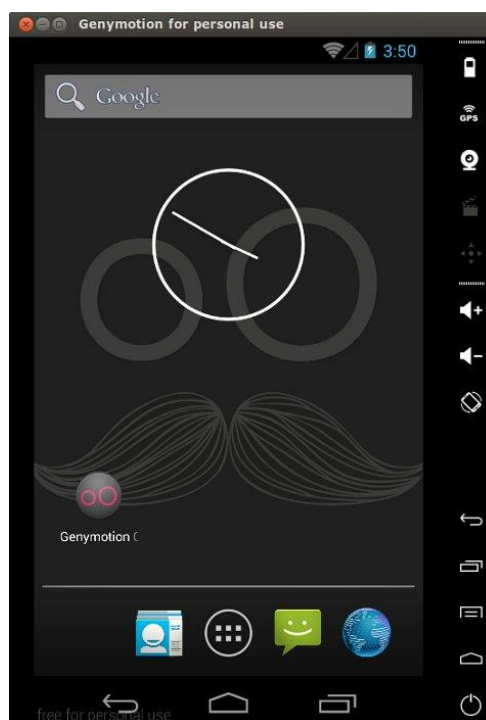
## 4 GENYMOTION

Genymotion, dříve známý jako AndroVM, je nástroj, který slouží pro spouštění OS Android ve Windows, Linux i Mac OS. Tento program pracuje jako emulátor, který je založen na virtualizačním nástroji VirtualBox. Program je užitečný hlavně pro vývojáře, kteří tak mohou testovat své aplikace nebo pokročilejší nastavení systému. [23]

V současné době se prodává ve třech verzích. Všechny podporují verze androidu 2.3, 4.1, 4.2, 4.3 a 4.4, mají GPS Widget – doplněk pro snímání polohy a Camera Widget, čili podpora webkamery. Tohle obsahují všechny balíčky, Free balíček, je zdarma pro nekomerční použití, balíček Indie se prodává za 99 € bez daně a za Business balíček 299 €. Za daň se platí 136 \$ (Indie) a 412 \$ (Business) za rok. [23,24]

Při instalaci je třeba mít nainstalovaný Oracle VirtualBox software, který je součástí instalace Genymotion a mít grafickou kartu, která podporuje OpenGL 2.0.

Genymotion patří k nejlepším Android emulátorům současnosti. Jeho velkou výhodou je rychlost, snadnost použití a konfigurace a obsahuje pokročilé funkce ladění. Jeho snadnou nevýhodou je, že se mohou vyskytnout problémy se stabilitou. [25]



Obrázek 4. Genymotion [25]



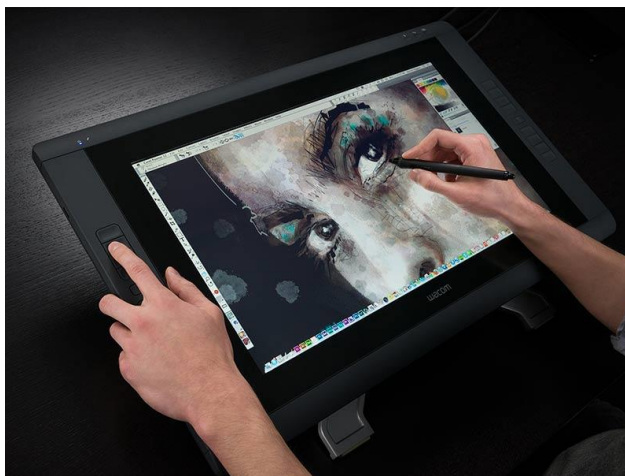
## 5 GRAFICKÉ NÁSTROJE

Pro aplikace s 3D modely je třeba si vybrat z již hotových 3D modelů (ve formátu \*.obj nebo \*.dae), jako nabízí např. <http://www.3dvia.com/> nebo je nutné využít 3D grafický program, ve kterém bude 3D model vytvořen. Na trhu je jich spousta. Zde je nastíněno pár významných představitelů jak z komerční, tak i z nekomerční oblasti. Rovněž bude představeno poměrně zajímavé řešení 3D modelů tzv. on-line aplikací. [39]

### 5.1 Jak se tvoří 3D model

3D modely lze vytvářet buď klasicky – pomocí myše a klávesnice nebo pomocí speciálních komponent. Na trhu už poměrně dlouho existují tzv. grafické tablety nebo novější tzv. leap motion. Oba nástroje jsou již poměrně specializovanými a s jejich pomocí lze vytvořit skutečně přesné modely – třeba u sculptingu<sup>1</sup>.

I když grafické tablety jsou typické zejména u 2D grafiky, své zastoupení má i u 3D. U grafických tabletů má největší zastoupení firma Wacom, která vyrábí tyto tablety od 1200 Kč až po profesionální za 65 000 Kč. Každý tablet má své speciální pero, které umí snímat různé stupně přítlaku. Existují i rukavice, díky kterým si uživatel nedělá čmouhy na displeji. [18,19]

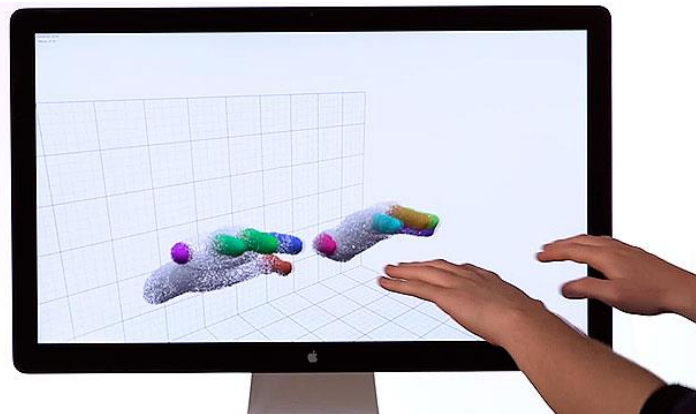


Obrázek 5. Ukázka tabletu Wacom Cintiq 22HD [20]

---

<sup>1</sup> Sculpting – je modelovací technika, která připomíná práci sochařů. Je výhodná pro tvorbu organických objektů. [15]

Leap motion, přibližně rok „stará“ novinka, je určena pouze pro snímání pohybu rukou (oproti Kinectu). Nejen, že s ním lze vytvářet 3D modely, ale lze s ním surfovat na internetu, hrát hry a mnohé další. Leap motion je možné pořídit za €89.99. [21]



Obrázek 6. Ukázka Leap Motion [21]

## 5.2 Komerční 3D programy

Komerčními programy rozumíme ty, za jejichž použití musí její uživatel zaplatit. Jsou zde možné různé demo (omezené funkce, nemožnost ukládání výsledné animace apod.) verze nebo zvýhodněné balíčky pro studenty, nicméně plné verze, neomezené, jsou vždy za peníze. Všechny níže uvedené programy nabízí vytvořit komplexní objekt. Obecně lze říci, že v důsledku nabízejí totéž, jen má každý program své specifické prostředí a ovládání. Samozřejmě mají také různě propracované funkce. U některých z nich se pro výslednou animaci může využít dalších specializovaných programů (V-Ray, Lumion 3D, Shade 3D, Luxrender, apod.), některé mají tyto renderovací programy zabudované v sobě a některé tuto možnost ani neumožňují, proto je třeba využít právě jiných programů.

### 5.2.1 Autodesk 3ds Max

Autodesk 3ds Max je jedním z nejpoužívanějších 3D programů na světě.

3ds Max 2014 obsahuje nové nástroje animaci částic (tzv. Particle Flow) a spojování perspektivy včetně podpory stínovačů Microsoft DirectX 11 a velkou novinkou jsou nástroje pro generování davu. Na trhu už je i Autodesk 3ds Max 2015 – doporučuje se mít už 8GB RAM, 4.5GB volného prostoru na disku a 64-bit Intel® nebo AMD® více jádrový procesor. [1]

### 5.2.2 Autodesk Maya

Autodesk Maya je dalším profesionálním 3D grafickým programem pro grafickou animaci, modelování, simulaci a renderování. Systémové požadavky jsou stejné jako na 3ds Max. Rovněž je využíván ve filmech i počítačových hrách. Maya byla využita pro mnoho významných filmů, které byly následně oceněny, např. Avatar, Hledá se Nemo, Příšerky s.r.o., Univerzita pro příšerky, Vzhůru do oblak a jiné. [8-10]

### 5.2.3 Autodesk Softimage

Autodesk Softimage je program, specializovaný pro animaci postav a procedurální tvorbu speciálních efektů (ICE). Smutnou zprávou pro všechny příznivce tohoto grafického programu je, že Autodesk Softimage v dubnu letošního roku končí. Vyjde tak její poslední verze, tj. 2015. Definitivní konec hlásí Autodesk na konci dubna 2016 a ukončí také jeho podporu. [4,5]

### 5.2.4 Zbrush

Dalším velmi zajímavým nástrojem pro vytváření objektů je ZBrush. Zde se očekávají spíše umělecké sklony a cit pro sochaření a malování. ZBrush je velmi podobný práci sochaře, který modeluje své sochy z hlíny. Program obsahuje spoustu funkcí pro reálnou podobu těchto organických modelů, u kterých není nutné držet přesný geometrický tvar. Požadovaný 3D objekt totiž vzniká obrazně nanášením jednotlivých vrstev „hlíny“ na povrch modelu, jeho vytahováním, hnětením, vrásněním a vyhlazováním. [3]

### 5.2.5 Poser

Poser, s aktuální verzí Poser 10 a Poser Pro 2014, je 3D program, který disponuje velkým množstvím již předem nachystaných postav, zvířat či jednoduchých objektů. Oblíbený je nejenom u umělců. V programu lze velmi rychle a efektivně vytvořit dokonalý snímek či animaci. Program dále disponuje speciálními nástroji, jako jsou designér chůze či designér řeči. Poser 9 s sebou přinesl velké novinky jako je např. tzv. Weight Map Rigging, což je „váhová mapa ohybu“. Tato funkce mění tvar povrchu objektu (pozici polygonů) během jeho ohybu. U postavy tak lze docílit velmi přirozeného ohybu svalů pohybujících se končetin. Další významnou funkcí představuje v podobě podpovrchového rozptylového světla. Tváře tak mají lépe nasvíceny a reálnější vzhled. [2]



Obrázek 7. Poser 10 ukázka postavy [7]

### 5.2.6 Lightwave

Lightwave, další 3D grafický program, určený pro modelování, animaci a renderování. Jedná se o jeden z finančně nejdostupnějších grafických programů (dá se sehnat za 22 875 Kč bez DPH). Lightwave se využívá nejen pro filmové efekty, vývoj her, tiskovou grafiku, architektonickou vizualizaci, ale i v televizní produkci. LightWave pomohl nejednomu umělci vyhrát cenu Emmy. [6]

### 5.2.7 Cinema 4D

Výrobce grafického programu Cinema 4D, německá společnost MAXON Computer, patří rovněž mezi světově uznávanou špičku ve svém oboru. Mezi hlavní produkty firmy MAXON nepatří jen Cinema 4D, ale i BodyPaint 3D, které se rovněž využívají ve filmovém průmyslu, architektuře, reklamě, vědě, strojírenství a jiných oblastech. Firma Maxon má své zastoupení ve Velké Británii, USA, Francii a Japonsku. [11]

## 5.3 Nekomerční 3D programy

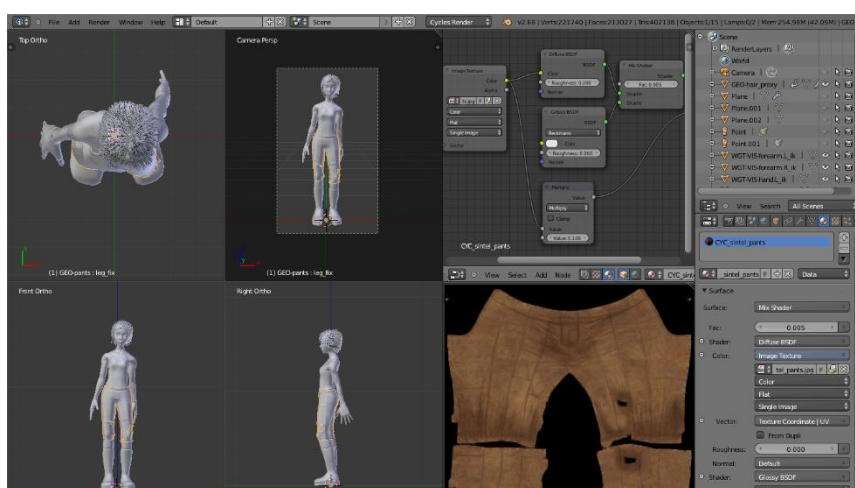
Je spousta důvodů, proč stačí využít nekomerční programy. Mezi dva nejlepší bezplatné 3D programy patří bezesporu Blender a pak Google SketchUp. Bezplatných 3D programů pro modelování objektů je ale mnohem více.

Patří sem např. Wings 3D, obsahuje nástroje i pro osvětlení, materiálů, ale nepodporuje animace. Dalším programem je DAZ Studio. Velkou výhodou tohoto programu je, že obsahuje již před chystané postavy a jiné objekty, které si může uživatel následně dotvořit

podle svých představ. Hodí se rovněž pro vektorovou grafiku – ilustrace knih či komiksů. Tomu všemu i odpovídá velikost instalačního souboru pro DAZ (911 MB). Dalším oblíbeným nástrojem je Bryce. Používá se pro doplnění tvorby z DAZ Studio. Bryce se na rozdíl od DAZ Studia specializuje na tvorbu krajín a prostředí. Takže opět obsahuje balíček před chystaných modelů. Je možné i přidat efekty vody, skály, oblohy, mlhy či oblaku a umožňuje animaci. Posledním úspěšným zástupcem bezplatných 3D programů je Sculptiris. Jde o bezplatnou variantu k profesionálnímu nástroji ZBrush od stejného autora. [12]

### 5.3.1 Blender

Blender je open-source 3D grafický program. Umožňuje modelování, renderování, animaci a tvorbu her. Výhodou Blenderu je, že je zdarma spolu se zdrojovými kódy. Další výhodou je velikost a hardwarová nenáročnost. Nevýhodou může být uživatelské prostředí pro začínající uživatele. Je třeba si zvyknout na jiné klávesové zkratky a jiné prostředí, než jsou zvyklí např. z OS Windows. [14,15]



Obrázek 8. Ukázka prostředí v programu Blender [14]

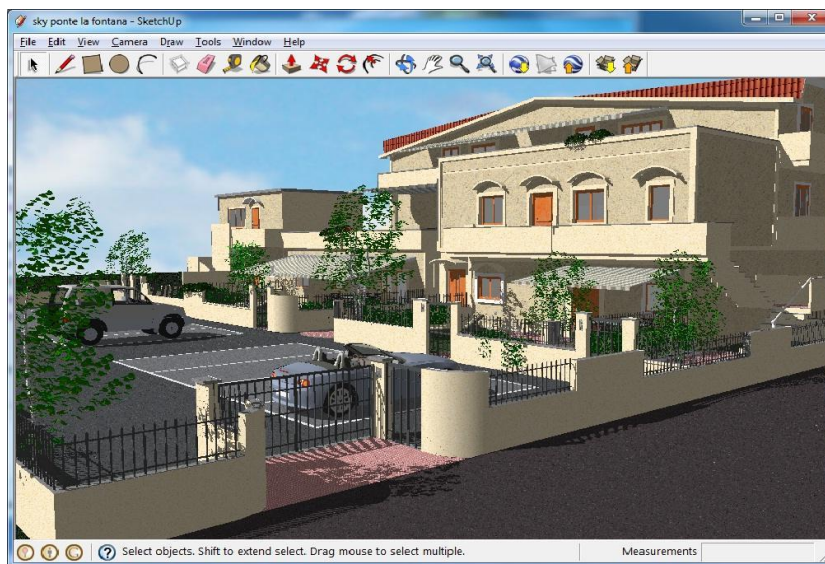
### 5.3.2 Google SketchUp

Google SketchUp se nabízí ve dvou verzích. První verze, SketchUp Make, je zdarma a umožňuje tvořit a sdílet 3D modely, hledat a stahovat modely z 3D Warehouse a pracovat offline bez internetového připojení. Ta druhá verze, SketchUp Pro 2014 obsahuje funkce navíc (je možné vytvářet dokumenty a prezentace, tisk výkresů v měřítku, technická podpora), za které už se platí. Cena je odvozena podle počtu licencí.

Pohybuje se přibližně od 10.130 Kč do 12.180 Kč pro Windows i OS Mac OS.

Pro vzdělávací instituce je tahle cena podstatně nižší. Studentská licence je \$49, ale musí splňovat určité podmínky.

Export objektů umožňuje do formátu 3DS File (\*.3ds), COLLADA File (\*.dae), OBJFile (\*.obj) a je kompatibilní s CAD, čili AutoCAD DWG File (\*.dwg), AutoCAD DXF File (\*.dxf) a další (\*.fbx, \*.ifc, \*.kmz, \*.wrl, \*.xsi). [13]



Obrázek 9. Ukázka prostředí v programu SketchUp [16]

## 5.4 Online 3D aplikace

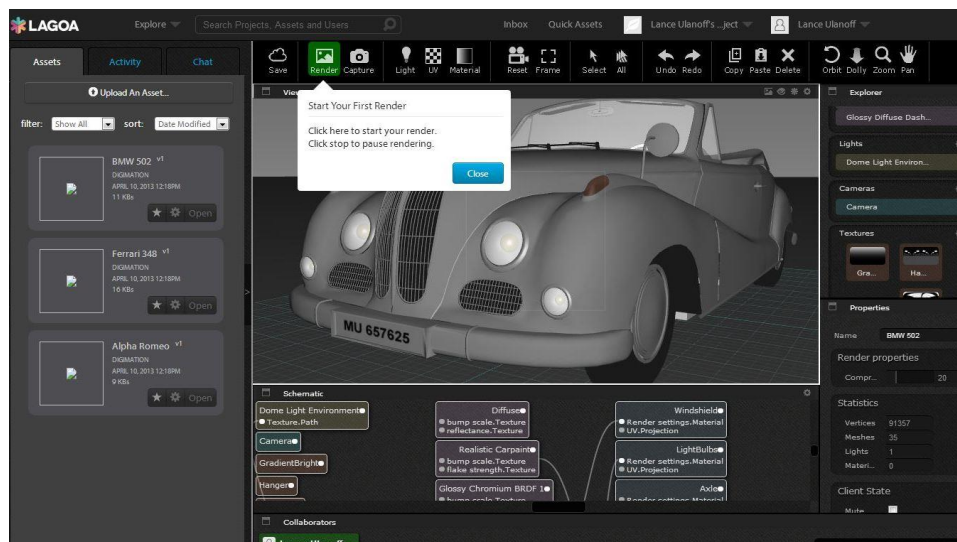
Dalším způsobem, jak tvořit 3D aplikace je poměrně nový způsob a to sice bez instalování speciálního 3D SW pro tvorbu objektů. Takovou poměrně známou je dnes online aplikace od společnosti Lagoa. Je to aplikace, která uživateli umožňuje vytvářet modely přímo na webových stránkách. Stačí se pouze přihlásit a uživatel může tvořit. Běží na všech OS a na prohlížečích, podporující WebGL. Dokonce běží i na iPadu.

Online aplikace mají tři velké výhody. První výhodou, jak již bylo naznačeno, je přístup do aplikace prakticky odkudkoliv prostřednictvím webového rozhraní, tudíž odpadá jakékoliv instalování, nastavování, pluginy apod. [22]

Druhou velkou výhodou je využití cloud computingu. K datům totiž může přistupovat nejen uživatel, ale i členové jeho týmu. Což má pozitivní dopad na týmovou spolupráci. Jakákoliv spolupráce tak probíhá rychleji a efektivněji. A poslední velkou výhodou společnosti Lagoa je, že je částečně zdarma. Neplatí se za free balíček, který obsahuje 5GB prostor, dvě hodiny



renderingu na měsíc, jeden background rendering a jednu web session. Pro neomezený render time, 100GB prostor a 3x background rendering a 3x web session si uživatel připlatí 50\$ měsíčně za balíček Professional. [22]



Obrázek 10. Ukázka prostředí v programu Lagoa [17]

## 6 VÝVOJOVÉ FRAMEWORKY

Vývojových frameworků pro práci s 3D modely existuje spousta. Pokud řešíme 3D frameworky s licencí Open Source, pak je většina psána v C, C++ nebo v Javě. Pro práci s 2D pak mohou být napsány v jazyce C, Javascript, Python, Cython, Objective-C, C#, haXe i pro 3D, Lua a ActionScript 3. [40]

Pro Javu s licencí Open Source je zde vybíráno ze 7 frameworků. Alien3D, Ardor3D, Min3D, Dwarf FW, jMonkeyEngine, jPCT-AE a libGDX. Mezi nejschopnější z nich patří Min3D, Dwarf FW, jPCT-AE a libGDX. Proto budou tyto frameworky v následující kapitole charakterizovány a ve výsledku budou v tabulce přehledně mezi sebou srovnány na stejném objektu. [40]

### 6.1 Min3D

Min3D je framework, který podporuje pouze Android, je určen pouze pro 3D s licencí MIT (Open Source), takže je zdarma a je napsán v jazyce Java. Použití frameworku je pro pokročilejší uživatele. [40]

Obsahuje například funkce pro klonování animovaných objektů, použití akcelometru, načtení 3Ds objektu i MD2 souboru, jak načíst soubor ve formátu obj a více obj, jak na SkyBox a zamlžení. [41]

#### 6.1.1 Klonování animovaných 3D objektů

Existují dva způsoby, jakými je možné objekty naklonovat. V prvním případě se jedná o kopii objektu, která přesně kopíruje originální objekt a veškeré změny provedené na tomto modelu se zobrazí i na kopii. Tzn., že jde o jakýsi pointer. Přenesou se všechny normály, textury i vertexy. [42]

Pro první příklad [42] tzn. shallow copy:

```
doomGuyClone = (AnimationObject3d) doomGuy.clone(false);
```

Druhý způsob pouze naklonuje objekt, včetně vertexů, UV souřadnic a normál, ale všechny objekty mohou být animovány nezávisle na sobě. To znamená, že se upravují nezávisle na originálním objektu. [42]

pro druhý příklad [42], tzn. deep copy:

```
doomGuyClone = (AnimationObject3d) doomGuy.clone(true);
```

Oba způsoby mají své výhody i nevýhody. První případ (*shallow copy*), může být výhodná v případě např. stromů, kdy v paměti zabírá místo pouze první strom. Nevýhodou, že všechny objekty jsou totožné a veškeré změny na originálním objektu se projeví i na dalších objektech. Výhodou druhého způsobu je, že není potřeba vytvářet další objekty ručně a je možnost si je libovolně přizpůsobit, nicméně stoupá paměťová náročnost. [42]

### 6.1.2 Použití akcelometru

Hlavní třída implementuje z `SensorEventListener` funkci `onSensorChanged()`, přičemž kód [44] může vypadat následovně:

```
@Override  
  
public void onSensorChanged(SensorEvent event) {  
  
    if (event.sensor.getType() != Sensor.TYPE_ACCELEROMETER)  
  
        return;  
  
  
    // low-pass filter to make the movement more stable  
  
    mAccVals.x = (float) (-event.values[1] * FILTERING_FACTOR +  
        mAccVals.x * (1.0 - FILTERING_FACTOR));  
  
    mAccVals.y = (float) (event.values[0] * FILTERING_FACTOR +  
        mAccVals.y * (1.0 - FILTERING_FACTOR));  
  
  
    scene.camera().position.x = mAccVals.x * .2f;  
  
    scene.camera().position.y = mAccVals.y * .2f;  
  
}
```

### 6.1.3 Načtení souboru \*.obj

Nejdříve je nutné exportovat tento objekt z grafického programu, včetně materiálů a souřadnic a tento soubor exportovat do `[your android project]/res/raw`, oba soubory `.obj` i `.mtl` je dobré si upravit např. `camaro.obj` na `camaro_obj` a `camaro.mtl` na `camaro_mtl`. [43]

```
private Object3dContainer objModel;
```

```
IParser parser = Parser.createParser(Parser.Type.OBJ,
getResources(), "min3d.sampleProject1:raw/camaro_obj");

parser.parse();

objModel = parser.getParsedObject();

scene.addChild(objModel);
```

Metoda `Parser.createParser()` má dva argumenty – typ (obj) a umístění souboru. [43]

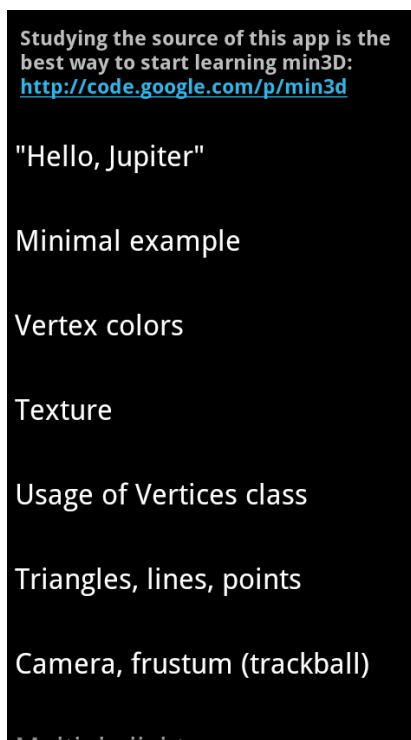
V případě načítání více objektů, je třeba si opět vytvořit kontejner [43]

```
Object3dContainer car;
```

a z něho jsou objekty načítány. Např. půjde o načtení 2 objektů. Pak je opět standardně načten objekt tak jak již bylo uvedeno výše, nicméně je třeba doplnit jej o následující kód [43]:

```
tireRR = car.getChildByName("tire_rr");
tireRF = car.getChildByName("tire_rf");
```

A pak jim opět určit pozici a rotaci.

A screenshot of a dark-themed interface showing a list of topics for studying min3D. The text is white and blue. At the top, it says "Studying the source of this app is the best way to start learning min3D:" followed by a blue hyperlink "http://code.google.com/p/min3d". Below this is a list of topics: "Hello, Jupiter", "Minimal example", "Vertex colors", "Texture", "Usage of Vertices class", "Triangles, lines, points", and "Camera, frustum (trackball)".

Studying the source of this app is the best way to start learning min3D:  
<http://code.google.com/p/min3d>

"Hello, Jupiter"

Minimal example

Vertex colors

Texture

Usage of Vertices class

Triangles, lines, points

Camera, frustum (trackball)

Obrázek 11. Min3D [50]

## 6.2 Rajawali

Rajawali je nástupcem Min3D. Vychází z frameworku Min3D, ale je obohacen o další funkce, propracovanější objekty a detailnější práci s texturami. Je to tedy 3D engine, využívá OpenGL ES 2.0/3.0. [45]

Hlavními přednostmi tohoto frameworku je import souborů ve formátu .obj, .md2, .3ds a .fbx, což jsou prakticky nejpoužívanějšími formáty při exportu modelů. Tyto formáty podporují i komerční i nekomerční 3D grafické programy. Rajawali dále nabízí předdefinované objekty jako jsou cube (krychle), sphere (koule), particle (částice), plane (rovina) a jiné jednoduché objekty. Obsahuje bodové a směrová světla, podporu materiálů, stíny, živé tapety, skybox, 2D renderer a mnohé další. [47]



<https://github.com/MasDennis/Rajawali>

01. Basic Example

02. Load .obj model

03. Materials

04. UI Elements

05. Skybox

06. Custom Material/Shader

07. 2D Renderer

08. Using The Accelerometer

09. Particles

Obrázek 12. Rajawali [51]

## 6.3 jPCT-AE

jPCT-AE je poměrně novým frameworkem s vlastními webovými stránkami a lepší podporou. Stejně jako Rajawali má poměrně obsáhlou dokumentaci. Jde o 3D grafický engine pro stolní Javu a OS Android. Framework funguje na OS Linux, Windows, Mac OS X, Solaris x86 a na mobilních zařízeních. Podporuje OpenGL LWJGL a JOGL a používá OpenGL ES 1.x a ES 2.0 a běží na Java VM ver. 1.4 a vyšší. Je možné renderovat

v prohlížeči, na počítači, dokonce na serveru bez GPU. Mezi funkce jPCT-AE patří načtení .3ds, .obj, .md2, .asc a xml soubory, podporuje kosterní animace přes raft's Bones API, osvětlení s neomezeným počtem světelných zdrojů, ambient, difúzní a zrcadlové osvětlení, zabudované základní objekty jako jsou cubes (krychle), spheres (koule), cones (kužely) a jiné, podporuje tzv. spherical (kulové) mapování, vykreslení do textury, efekty průhlednosti, detekce kolize, rotaci interpolace a zarovnání pro lepší ovládání kamery, automaticky generuje vertexy, faces a tečné vektory, VBOs, FBOs a vertexové pole. [48]

jPCT-AE obsahuje harwarový i softwarový renderer. Příkladem významné funkce pro harwarový renderer může být například podpora RGB-scaling, podpora shaderů pomocí GLSL a že může být kombinován s FengGUI. [48]

Mezi další významné funkce ale softwarového rendereru patří zejména 2x a 1.5x převzorkování a 0.5x podvzorkování, 32bit ukládání do vyrovnávací paměti, podpora alfa kanálů a filtrování texel. [48]



Obrázek 13. jPCT-AE [53]



## 6.4 Ostatní frameworky

Tyto frameworky se pro práci s 3D objekty na úrovni pokročilý nehodí. Buď nabízí pouze základní funkce – většinou to bývají starší frameworky, které ovšem ještě stále poměrně stabilně fungují nebo jsou to profesionální frameworky určené pro herní průmysl.

Například 3D frameworky Alien3D a Ardor3D jsou sice napsány v Javě a s licencí Open source, nicméně jejich demo jsou pomalá a nestabilní. Z tohoto důvodu nebudou ani popsány.

### 6.4.1 Dwarf FW

Svého času velmi schopný 3D framework - Dwarf FW, využívá OpenGL ES. Obsahuje akcelerometr, magnetometr, touchscreen a trackball. Tento framework má základní vlastnosti – materiály, podpora VBO, osvětlení, import souborů.obj, prostorové klíčové animace, výběr objektu, binární import a export, jednoduché filtrace dat ze souborů.

Jeho velkým nedostatkem je pak práce s texturami, přímkami a body a mnohé další. Framework je starší a jeho vývoj byl pozastaven. [46]

### 6.4.2 libGDX

Dalším významným frameworkem je libGDX. Tento framework se využívá pro spoustu komerčních aplikací. Je tzv. Cross Platform, tzn. vytvořené aplikace je možné publikovat jak na OS Windows, Mac, Linux, Android, iOS tak i BlackBerry a HTML5. Nespornou výhodou libGDX je, že má licenci pod Apache 2.0, čili je Open Source. Podporuje samozřejmě i 2D. Je velmi rychlý a má vynikající podporu. Podporuje formáty .obj a .md5. [49]



Obrázek 14. libGDX [52]

## **II. PRAKTICKÁ ČÁST**

## 7 SROVNÁNÍ DOSTUPNÝCH FRAMEWORKŮ

Pro srovnání dostupných frameworků byli vybráni tři významní kandidáti. Min3D, Rajawali a jPCT-AE a využívají Open-GL. Všechny splňují požadavky – tzn., jsou napsány v jazyce Java, jsou Open-Source, jsou zdarma, jsou pro 3D a jsou určeny pro platformu Android.

Tito tři kandidáti jsou srovnáni v následujících tabulkách. Jsou porovnávány parametry, které zajímají nejen programátora, ale i uživatele. Pro přehlednost je uvedena vždy tabulka pro konkrétní 3D model.

Nejdříve bylo třeba každý framework spustit a vyzkoušet si, jak funguje. Jak se v něm načítají objekty, jak nastavují světla, kamera, materiál, textury, jakým způsobem se scaluje objekt. S každým frameworkem se pracuje jinak, takže bylo nutné pro každý framework vytvořit zvlášť testovací aplikaci. V každé testovací aplikaci byly porovnávány totožné 3D modely.

Pro testy byly využity modely ve formátu .obj + .mtl a populární formát .3ds. Byla využita databáze 3dvia<sup>2</sup>, která obsahuje spoustu 3D modelů různých formátů a různých velikostí. Tyto stránky rovněž obsahují tzv. online 3D model, kde je možné s objektem libovolně otáčet. Každý objekt obsahuje informace o souboru jako je velikost souboru, původní formát, počet povrchů, trojúhelníků, vertexů, textur a v jakém programu byl vytvořen.

Zajímavostí zůstává, že když je takový soubor importován do programu Blender, tak se tyto informace o počtu vertexů, trojúhelníků a povrchů, liší.

Bylo třeba otestovat jak jednoduché, tak i složitější objekty, aby bylo objektivně demonstrováno, jak je daný framework schopen s jednotlivými objekty pracovat.

Modely byly testovány na mobilním zařízení Sony Ericsson Xperia X12 Arc LT15i s verzí Androidu 2.3.7 a na tabletu Colorovo CityTab Lite CVT-CTL-7-WLAN s verzí Androidu 4.0.4 byla testována výsledná aplikace, která zde fungovala rovněž bez problému. Výsledné srovnání bylo vyhodnoceno na mobilním telefonu Sony Ericsson Xperia X12 Arc LT15i s verzí Androidu 2.3.7 spolu se screeny aplikací. Tablet Colorovo sloužilo pouze jako pomocné zařízení při ladění aplikace.

---

<sup>2</sup> 3DVia [online]. 2014 [cit. 2014-05-21]. Dostupné z: <http://www.3dvia.com/>

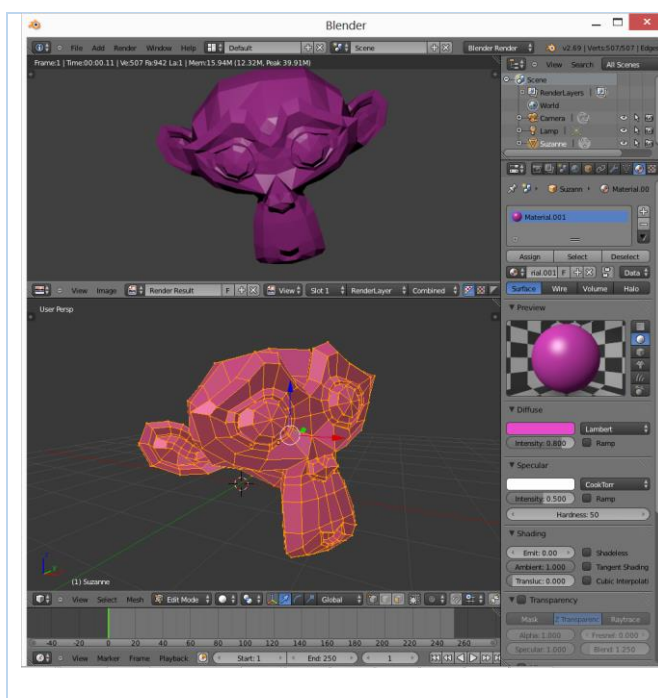


Obrázek 15. Ukázka textury draka

## 7.1 Test č. 1 - 3D model opice

Pro první test byl exportován jednoduchý 3D model ze základní nabídky v programu Blender. Výsledný model byl vyexportován do formátu .obj a .mtl, přičemž model obsahuje jednobarevný materiál a má 507 vertexů.

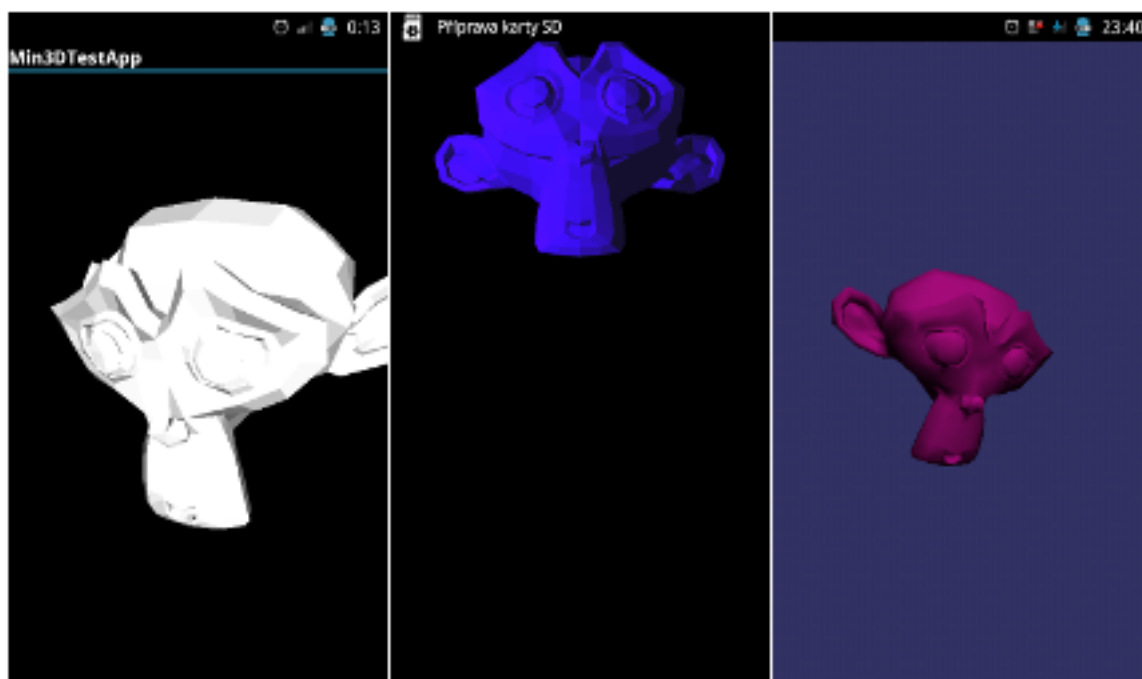
Tabulka 2. Model 1



Vertexy	507
Textura	ne
Formát	.obj, .mtl
Velikost	1 kB, 48 kB
Trojúhelníky	968
Povrchy	1
Vytvořeno	Blender

Tabulka 3. Srovnání frameworků pro Model 1

	FPS	Načtení modelu	Stabilita	Vykreslení
<b>Min3D</b>	60	2s	Velmi stabilní	Bez materiálu
<b>Rajawali</b>	60	15s	Velmi stabilní	Barvy odpovídají
<b>jPCT-AE</b>	60	9s	Velmi stabilní	Barvy odpovídají



Obrázek 16. Test č. 1

Tab. 3 shrnuje využití modelu s 507 vertexy a jednobarevným materiálem v praxi.

Všechny frameworky načítají oba soubory tzn. .obj i .mtl, přičemž nejrychlejší načtení modelu proběhlo ve frameworku Min3D (2s) a nejpomaleji byl objekt načten ve

frameworku Rajawali (15s). JPC-AE tvoří horší průměr načtení modelu a to 9s. Ve všech frameworkcích je objekt vykreslován rychlostí 60 snímků za sekundu, tj. nedochází

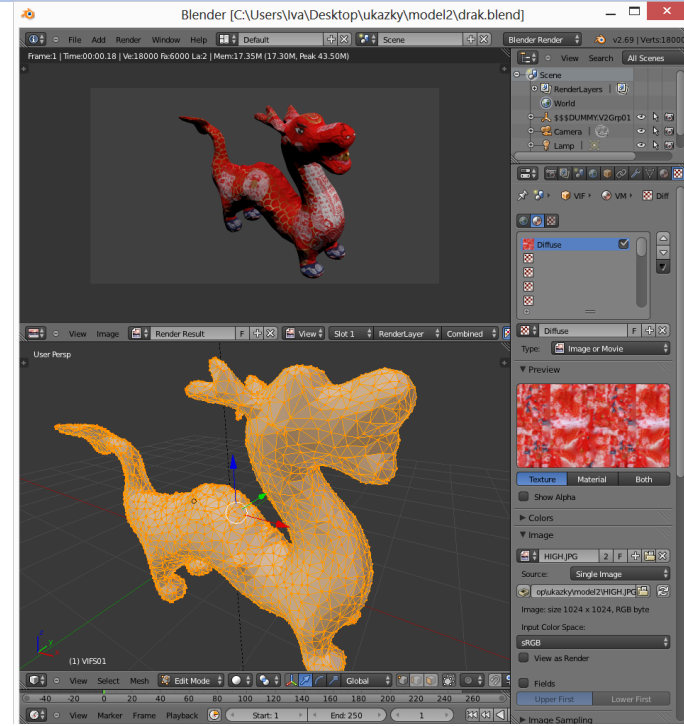
k žádným poklesům výkonu při vykreslení modelu. JPCT-AE a Rajawali vykreslují model

včetně materiálu (barevná odchylka může být dána nasvícením modelu), avšak Min3D materiál vůbec nevykreslí. Výsledná aplikace je u všech frameworků stabilní, nedochází k žádnému zasekávání, zpomalování a aplikace reaguje na změny uživatele velmi rychle.

Implementace do jednotlivých frameworků byla stejně náročná.

## 7.2 Test č. 2 – 3D model draku

Tabulka 4. Model 2

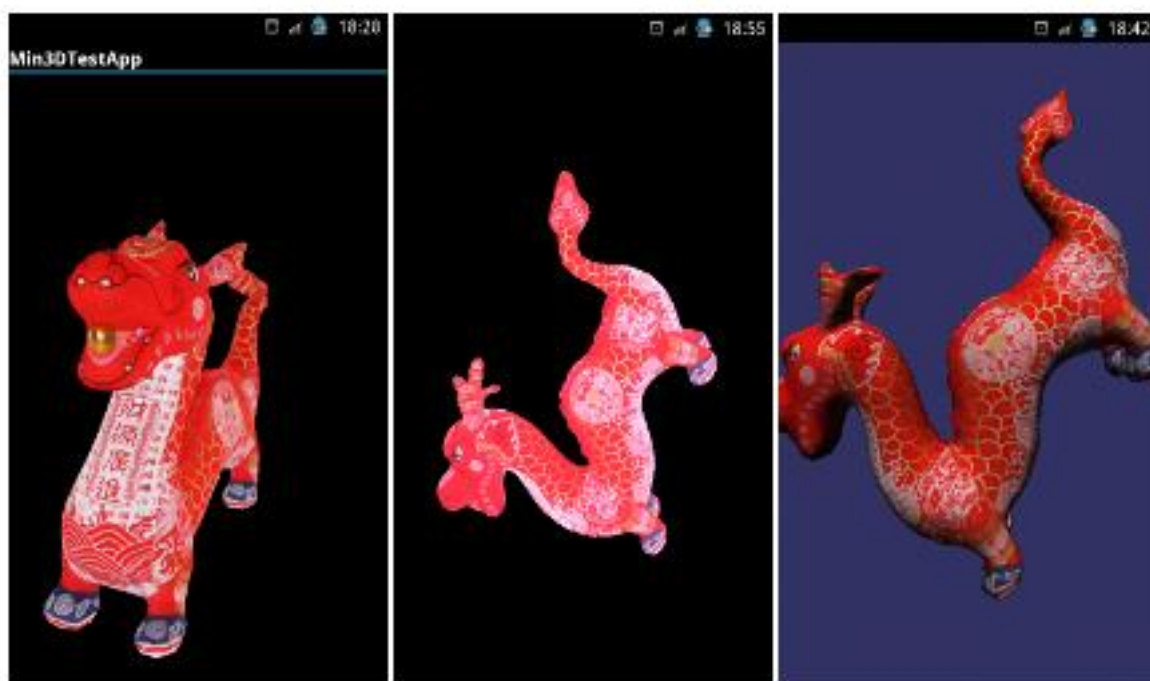
	<b>Vertexů</b>	<b>18 000</b>	
	<b>Textura</b>	1	
	<b>Formát</b>	.3ds	
	<b>Velikost</b>	797 kB	
	<b>Triangles</b>	6000	
	<b>Surfaces</b>	1	
	<b>Vytvořeno</b>	3DS (3dvia) <sup>3</sup>	3.0

<sup>3</sup>

3DVia [online]. 2014 [cit. 2014-05-21]. Dostupné z: <http://www.3dvia.com/>

Tabulka 5. Srovnání frameworků pro Model 2

	FPS	Načtení modelu	Stabilita	Vykreslení
<b>Min3D</b>	60	3s	Velmi stabilní	Barvy odpovídají
<b>Rajawali</b>	60	27s	Nestabilní	Zkreslený
<b>jPCT-AE</b>	60	7s	Velmi stabilní	Barvy odpovídají



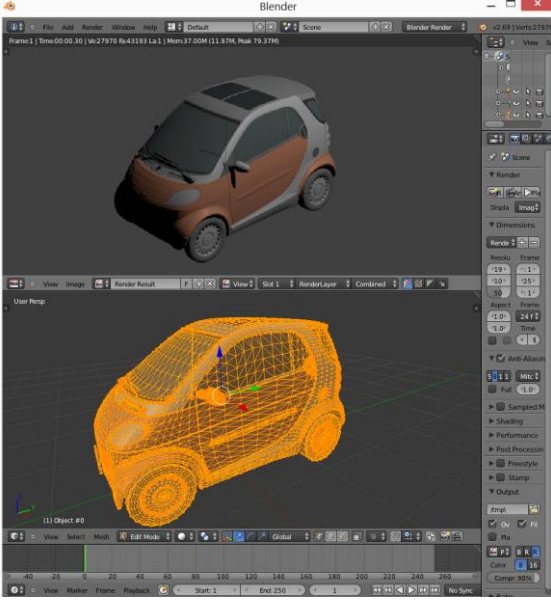
Obrázek 17. Test č. 2

Tab. 5 shrnuje výsledky pro model s texturou. Model má 18 000 vertexů a 6000 trojúhelníků. Z tabulky je vidno, počet snímků u všech testovaných frameworků zůstal neměnný, 60. Při načítání modelu měl největší problém Rajawali, kterému to trvalo 27s. Nejrychlejším framework byl opět Min3D. Při manipulaci s objektem byla aplikace ve dvou případech velmi stabilní a v jednom nestabilní. Vykreslení 3D modelu draka vypadalo v Min3D a v JPCT-AE totožně, nicméně framework Rajawali vykreslil model pokaždé v jiné barvě.



### 7.3 Test č. 3 – 3D model auta

Tabulka 6. Model 3

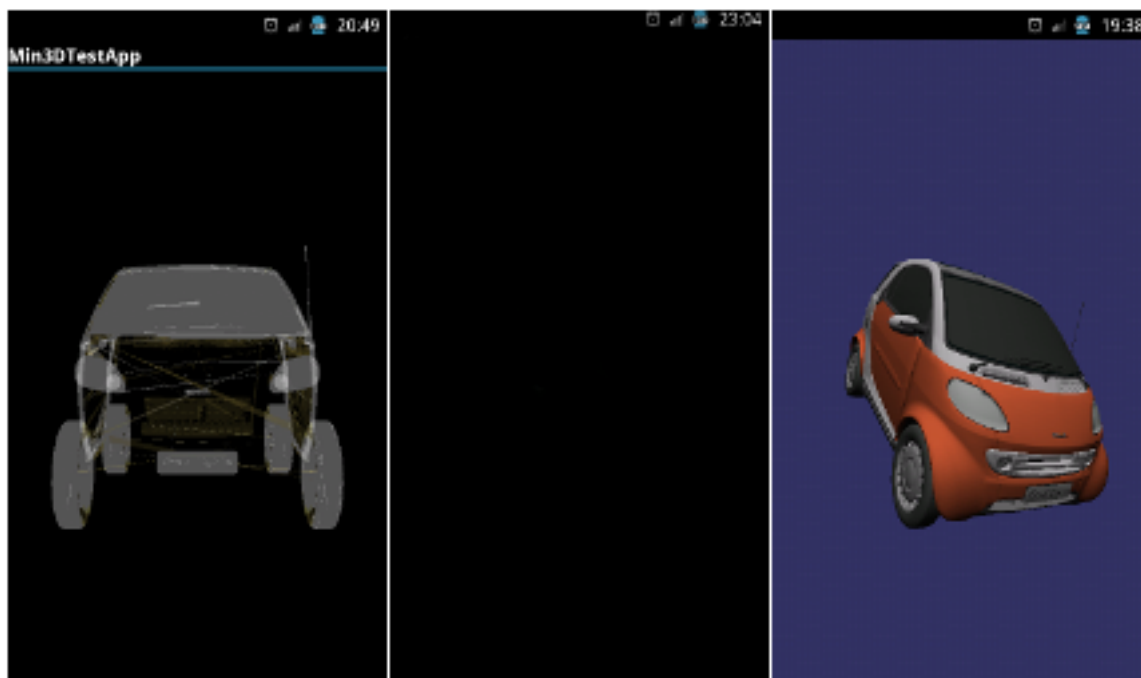
	Vertexů	33 531
	Textura	0
	Formát	.3ds
	Velikost	1,12 MB
	Triangles	43 193
	Surfaces	705
	Vytvořeno	3DS 3.0 (3dvia) <sup>4</sup>

Tabulka 7. Srovnání frameworků pro Model 3

	FPS	Načtení modelu	Stabilita	Vykreslení
<b>Min3D</b>	51	10s	Velmi stabilní	Chybné zobrazení
<b>Rajawali</b>	-	Déle než 600s	-	-
<b>jPCT-AE</b>	55	10s	Velmi stabilní	Barvy odpovídají

4

3DVia [online]. 2014 [cit. 2014-05-21]. Dostupné z: <http://www.3dvia.com/>



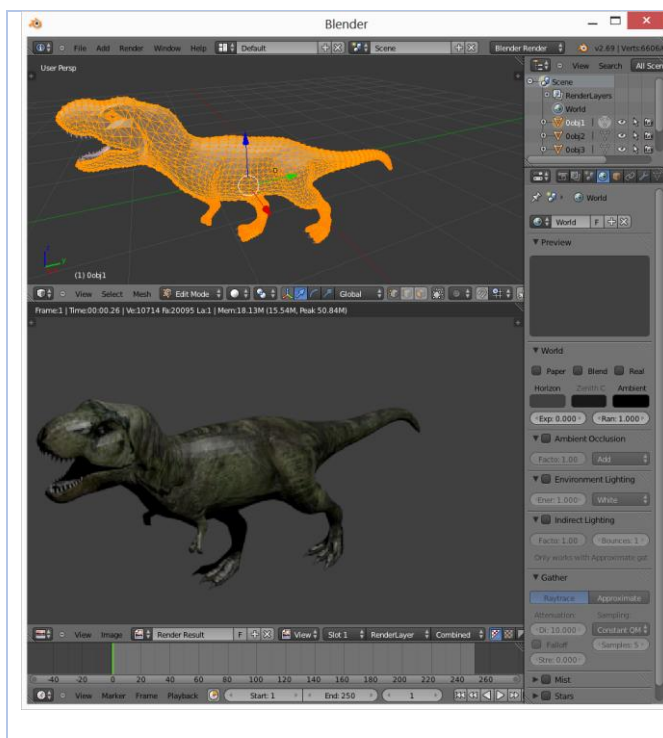
Obrázek 18. Test č. 3

Pro tento test byl zvolen velmi složitý model. Model je tvořen 33 531 vertexy, 43 193 trojúhelníky a 705 povrchy a obsahuje pouze materiál. U modelu je totiž zpracován nejen exteriér, ale i interiér vozidla.

Na základě Tab.7. je vidět, že se hodnoty FPS již značně rozcházejí. Hůře dopadlo Min3D, vykreslil totiž jen 51 snímků za sekundu. Oproti tomu jPCT-AE dopadlo nejlépe. Nejrychleji dokázal načíst model Min3D a to za 10s a framework Rajawali nenačetl model vůbec, resp. testování bylo pro Rajawali po 600s neúspěšného načtení ukončeno. V obou případech je aplikace stabilní, avšak u Min3D se nezobrazí model korektně.

## 7.4 Test č. 4 – 3D model Tyrannosaurus Rex

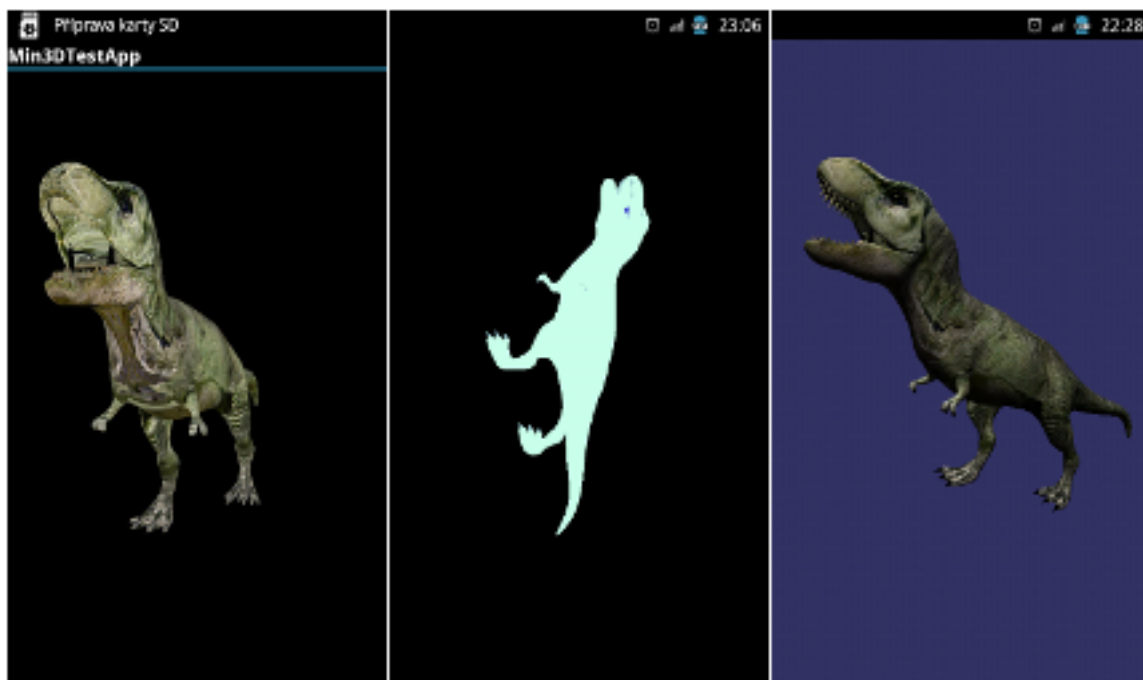
Tabulka 8. Model 4



Vertexů	12 905
Textura	1
Formát	.3ds
Velikost	2,4 MB
Triangles	20 095
Surfaces	58
Vytvořeno	3DS 3.0 (3dvia) 5

Tabulka 9. Srovnání frameworků pro Model 4

	FPS	Načtení modelu	Stabilita	Vykreslení
Min3D	59	8s	Velmi stabilní	Barvy odpovídají
Rajawali	60	24s	Nestabilní	Zkreslený
jPCT-AE	60	13s	Velmi stabilní	Barvy odpovídají



Obrázek 19. Test č. 4

Model 4 byl vybrán kvůli tomu, že je načítán ze 3 textur současně. Model obsahuje texturu zvlášť pro tělo, hlavu a oči. Ve srovnání v Tab. 8 si v FPS vedly poměrně všechny frameworky stejně – 60 FPS. V době načtení 3D modelu hodnoty velmi kolísaly. Nejrychleji model načetl Min3D za 8s, nejpomaleji Rajawali 24s. Aplikace je stejně stabilní jak u Min3D, kdy uživatel může bez problému zvětšovat objekt, tak stejně tak i v jPCT-AE. Textura byla načtena v obou případech v pořádku, chybně se zobrazila jen u Rajawali.

## 7.5 Výsledné srovnání

Uvedené frameworky se také mohou lišit například ve složitosti implementace, dokumentací a velikostí samotného frameworku. Min3D se implementuje do projektu přes *Properties*, *Java Build Path* – *Source* – *Link Source* – *Min3D framework*. Tím samym způsobem se může implementovat i Rajawali a nebo také již stačí (stejně jako u jPCT-AE) nakopírovat .jar soubor do složky *libs*.

Jediný framework jPCT-AE má vlastní webové stránky.

Na základě těchto výsledků dopadl nejlépe jPCT-AE. Proto byl následně zvolen.

## 8 3D MODEL VIEWER

Výsledek diplomové práce představuje prohlížeč 3D modelů – 3D Model Viewer, vytvořený s pomocí frameworku JPCT-AE, který využívá Open-GL. V této konkrétní aplikaci prohlížeč podporuje 3D modely ve formátu .3ds. Modely mohou být použity buď s materiálem (i bez něj) a modely s jednou texturou. Samotný import se provádí nakopírováním modelu .3ds do složky **res/raw**. V případě, že model obsahuje texturu, je nutné tuto texturu vložit do téhož adresáře **res/raw**. Framework JPCT-AE sice model načte, ale nedokáže vyrenderovat výsledný obraz pro 3D model načtený z SD karty, proto možnost *Otevřít ze souboru* byla z GUI aplikace vynechána. Nicméně její funkčnost je v následujícím textu popsána a připravena pro pozdější implementaci. Importované modely pocházejí ze stránek 3dvia<sup>6</sup> a Freebie 3D<sup>7</sup> pro svou různou složitost modelů. Nejsložitějším modelem je model auta, u kterého je zpracován i interiér vozu.

Název souborů nesmí obsahovat diakritiku, velká písmena, mezery či jiné znaky, jinak se projekt nezkompiluje. Importované soubory musí být pojmenovány pouze malými písmeny, číslicemi, případně podtržítkem a tečkou.

### 8.1 MainActivity.java

MainActivity.java, představuje hlavní, spouštěcí aktivitu aplikace. Ta je tvořena z GridView a dvou TextView pro název aplikace v hlavičce a názvem autora v patičce.

- **void onCreate(Bundle savedInstanceState)**

Tato funkce se provede při spouštění aktivity. Předávaný objekt *Bundle* slouží pro uložení předchozího stavu aktivity.

Následující funkce propojuje výsledný layout s logikou aplikace.

```
setContentView(R.layout.activity_main);
```

Rovněž zde byl nastaven příznak *FLAG\_FULLSCREEN*, který umožňuje zobrazit vykreslenou aktivitu přes celou obrazovku.

---

<sup>6</sup> 3DVia [online]. 2014 [cit. 2014-05-21]. Dostupné z: <http://www.3dvia.com/>

<sup>7</sup> Freebie 3D [online]. 2014 [cit. 2014-05-21]. Dostupné z: <http://www.freebie3d.com/>

```
requestWindowFeature(Window.FEATURE_NO_TITLE);
getWindow().setFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN, WindowManager.L
ayoutParams.FLAG_FULLSCREEN);
```

Dále byl vytvořen list a vlastní gridViewCustomAdapter, který slouží pro propojení GUI s logikou, k vytváření listů na GUI. Tomuto adaptéru je předáván seznam, který tvoří vlastní objekty a následně je tento seznam vykreslen pomocí mřížky.

```
gridViewCustomAdapter = new GridViewCustomAdapter(this, list);
// Set the Adapter to GridView
gridView.setAdapter(gridViewCustomAdapter);
```

- `gridView.setOnItemClickListener(new OnItemClickListener()`

Tato metoda obsluhuje událost kliknutí uživatele na vybranou položku.

- `void onItemClick(AdapterView<?> arg0, View arg1, int position, long arg3)`

Metoda `onItemClick(AdapterView<?> arg0, View arg1, int position, long arg3)` je děděná z `onClickListeneru` a slouží pro obsluhu dané události.

- `Object3D load3DSModel(InputStream is, float scale)`

Tato metoda slouží pro načtení 3D modelu ve formátu .3ds, přičemž má dva parametry. Prvním z nich je načítaný objekt, který je předáván ve formě *InputStreamu*. Následuje druhý parametr, který udává zvětšení modelu.

- `List<CustomModel> createList()`

Metoda `createList()` vytváří seznam modelů. Tento seznam je tvořen vlastními objekty, které reprezentují vlastní model. Model je tvořen ikonou, kterou vidí uživatel a dále jeho názvem. Další položky obsahují unikátní identifikátor daného modelu a textury, přičemž textura nemusí být specifikována. Pokud není textura specifikována, jedná se o model tvořený jednoduchým materiálem.

```
CustomModel model = new
CustomModel(getResources().getDrawable(R.drawable.icon1), "Opice", R.raw.monkey, 0
);
list.add(model);
```

- `void onResume()`

Tato metoda je volána většinou po metodě `onStart()` případně po metodě `onPause()`. V této metodě je zároveň volán i její rodič. Z důvodu zachování integrity dat je vždy nastaven objekt pro přenášení dat na hodnotu *null*.

```
super.onResume();
```

```
DataHolder.getInstance().setData(null);
```

- **void** fileDialog()

V metodě fileDialog() je vytvořen Intent, který spouští aktivitu. Je zde nastavena cesta, ze které se mají modely číst, mód pro výběr adresáře, filtr pro příponu souborů a mód pro otevření souboru.

```
Intent intent = new Intent(getBaseContext(), FileDialog.class);  
intent.putExtra(FileDialog.START_PATH, "/sdcard");  
  
intent.putExtra(FileDialog.CAN_SELECT_DIR, false);  
intent.putExtra(FileDialog.FORMAT_FILTER, new String[] { "3ds", "3DS" });  
intent.putExtra(FileDialog.SELECTION_MODE, SelectionMode.MODE_OPEN);  
startActivityForResult(intent, REQUEST_LOAD);
```

- **synchronized void** onActivityResult(**final int** requestCode, **int** resultCode, **final** Intent data)

Funkce, která zavolána po ukončení aktivity s výběrem 3D modelu.

- **void** load3dsFromFile(String path)

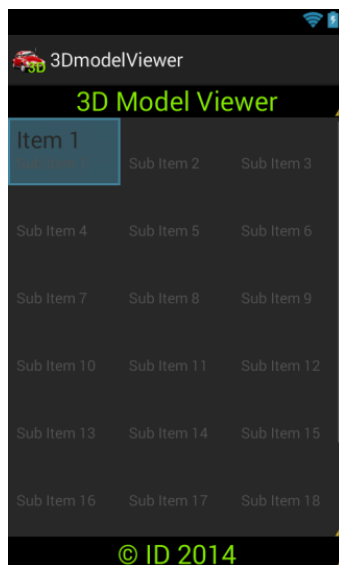
Funkce, která umožní načíst uživatelem vybraný 3D model z úložiště zařízení po načtení souboru je volána funkce startRendering().

- **void** startRendering()

Tato funkce slouží ke spuštění aktivity, která obstarává vykreslování jednotlivých modelů.

### 8.1.1 activity\_main.xml

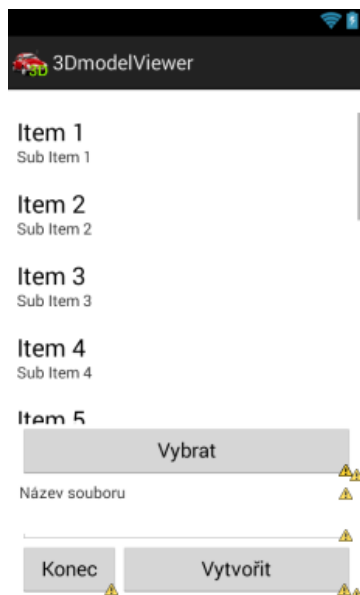
activity\_main.xml je soubor, který slouží pro návrh GUI aplikace. Návrh je složen z TextView „3D Model Viewer“, který má nastaven barvu #84e900 a je zarovnán na střed, GridViewActivity se třemi sloupci a TextView „© ID 2014“.



Obrázek 20. activity\_main.xml

### 8.1.2 file\_dialog\_main.xml, file\_dialog\_row.xml

file\_dialog\_main představuje dialog pro načtení souboru z SD karty. Tento dialog je tvořen z jednotlivých řádků, jejichž grafické rozhraní se načítá ze souboru file\_dialog\_row.xml. Tím lze specifikovat zobrazení jednotlivých řádků.



Obrázek 21. file\_dialog.xml

### 8.1.3 CustomModel.java

Třída CustomModel obsahuje deklaraci proměnných, jako je ikona, text, objectID, textureID. Při vytváření instance objektu jsou všechny tyto parametry předávány přímo



v konstruktoru. Třída tedy neobsahuje metody pro nastavování a získávání hodnot z proměnných, tzv. gettery a settery. Konstruktor vypadá následovně:

```
public CustomModel(Drawable icon, String text, int objectId, int textureId) {  
    this.icon = icon;  
    this.text = text;  
    this.objectId = objectId;  
    this.textureId = textureId;  
}
```

#### 8.1.4 DataHolder.java

Třída DataHolder představuje návrhový vzor Singleton. Tento návrhový vzor je zde využit protože není možné využít vlastní objekt pro předávání dat mezi aktivitami. Jak je u Singletonu známo, má jen jednu instanci v rámci celé aplikace.

```
private static final DataHolder holder = new DataHolder();  
public static DataHolder getInstance() {  
    return holder;  
}
```

#### 8.1.5 GridViewCustomAdapter.java

Třída GridViewCustomAdapter rozšiřuje rodičovskou třídu BaseAdapter. Je jí předávána aktivita a list List<CustomModel>. Třída umožňuje vykreslit libovolný seznam objektů, přičemž ty, které nejsou zobrazeny na viditelné části obrazovky, maže z paměti. Tím je velice inteligentně šetřena paměť zařízení.

- **SelectionMode.java**

Třída, která pouze specifikuje konstanty pro vytváření nebo načítání souborů. V rámci této konkrétní aplikace je umožněno pouze otevírání souborů.

```
public static final int MODE_CREATE = 0;  
public static final int MODE_OPEN = 1;
```

#### 8.1.6 FileDialog.java

Třída FileDialog reprezentuje další aktivitu, která umožňuje vykreslit souborovou strukturu zařízení a následně vybrat soubor podle zadaných parametrů.

### 8.2 RenderActivity.java

RenderActivity představuje druhou aktivitu aplikace. V této aktivitě dochází k načtení samotného frameworku a následně vykreslování zvoleného objektu.

- **void** onCreate(Bundle savedInstanceState)

Zde je opět nastaven Fullscreen, aby byla aplikace přes celou obrazovku. Aktivita obsahuje i ProgressDialog, který vykresluje informační hlášku pro uživatele. Tato hláška obsahuje text s tím, že probíhá vykreslování zvoleného 3D modelu.

- **boolean** onCreateOptionsMenu(Menu menu)

Tato funkce je zavolána proto, aby vytvořila uživatelské menu. Uživatelské menu je zobrazováno pomocí hardwarové klávesy na spodní straně telefonu. V závislosti na OS Android se uživatelské menu zobrazuje buď nahoře, nebo dole.

- **boolean** onOptionsItemSelected(MenuItem item)

Funkce obsahuje switch pro jednotlivé položky z nabídky. Položka „Textura“ má za úkol znovu zobrazení textury. Tzn. v momentě, kdy si uživatel zvolí z položky „Barva“ barvu pro svůj model a bude chtít zpětně vrátit svoji texturu, zvolí tuto možnost. V případě modelu s materiálem se model zbarví do bílé barvy. Jako informaci vypíše Toast hlášku „Vybrána textura“, popř. „Objekt nemá texturu.“

V případě druhé možnosti „Barva“ je vyvolán ColorPickerDialog() pro nastavení barvy uživatelem. Ten je popsán v ColorPickerDialog.java.

Poslední možností této nabídky je zvolení animace. Ta běží, dokud není znovu zvolena tato možnost. Pak se zastaví.

- **boolean** onTouchEvent(MotionEvent me)

Tato metoda zpracovává dotyky na dotykové vrstvě telefonu. V této metodě je rozhodováno, zda se jedná o dotyk jedním nebo dvěma prsty. V případě dotyku jedním prstem je vyhodnocována poloha a následně rotace daného objektu. V případě dotyku dvěma prsty je vyhodnocováno poloha bodu uprostřed těchto dvou prstů a na základě vzdálenosti mezi nimi je vypočítávána míra zvětšení či zmenšení daného objektu.

- **void** onSurfaceChanged(GL10 gl, **int** w, **int** h)

Tato metoda je volána tehdy, kdy je změněna plocha pro vykreslování objektu. Tato metoda je zavolána tehdy, když je připraveno Open-GL. Následně je v této metodě vykreslen zvolený objekt, případně přidány jeho textury.

- **void** onDrawFrame(GL10 gl)

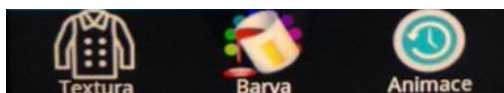
Tato metoda je volána periodicky v závislosti na vykreslení jednotlivých snímků, zpravidla 60krát za sekundu. V této metodě je obnovována uživatelská obrazovka, dále je zde počítán počet snímků za sekundu a následně pokud je to povoleno, tak automatické otáčení objektů.

- **void** colorChanged(**int** color)

Tato metoda je volána z dialogu pro výběr barvy, přičemž zvolená barva je aplikována na zvolený model.

### 8.2.1 menu.xml

Menu tvoří 3 ikony. Je tvořena z ImageView a TextView. Ikony pro aplikaci byly staženy ze stránek Iconfinder<sup>8</sup>.



Obrázek 22. Menu aplikace

### 8.2.2 model\_miniature.xml

model\_miniature obsahuje ImageView a TextView pro detail modelu v Linear Layoutu s vertikální orientací. Slouží pro naplnění Custom Adapteru.

---

<sup>8</sup> Iconfinder [online]. 2014 [cit. 2014-05-21]. Dostupné z: <https://www.iconfinder.com/>



Obrázek 23. model\_miniature.xml

### 8.2.3 ColorPickerDialog.java

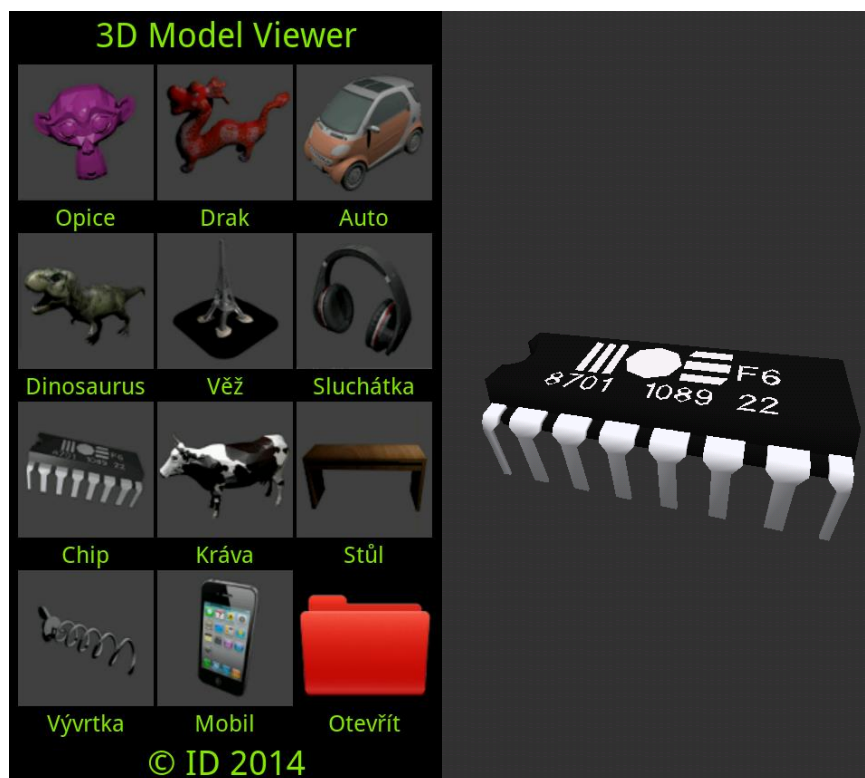
U metody ColorPickerDialog se využilo hotového řešení pro výběr barvy. Barvy jsou v kruhu modelu HSV.



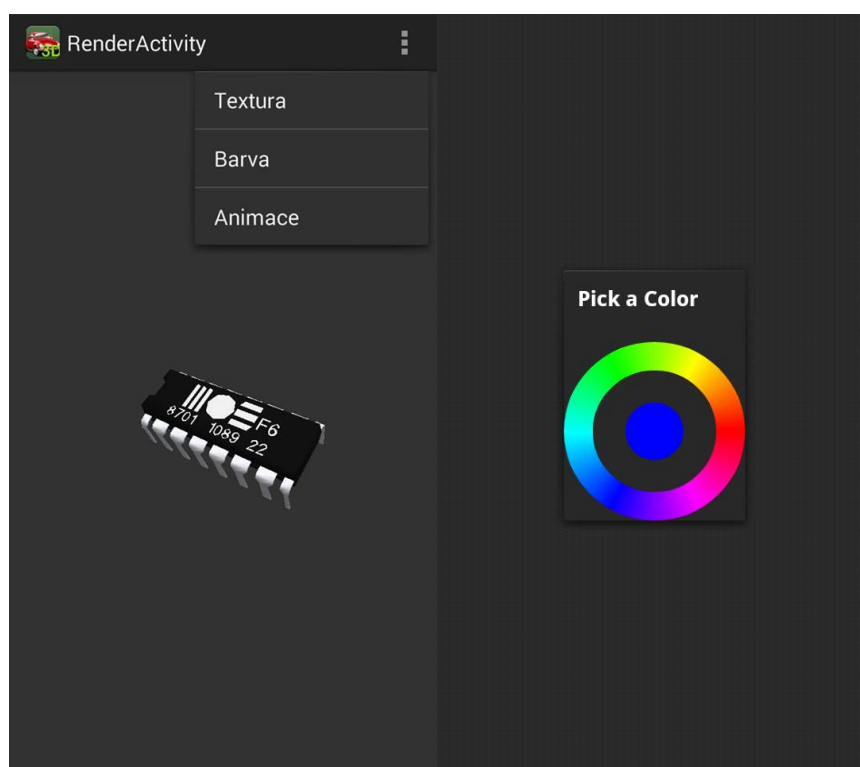
Obrázek 24. Color Picker

### 8.2.4 Výsledná aplikace

Výsledná aplikace je tedy tvořena z hlavní aktivity, která zobrazuje jednotlivé modely v mřížce a uživatel si tak může vybrat z různých předdefinovaných modelů. Po kliknutí na zvolený model se uživateli zobrazí obrazovka s modelem. Tento 3D model si může uživatel libovolně zvětšovat či zmenšovat a otáčet. V menu si pak může zvolit jednu ze tří možností. Texturu, barvu nebo animaci modelu. Po kliknutí na texturu se (znovu) zobrazí textura, po kliknutí na barvu se model zbarví do uživatelem zvolené barvy a v závěru si může model nechat animovat. Model zastaví opětovným kliknutím v nabídce na *animace*.



Obrázek 25. Ukázka výsledné aplikace



Obrázek 26. Ukázka výsledné aplikace

## ZÁVĚR

Cílem této diplomové práce bylo prostudovat využití frameworků pro práci s 3D modely v mobilních aplikacích. Dále tyto frameworky otestovat a srovnat je mezi sebou. Na základě srovnání vhodných frameworků bylo třeba vybrat nejvhodnějšího z nich a s tímto frameworkem pak vytvořit výslednou aplikaci.

Pro testování byli vybráni tři významní kandidáti, kteří využívají OpenGL. Prvním vybraným frameworkem bylo známé Min3D. Jeho vývoj byl však před nějakou dobou pozastaven a byl nahrazen frameworkem Rajawali. Tento framework byl však velkým zklamáním, protože neuspěl snad v žádném testu na výbornou. Byly vytvořeny 3 testovací aplikace, pro každý framework zvlášť. Framework jPCT-AE sice načítal modely pomaleji než Min3D, nicméně dopadl v testu jednoznačně nejlépe.

Výsledná aplikace představuje prohlížeč 3D modelů, podporující formát .3ds. Tento formát byl zvolen úmyslně, protože jeho export a import umožňuje drtivá většina grafických programů a je jednodušší ho implementovat, než .obj + .mtl. V případě importu nového modelu do 3DModelViewer je třeba tento soubor .3ds popř. i s texturou vložit do složky raw a nalinkovat jej v kódu. Také je vhodné mu přiřadit ikonu, aby uživatel viděl hned v GridView o jaký model se jedná. Po dlouhodobější práci s jPCT-AE bylo zjištěno, že práce s více texturami je poměrně složitá záležitost a zatím nebyla vyřešena. Aplikace je tedy schopná pracovat pouze s jednou texturou. Dalším negativním zjištěním bylo načítání objektu z externí paměti. Framework sice 3D model načte bez problému, avšak neumí jej vyrenderovat.

I přes tyto neduhy se podařilo vytvořit aplikaci, která je plně funkční, stabilní a umožňuje zobrazení i složitějších modelů (v závislosti na testovaném HW). Aplikace tedy zobrazuje miniatury jednotlivých 3D modelů v mřížce GridView a po kliknutí na zvolený model se zobrazí. Uživatel si může pohodlně dvě prsty objekt zvětšit či zmenšit, vyvoláním nabídky si může změnit barvu modelu, vrátit mu zpátky texturu a spustit si krátkou animaci modelu.

## ZÁVĚR V ANGLIČTINĚ

The aim of this thesis was to study the use of frameworks for working with 3D models in mobile applications. Furthermore, these frameworks to test and compare them with each other. Based on a comparison of appropriate frameworks were necessary to choose the most suitable one and then create the final application with this framework.

To test were selected three major candidates who use OpenGL. The first selected framework was Min3D. Its development, however, was suspended some time ago and was replaced by framework Rajawali . This framework was very disappointing, because it did not succeed in any test with flying colors. Three test applications were created for each framework separately. Framework jPCT -AE loaded models more slower than Min3D , but hit the test clearly best.

The resulting application is a 3D model viewer that supports .3ds format. This format was chosen deliberately because it allows to export and import the vast majority of graphics programs and it is easier to implement than .obj + .mtl. If case of importing a new model into 3DModelViewer its necessary to put the file .3ds or texture into the raw directory and link to the code.

Also, it is advisable to assign an icon to the user immediately saw the GridView on what model it is. After longer work with jPCT -AE was found to work with multiple textures is quite complicated and has not yet been resolved . The application is therefore able to work with only one texture. Another negative finding was loading an object from external memory. Framework loads 3D model without a problem, but it can not render .

Despite these maladies was created application that is fully functional, stable and allows to display more sophisticated models (depending on the test HW) . So that the application displays a thumbnail of each 3D model in a grid GridView and after click on the selected model is displayed. The user can easily scale or shrink object by two fingers, by calling the menu, user can change model color to return texture back and run a short animation model.

## SEZNAM POUŽITÉ LITERATURY

- [1] Software pro 3D modelování a animace. *Autodesk* [online]. 2014 [cit. 2014-04-12]. Dostupné z: <http://www.autodesk.cz/products/autodesk-3ds-max/overview#>
- [2] KUPSKÝ, Petr. Poser 9: Svět 3D postav pro každého. *Pixel*. 2012, č. 10.
- [3] KUPSKÝ, Petr. ZBrush 4: Staňte se virtuálním sochařem!. *Pixel*. 2013, č. 4.
- [4] HRACHOVINA, Petr. Softimage 2014: Novinky v nové verzi 3D animačního programu. *Pixel*. 2013, č. 7-8.
- [5] *Autodesk* [online]. 2014 [cit. 2014-04-12]. Dostupné z: <http://www.autodesk.com/>.
- [6] About LightWave. *Lightwave 3D* [online]. 2014 [cit. 2014-04-12]. Dostupné z: <https://www.lightwave3d.com/overview/>.
- [7] Poser 10. *64bitprogramlar* [online]. 2014 [cit. 2014-04-12]. Dostupné z: <http://www.64bitprogramlar.com/poser-10-39158.html>
- [8] The software used in the making of Avatar. *Twin pixels* [online]. 2010 [cit. 2014-04-12]. Dostupné z: <http://www.twin-pixels.com/software-used-making-of-avatar/>
- [9] Finding Nemo - Shading a Coral Reef. BERNARDI, Chris. PIXAR ANIMATION STUDIOS. *Pixar* [online]. 2014 [cit. 2014-04-12]. Dostupné z: <http://renderman.pixar.com/view/finding-nemo-shading-a-coral-reef>
- [10] *Pixar* [online]. 2012 [cit. 2014-04-12]. Dostupné z: <http://www.pixar.com/>
- [11] Informace o firmě. *Cinema 4D* [online]. 2014 [cit. 2014-04-12]. Dostupné z: <http://www.cinema4d.cz/firma/>
- [12] Nejlepší bezplatný program pro 3D modelování. *Živě* [online]. 2012 [cit. 2014-04-12]. Dostupné z: <http://www.zive.cz/clanky/nejlepsi-bezplatny-program-pro-3d-modelovani/sc-3-a-163845/>
- [13] *SketchUp* [online]. 2014 [cit. 2014-04-14]. Dostupné z: <http://www.sketchup.com/>
- [14] *Blender* [online]. 2014 [cit. 2014-04-14]. Dostupné z: <http://www.blender.org/>
- [15] POKORNÝ, Pavel. *Blender: naučte se 3D grafiku*. 2., aktualiz. a rozš. vyd. Praha: BEN - technická literatura, 2009, 286 s. ISBN 978-80-7300-244-2.
- [16] Google SketchUp 14.0.4900. *Pc world* [online]. 2014 [cit. 2014-04-14]. Dostupné z: <http://www.pcworld.pl/ftp/google.sketchup.html>



- [17] Lagoa : 云上的 3D. *Ifanr* [online]. 2013 [cit. 2014-04-16]. Dostupné z: <http://www.ifanr.com/275808>
- [18] Wacom SmudgeGuard. *Alza.cz* [online]. 2014 [cit. 2014-04-18]. Dostupné z: <http://www.alza.cz/wacom-smudgeguard-d576701.htm>
- [19] Grafické tablety Wacom. *Heureka* [online]. 2014 [cit. 2014-04-18]. Dostupné z: <http://graficke-tablety.heureka.cz/f:2860:6051/?o=5>
- [20] Wacom Cintiq 22HD. *Wacom* [online]. 2014 [cit. 2014-04-18]. Dostupné z: <http://us.wacom.com/en/shared/products/pen-displays/cintiq/cintiq-22hd-touch/>
- [21] Leap Motion - Kouzla mávnutím ruky. *Alza.cz* [online]. 2014 [cit. 2014-04-18]. Dostupné z: <http://www.alza.cz/leap-motion---kouzla-mavnutim-ruky-art8400.htm>
- [22] JANEK, Leoš. Lagoa: On-line 3D aplikace pro všechny. *Pixel*. 2013, č. 12.
- [23] Genymotion 2.1. *Chip* [online]. 2014 [cit. 2014-04-18]. Dostupné z: <http://www.chip.cz/casopis-chip/plne-verze/genymotion-2-1/>
- [24] *Genymotion* [online]. 2014 [cit. 2014-04-18]. Dostupné z: <http://www.genymotion.com/>
- [25] Run Android on Your Desktop With Genymotion Android Emulator. *Maketecheasier* [online]. 2013 [cit. 2014-04-18]. Dostupné z: <http://www.maketecheasier.com/run-android-on-desktop-with-genymotion-android-emulator/>
- [26] Genymotion. *Softonic* [online]. 2013 [cit. 2014-04-18]. Dostupné z: <http://genymotion.en.softonic.com/>
- [27] About the Eclipse Foundation. *Eclipse* [online]. 2014 [cit. 2014-04-18]. Dostupné z: <https://www.eclipse.org/org/>
- [28] JACKSON, Wallace. *Android apps for absolute beginners* [online]. 2nd edition. 2011, 378 pages [cit. 2014-04-18]. ISBN 978-1-4302-4788-3. Dostupné z: [http://books.google.cz/books/about/Android\\_Apps\\_for\\_Absolute\\_Beginners.html?id=GREVbtctx8sC&redir\\_esc=y](http://books.google.cz/books/about/Android_Apps_for_Absolute_Beginners.html?id=GREVbtctx8sC&redir_esc=y)
- [29] Browsing Eclipse Java Perspectives. *State of Flow* [online]. 2007 [cit. 2014-04-19]. Dostupné z: <http://www.stateofflow.com/journal/65/browsing-eclipse-java-perspectives>

- [30] Exploring the SDK. *Developers* [online]. 2014 [cit. 2014-04-19]. Dostupné z: <http://developer.android.com/sdk/exploring.html>
- [31] Android, the world's most popular mobile platform. *Developers* [online]. 2014 [cit. 2014-04-19]. Dostupné z: <http://developer.android.com/about/index.html>
- [32] Gartner: Android je světově nejpobulárnější operační systém. Jeho dominance se zvyšuje vstupem na trh počítačů. *Marketing journal.cz* [online]. 2014 [cit. 2014-04-19]. Dostupné z: [http://www.m-journal.cz/cs/aktuality/gartner--android-je-svetove-nejpopularnejsi-operacni-system--jeho-dominance-se-zvysuje-vstupem-na-trh-pocitacu\\_\\_s288x10315.html](http://www.m-journal.cz/cs/aktuality/gartner--android-je-svetove-nejpopularnejsi-operacni-system--jeho-dominance-se-zvysuje-vstupem-na-trh-pocitacu__s288x10315.html)
- [33] Krátké ohlédnutí za historií Androidu. *Svět Androida* [online]. 2013 [cit. 2014-04-19]. Dostupné z: <http://www.svetandroida.cz/kratke-ohlednuti-za-historii-androidu-201305>
- [34] Android 4.3 Jelly Bean: přibily užitečné drobnosti. *Mobilmania.cz* [online]. 2013 [cit. 2014-04-19]. Dostupné z: <http://www.mobilmania.cz/clanky/android-43-jelly-bean-pribily-uzitecne-drobnosti/sc-3-a-1324334/default.aspx>
- [35] Android 4.4 KitKat – podrobný přehled novinek obrazem i slovem. *Svět Androida* [online]. 2013 [cit. 2014-04-19]. Dostupné z: <http://www.svetandroida.cz/android-4-4-kitkat-podrobny-prehled-novinek-obrazem-i-slovem-201311>
- [36] Co je to ten Android?. *Android aplikace* [online]. 2014 [cit. 2014-04-19]. Dostupné z: <http://androidaplikace.cz/index.php/co-je-operacni-system-android/>
- [37] Android Architecture. *Android NoteBook* [online]. 2014 [cit. 2014-04-19]. Dostupné z: <http://asishinwp.wordpress.com/training/architecture/>
- [38] Android 5 release date, news and rumors. *Techradar.phones* [online]. 2014 [cit. 2014-04-27]. Dostupné z: <http://www.techradar.com/news/phone-and-communications/mobile-phones/android-5-0-key-lime-pie-release-date-news-and-rumours-1091500>
- [39] *Web & Mobile Marketing Apps Enhanced with 3D / 3DVIA Cloud* [online]. 2014 [cit. 2014-04-27]. Dostupné z: <http://www.3dvia.com/>
- [40] Open Source Android Game Engines. *Mobile Game Engine* [online]. 2014 [cit. 2014-04-27]. Dostupné z: [http://mobilegameengines.com/android/open\\_source\\_game\\_engines](http://mobilegameengines.com/android/open_source_game_engines)

- [41] Min3D. *A 3d library/framework for Android using Java and OpenGL ES - Google Project Hosting* [online]. 2014 [cit. 2014-04-28]. Dostupné z: <http://code.google.com/p/min3d/>
- [42] Cloning animated 3D objects (min3D framework for Android). *Rozengain* [online]. 2010 [cit. 2014-04-28]. Dostupné z: <http://www.rozengain.com/blog/2010/05/26/cloning-animated-3d-objects-min3d-framework-for-android/>
- [43] Loading 3D models with the min3D framework for Android. *Rozengain* [online]. 2010 [cit. 2014-04-28]. Dostupné z: <http://www.rozengain.com/blog/2010/05/17/loading-3d-models-with-the-min3d-framework-for-android/>
- [44] Cloning animated 3D objects (min3D framework for Android). *Rozengain* [online]. 2010 [cit. 2014-04-28]. Dostupné z: <http://www.rozengain.com/blog/2010/05/26/cloning-animated-3d-objects-min3d-framework-for-android/>
- [45] MasDennis/Rajawali. *GitHub* [online]. 2013 [cit. 2014-04-28]. Dostupné z: <https://github.com/MasDennis/Rajawali>
- [46] Dwarf-fw. *Dwarf: Android 3D Framework - Google Project Hosting* [online]. 2010 [cit. 2014-04-28]. Dostupné z: <http://code.google.com/p/dwarf-fw/>
- [47] Announcing Rajawali: An OpenGL ES 2.0 Based 3D Framework For Android. *Rozengain.com/Blog* [online]. 2011 [cit. 2014-05-08]. Dostupné z: <http://www.rozengain.com/blog/2011/08/23/announcing-rajawali-an-opengl-es-2-0-based-3d-framework-for-android/>
- [48] *JPCT 3D engine* [online]. 2014 [cit. 2014-05-08]. Dostupné z: <http://www.jpct.net/>
- [49] *LibGDX* [online]. 2014 [cit. 2014-05-08]. Dostupné z: <http://libgdx.badlogicgames.com/>
- [50] Min3D framework examples. *Google play* [online]. 2011 [cit. 2014-05-08]. Dostupné z: <https://play.google.com/store/apps/details?id=min3d.sampleProject1>
- [51] Rajawali 3D Engine Examples. *Google play* [online]. 2013 [cit. 2014-05-08]. Dostupné z: <https://play.google.com/store/apps/details?id=com.monyetmabuk.rajawali.tutorials&hl=cs>

- [52] Libgdx. *GitHub* [online]. 2014 [cit. 2014-05-08]. Dostupné z: <https://github.com/libgdx/libgdx>
- [53] 안드로이드 jPCT-AE로 OpenGL ES 3D 데모 개발. *Jidolstar* [online]. 2012 [cit. 2014-05-08]. Dostupné z: <http://blog.jidolstar.com/824>
- [54] *JPCT 3D engine* [online]. 2014 [cit. 2014-05-08]. Dostupné z: <http://www.jpct.net/>
- [55] Vyvíjíme pro Android – tvoříme aktivity. *ABC Linuxu* [online]. 2011 [cit. 2014-05-22]. Dostupné z: <http://www.zdrojak.cz/clanky/vyvijime-pro-android-dialogy-a-activity/>

**SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK**

API	Application Programming Interface
DVM	Dalvik Virtual Machine
GPS	Global Positioning System
HTML	HyperText Markup Language
IDE	Integrated Development Environment
JVM	Java Virtual Machine
LWJGL	The Lightweight Java Game Library
PDA	Personal Digital Assistant
SDK	Software Development Kit
VBO	Vertex Buffer Object

**SEZNAM OBRÁZKŮ**

Obrázek 1. Ukázka vývojového prostředí Eclipse [29] .....	11
Obrázek 2. Android architektura [37] .....	15
Obrázek 3. Životní cyklus aktivity v Androidu [55] .....	19
Obrázek 4. Genymotion [25] .....	20
Obrázek 5. Ukázka tabletu Wacom Cintiq 22HD [20] .....	21
Obrázek 6. Ukázka Leap Motion [21] .....	22
Obrázek 7. Poser 10 ukázka postavy [7] .....	24
Obrázek 8. Ukázka prostředí v programu Blender [14] .....	25
Obrázek 9. Ukázka prostředí v programu SketchUp [16] .....	26
Obrázek 10. Ukázka prostředí v programu Lagoa [17] .....	27
Obrázek 11. Min3D [50] .....	30
Obrázek 12. Rajawali [51] .....	31
Obrázek 13. jPCT-AE [53] .....	32
Obrázek 14. libGDX [52] .....	34
Obrázek 15. Ukázka textury draka .....	37
Obrázek 16. Test č. 1 .....	38
Obrázek 17. Test č. 2 .....	40
Obrázek 18. Test č. 3 .....	42
Obrázek 19. Test č. 4 .....	44
Obrázek 20. activity_main.xml .....	48
Obrázek 21. file_dialog.xml .....	48
Obrázek 22. Menu aplikace .....	51
Obrázek 23. model_miniaature.xml .....	52
Obrázek 24. Color Picker .....	52
Obrázek 25. Ukázka výsledné aplikace .....	53
Obrázek 26. Ukázka výsledné aplikace .....	53

**SEZNAM TABULEK**

Tabulka 1. Dostupné balíčky pro Android SDK [30] .....	12
Tabulka 2. Model 1 .....	37
Tabulka 3. Srovnání frameworků pro Model 1 .....	38
Tabulka 4. Model 2 .....	39
Tabulka 5. Srovnání frameworků pro Model 2 .....	40
Tabulka 6. Model 3 .....	41
Tabulka 7. Srovnání frameworků pro Model 3 .....	41
Tabulka 8. Model 4 .....	43
Tabulka 9. Srovnání frameworků pro Model 4 .....	43

## SEZNAM PŘÍLOH

PI      CD se zdrojovými kódy a použitými modely