

**Česká zemědělská univerzita v Praze**

**Provozně ekonomická fakulta**

**Katedra informačních technologií**



**Diplomová práce**

**Aplikace Android – teorie a praxe**

**Tomáš Jirsa**

© 2017 ČZU v Praze

## ZADÁNÍ DIPLOMOVÉ PRÁCE

Bc. Tomáš Jirsa

Informatika

Název práce

**Aplikace Android – teorie a praxe**

Název anglicky

**Android Applications – Theory and Practice**

---

### Cíle práce

Diplomová práce je tématicky zaměřena na problematiku aplikací pro operační systém Android od společnosti Google. Hlavním cílem práce je charakteristika vývoje aplikací, jejich distribuce a uplatnění na trhu.

Dílní cíle diplomové práce jsou:

- analyzovat obecné požadavky na aplikace pro operační systém Android
- charakterizovat různé pohledy na využití operačního systému Android a vyzkoušet některé aplikace v praxi

### Metodika

Metodika řešené problematiky diplomové práce je založena na studiu a analýze odborných informačních zdrojů. Vlastní řešení je realizováno formou návrhu a implementace vlastní jednoduché aplikace pro operační systém Android. Na základě syntézy teoretických poznatků a výsledků vlastního řešení budou formulovány závěry diplomové práce.

## Doporučený rozsah práce

60-80 stran

## Klíčová slova

Android, mobilní operační systém, Java, smartphone, Google, Android SDK, Android studio, Unity

---

## Doporučené zdroje informací

Android <http://developer.android.com/guide/index.htm>

Android SDK for Windows – <http://code.google.com/android/documentation.html>

MEIER R. Professional Android Application Development, 2. vydání. Indianapolis: Wrox, 2010, 576 str., ISBN 0470565527.

MURPHY, Mark L. Beginning Android 2. 1. vydání. New York: Springer Verlag 2010. 416s. ISBN 978-1-4302-2629-1.

ROGERS, R., LOMBARDO, J. Android Application Development, 1st Edition. Cambridge: O'Reilly Media, Inc. 2009, 336s., ISBN 978-0-596-52147-9.

Symbian Developer: home – <http://www.symbian.com/developer/index.html>

TOPLEY, Kim. J2ME v kostce – Pohotová referenční příručka. Praha: GRADA, 2004, 536s., ISBN 80-247-0426-9.

Windows Mobile Developer Center – <http://msdn2.microsoG.com/en-us/windowsmobile/default.aspx>

---

## Předběžný termín obhajoby

2016/17 LS – PEF

## Vedoucí práce

Ing. Čestmír Halbich, CSc.

## Garantující pracoviště

Katedra informačních technologií

Elektronicky schváleno dne 18. 10. 2016

**Ing. Jiří Vaněk, Ph.D.**

Vedoucí katedry

Elektronicky schváleno dne 24. 10. 2016

**Ing. Martin Pelikán, Ph.D.**

Děkan

V Praze dne 16. 03. 2017

### **Čestné prohlášení**

Prohlašuji, že svou diplomovou práci "Aplikace Android – teorie a praxe" jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou citovány v práci a uvedeny v seznamu použitých zdrojů na konci práce. Jako autor uvedené diplomové práce dále prohlašuji, že jsem v souvislosti s jejím vytvořením neporušil autorská práva třetích osob.

V Praze dne 31.3.2017

---

### **Poděkování**

Rád bych touto cestou poděkoval vedoucímu své diplomové práce Ing. Čestmíru Halbichovi, CSc., za odborné vedení a cenné rady při psaní této diplomové práce.

# Aplikace Android – teorie a praxe

## Souhrn

Tato diplomová práce je zaměřena na mobilní operační systém Android a především vývoj aplikací pro tuto platformu. Teoretická část je strukturována od obecných poznatků o mobilních operačních systémech až po kapitoly primárně se zabývající OS Android, kde je představena historie platformy, architektura samotného systému a v neposlední řadě také změny, které přinesly jednotlivé vývojové verze. Obsahem posledních kapitol teoretické části je představení hlavních konkurentů na trhu s operačními systémy pro mobilní zařízení a typických vývojových prostředí pro vytvoření aplikace pro OS Android.

Praktická část se zaměřuje na vytvoření jednoduchého prototypu aplikace v oficiálním vývojovém prostředí Android Studia. Podrobně je zde popsán postupný vývoj aplikace za využití znalostí z teoretické části a vývojového prostředí představeného v předešlých kapitolách.

**Klíčová slova:** Android, mobilní operační systém, Java, smartphone, Google, Android SDK, Android Studio, Unity

# **Application Android – theory and practice**

## **Summary**

This diploma thesis is focused on the mobile operating system Android, and particularly the development of applications for the platform. The theoretical part is structured from general knowledge of mobile operating systems to chapters dealing primarily with the Android OS, which introduces the history of the platform, the architecture of the system itself, and last but not least the changes that each individual version of the system brought. The final chapters of the theoretical part are introducing the main competitors on the market for operating systems for mobile devices and typical integrated development environments for creating applications for the Android OS.

The practical part is focused on creating a simple prototype of application in the official Android Studio development environment. It includes detailed development of application using knowledge from the theoretical part and one of the earlier mentioned development environments.

**Keywords:** Android, mobile operating system, Java, smartphone, Google, Android SDK, Android Studio, Unity

# Obsah

<b>1 Úvod.....</b>	<b>11</b>
<b>2 Cíl práce a metodika .....</b>	<b>12</b>
2.1 Cíl práce .....	12
2.2 Metodika .....	12
<b>3 Teoretická východiska .....</b>	<b>13</b>
3.1 Vznik a historie mobilních OS .....	13
3.1.1 První počátky .....	14
3.1.2 Od historie k současnosti .....	15
3.2 Android .....	20
3.2.1 Historie.....	21
3.2.2 Vývoj systému .....	23
3.2.2.1 Betaverze systému .....	24
3.2.3 Vývojové verze .....	24
3.2.3.1 Android 1.X.....	25
3.2.3.2 Android 2.X.....	26
3.2.3.3 Android 3.X.....	28
3.2.3.4 Android 4.X.....	28
3.2.3.5 Android 5.X.....	30
3.2.3.6 Android 6.X.....	30
3.2.3.7 Android 7.X.....	31
3.2.3.8 Zastoupení jednotlivých verzí na trhu .....	31
3.2.4 Architektura .....	32
3.2.4.1 Zavaděč.....	33
3.2.4.2 Linux Kernel.....	34
3.2.4.3 Knihovny .....	34
3.2.4.4 Android Runtime .....	35
3.2.4.5 Aplikační rámec.....	36
3.2.4.6 Aplikace.....	37
3.2.5 Distribuce.....	37
3.2.5.1 Google Play .....	38
3.2.6 Bezpečnost.....	40



3.3	Aplikace .....	41
3.3.1	Android Manifest .....	42
3.3.2	Activity .....	42
3.3.3	Services .....	44
3.3.4	Intent .....	47
3.3.5	Content providers .....	48
3.3.6	Broadcast Receivers .....	49
3.3.7	Fragmenty .....	49
3.4	Vývoj na Android .....	52
3.4.1	Programovací jazyk .....	52
3.4.2	Vývojové nástroje .....	52
3.4.2.1	Eclipse .....	54
3.4.2.2	AppInventor .....	55
3.4.2.3	Android Studio .....	56
3.5	Konkurence .....	57
3.5.1	iOS .....	57
3.5.2	Windows Phone .....	59
3.5.3	Porovnání OS Android s konkurencí .....	60
3.5.3.1	Kritéria .....	60
3.5.3.2	Metoda pořadí .....	61
3.5.3.3	Metoda stejné váhy .....	61
3.6	Využití OS Android mezi různými druhy zařízeními .....	62
3.6.1	Aplikace .....	62
3.6.2	Android Wear .....	64
3.6.3	Televize .....	64
3.6.4	Ostatní elektronika .....	65
<b>4</b>	<b>Vlastní práce .....</b>	<b>66</b>
4.1	Aplikace počasí .....	66
4.2	Grafický vzhled .....	67
4.3	Manifest .....	68
4.4	Použité API rozhraní .....	69
4.5	Main Activity .....	70
4.6	Výsledná aplikace .....	71
<b>5</b>	<b>Výsledky a diskuse .....</b>	<b>72</b>
<b>6</b>	<b>Závěr .....</b>	<b>74</b>

<b>7 Seznam použitých zdrojů .....</b>	<b>75</b>
<b>8 Přílohy .....</b>	<b>79</b>
<b>9 Seznam zkratk .....</b>	<b>79</b>

## **Seznam obrázků**

Obrázek 1 - Evoluce ve vzhledu telefonů .....	13
Obrázek 2 - IBM Simon.....	14
Obrázek 3 - MDA T-mobile .....	16
Obrázek 4 - Nokia 5800XpressMusic.....	18
Obrázek 5 - Prodaná mobilní zařízení , rozdělení dle OS .....	20
Obrázek 6 - Někteří členové Open Headset Alliance .....	22
Obrázek 7 - HTC Dream ( T-mobile G1) .....	22
Obrázek 8 - Data vydání jednotlivých verzí a jejich loga.....	25
Obrázek 9 - Vzhled zamknuté a domácí obrazovky verze 2.0 .....	27
Obrázek 10 - Zastoupení verzí systému mezi uživateli .....	32
Obrázek 11 - Znárodnění architektury OS Android.....	33
Obrázek 12 - Vnitřní architektura Kernel .....	34
Obrázek 13 - Průběh kompilace Java kódu .....	36
Obrázek 14 - Vzhled Google Play .....	39
Obrázek 15 - Životní cyklus Activity .....	43
Obrázek 16 - Životní cyklus Services.....	45
Obrázek 17 - Příklad implicitního záměru.....	48
Obrázek 18 - Využití fragmenů na tabletu a mobilním zařízení.....	50
Obrázek 19 - Porovnání životního cyklu aktivity a fragmentu.....	51
Obrázek 20 - Ukázka AppInventor .....	55
Obrázek 21 - První vs. poslední iPhone.....	58
Obrázek 22 - Ukázka vzhledu grafického rozhraní Windows (dlaždice).....	59
Obrázek 23 - Vzhled aplikace záchranka .....	63
Obrázek 24- První okno po spuštění programu .....	66
Obrázek 25 - XML zobrazení .....	67
Obrázek 26 - Grafické znázornění rozložení prvků.....	68
Obrázek 27 – Manifest.....	69
Obrázek 28 - Nabídka API rozhraní .....	69
Obrázek 29 - Main Activity .....	70
Obrázek 30 - Výsledný vzhled aplikace .....	71

## **Seznam tabulek**

Tabulka 1 - Tabulka minimálních požadavků .....	56
Tabulka 2 - Metoda pořadí.....	61
Tabulka 3 - Metoda stejné váhy.....	62

# 1 Úvod

V současné době dochází k velkému rozkvětu moderních technologií, tento posun v technice směrem vpřed se nevyhnul ani mobilním zařízením. Mobilní telefon už nadále není vnímán, jak název sám napovídá, pouze jako pomocník pro kontakt s okolím, ale v důsledku rozšíření možností využití se mobilní zařízení stává nástrojem zábavy a v některých případech i podnikání a je tak běžnou součástí života každého člověka. Nabídka výrobců chytrých telefonů se během uplynulých let rozrostla a v důsledku snižování cen postupně vytlačují z trhu běžné mobilní telefony.

Operační systém Android přišel na trh s velkým potenciálem úspěš, především z důvodu, kterým se odlišoval od své konkurence. Dorazil s tzv. platformou open source a umožnil tak každému, kdo dostatečně ovládal schopnost programování v daném jazyce, navrhnout jakoukoliv aplikaci na nejnovější zařízení, která dorazila na trh. Především tato výhoda stála za strmým vzrůstem popularity operačního systému Android, který byl nainstalován na více než jedné miliardě zařízení, kdy se systém v jednotlivých telefonech výrobců odlišuje pouze drobnými úpravami v podobě vzhledu. Obrovský úspěch zaznamenal i ze strany vývojářů a během prvních čtyř let od představení bylo v online obchodě Google Play zaznamenáno přes sedmdesát jedna tisíc registrací vývojářských účtů. Samotná společnost Google nabízí podporu začínajícím vývojářům skrze volně stažitelné nástroje pro tvorbu aplikací, které následně mohou využít nejen ve svých mobilních zařízeních, ale v dnešní době i v hodinkách, televizích a některých domácích spotřebičích označených mezi širokou veřejností jako bílá elektronika.

Tato práce je primárně určena pro ty, kteří chtějí poznat operační systém Android nejen z pohledu běžného uživatele, ale zajímají se o svět moderních technologií a především o příčiny, proč je operační systém Android dnes nejrozšířenějším na trhu s telefony a jak takové běžné vytvoření aplikace a následná její distribuce vypadá v reálném světě.

## **2 Cíl práce a metodika**

### **2.1 Cíl práce**

Diplomová práce je tematicky zaměřena na problematiku vývoje aplikací pro mobilní operační systém Android od společnosti Google. Hlavním cílem práce je charakteristika vývoje aplikací, jejich následná distribuce a uplatnění na trhu.

Mezi dílčí cíle diplomové práce patří analyzování obecných požadavků na aplikace pro operační systém Android a charakterizování různých pohledů na využití operačního systému Android, mimo oblast mobilních zařízení a vyzkoušet některé aplikace v praxi.

### **2.2 Metodika**

Metodika řešení problematiky diplomové práce je založena na studiu a analýze odborných informačních zdrojů a získaných zkušenostech při práci s operačním systémem Android. Nedostatečná nabídka knižních zdrojů psaných v českém jazyce způsobila, oprávněnost čerpání většího množství informací z internetových stránek a knižních publikací zahraničních států.

Vlastní řešení je realizováno formou návrhu a implementace vlastní jednoduché aplikace pro operační systém Android vytvořené ve vývojové prostředí představeného v teoretické části.

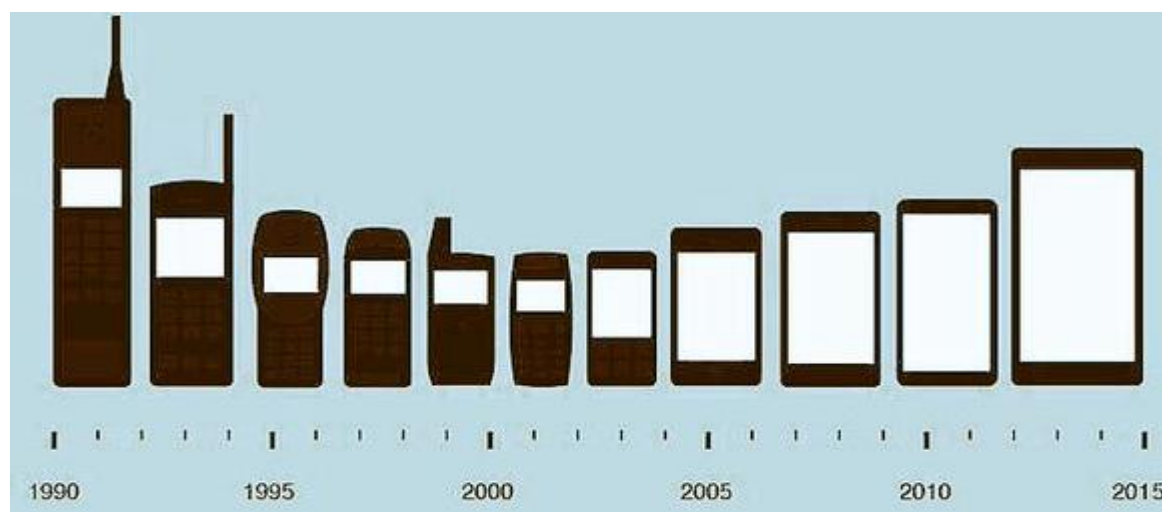
Na základě syntézy teoretických poznatků, vlastních praktických zkušeností získaných při vytváření řešení a samotného výsledků řešení budou formulovány závěry diplomové práce.

## 3 Teoretická východiska

### 3.1 Vznik a historie mobilních OS

Není to tak dlouho, kdy mobilní telefony, označovány dnes také jako smartphony, spatřily světlo světa. Na začátku uvedení na trh využívalo tato zařízení pouze hrstka lidí, která dokázala ocenit a efektivně využít všechny vlastnosti, které telefon oproti běžným mobilním zařízením nabízel. Pro běžného uživatele bylo cenově výhodnější pořídit si klasický mobilní telefon s tzv. uzavřeným operačním systémem, kdy funkce telefonu nebylo možné dále rozšiřovat. Právě rozšíření funkčnosti mobilního zařízení, které do té doby sloužilo pouze ke komunikaci, ve smyslu psaní sms zpráv a telefonování, se postaral příchod otevřených operačních systémů pomocí aplikací, které obyčejný mobilní telefon transformovaly po praktické části využití do téměř domácí kanceláře. Rozšiřujícími funkcí byla celá řada, od e-mailového klienta až po přehrávač medií. [36, 37]

Celý nápad chytrých telefonů byl lákavý, ale právě cena byla jedním z velkých problémů, které brzdily rozmach této technologie. Chytré mobilní telefony potřebovaly pro svůj bezproblémový chod aplikací vyšší výkon, než průměrné přístroje. Vyšší výkon znamenal v důsledku vyšší spotřebu baterie a i když rozměry telefonu byly přizpůsobeny ke konstrukci větší baterie, stále to nestačilo k udržení chytrého telefonu v zapnutém stavu alespoň na jeden den. Postupem času se však tyto nevýhody podařilo odstranit včetně velké nestabilitosti některých integrovaných aplikací, což ve finále vedlo k velkému rozšíření telefonů tohoto druhu. [37]



Obrázek 1 - Evoluce ve vzhledu telefonů  
Zdroj: [16]

### 3.1.1 První počátky

Je to téměř 24 let, přesněji rok 1993, kdy společnost IBM uvedla na trh vůbec první telefon s vlastním operačním systémem. Telefon byl představen pod jménem IBM Simon a obsahoval základní funkce jako kalendář, kalkulačka, blok na poznámky, e-mailový klient či kontakty. Stále zbývá ve výčtu funkcí jedno průlomové řešení se kterým také přišel IBM Simon a tím byl displej, ale ne jen tak obyčejný displej, jednalo se o černobílou dotykovou obrazovku. Společnost IBM si byla vědoma, že tomuto řešení nemůže žádná jiná společnost momentálně konkurovat, tak si svůj přístroj také náležitě ocenila na tu dobu astronomickou částkou 900 amerických dolarů. [38]

Dalším důležitým průkopníkem technologie chytrých telefonů byl Nokia 9000 Communicator od stejnojmenné společnosti Nokia uvedený v roce 1996. Technické parametry informovali o vybavenosti procesorem Intel s frekvencí 24 MHz a paměti telefonu 8 MB, softwarem vloženým do skoro 400 g těla telefonu byl 16 bitový OS Geos.



**Obrázek 2 - IBM Simon**  
Zdroj: [17]

Samotné telefony a s nimi ruku v ruce i operační systémy prošly za ty roky znatelným vývojem. Hardwarová vybavenost tehdejších počítačů by dnes nestačila na průměrný chytrý telefon. Výrobci se snaží o větší přehlednost i zjednodušování při zachování stejné nabídky funkcí za cílem přiblížit chytré telefony i starším generacím. Dříve vysoká cena telefonů určující, že majitelem musel být někdo z vyšších vrstev také už dnes není pravidlem. Cena chytrých telefonů stále klesá a to je jedním z hlavních faktorů jeho masivního rozšíření. [37, 38]

Při porovnání trhu jenom na Americkém kontinentě, deklarují výsledky vzrůstající popularitu smartphone telefonů, kdy z údajů pro rok 2012 bylo zřejmé, že na trhu je chytrých telefonů stále méně než těch „hloupých“, vzhledem k výsledkům 60 % vs. 40 %. avšak v dnešní době se již mluví o nárůstu okolo 65 % ve prospěch chytrých telefonů.

Ve skutečnosti ale, ani v jedné z firem se o těchto zařízeních nemluvílo jako o chytrých telefonech. S takovým označením přišla až firma Ericsson v roce 1997 a zařízením označeným jako model GS88. Bohužel šlo pouze o prototyp a ve finále u prototypu také firma zůstala. [36, 37, 38]

### **3.1.2 Od historie k současnosti**

Psal se rok 1997 a firma Nokia uvedla na trh druhou generaci přístroje 9000 a definitivně tak potvrdila svou roli v tomto odvětví a vyslala tak zprávu pro konkurenci, že se s ní musí počítat. Druhá generace označená jako 9110, byla menších rozměrů a také lehčí než předešlý model, rozlišení displeje zůstalo totožné, stejně tak jeho černobílé zobrazení grafických prvků, procesor měl vyšší frekvenci o taktu 33 MHz.

V dnešní době je firma Qualcomm známá, jako jeden z předních dodavatelů procesorů do chytrých telefonů, nicméně i tato firma přispěla k evoluci mobilních telefonů svým dílem. V roce 1999 vydala stejnojmenná firma smartphone s označením pdQ 1900, určeného výhradně sítím, v České republice ne příliš známého operátora Sprint. Jednalo se o první smartphone s operačním systémem Palm OS, který se do této doby distribuoval pouze prostřednictvím PDA, malých kapesních počítačů. [36]

S novým tisíciletím přicházely výrobci s novými nápady a v souvislosti s tím i novými přístroji. Byl představen první chytrý telefon s operačním systémem od firmy Microsoft. I když je důležité podotknout, že se nejednalo o čistokrevný smartphone, ale v základu o PDA značky Mistsubishi s procesorem 166 Mhz. Začátkem nového tisíciletí využila i značka Sony Ericsson k představení nového operačního systému Symbian, který použila ve svém zařízení R380. Příležitosti se dostalo i značce Motorola s mobilním telefonem A6188, který byl průkopníkem mezi přístroji s linuxovým operačním systémem. Zařízení příliš světového úspěchu nezaznamenal, svou přízeň si však získal v Číně, z důvodu schopnosti rozpoznávání znaků psaných ručně.

Vzhledem k rychlému růstu technologií, zákazníci netrpělivě čekali, kdy přijde na trh první mobilní telefon s barevným displejem. Tento milník si mohla zapsat do historie své firmy značka Nokia s uvedením třetí generace zařízení 9210 v roce 2000, kdy byl mobilní telefon představen a v roce 2001 i uveden na trh, Zbytek konstrukce telefonu zůstal bez viditelnějších změn, důležitá inovace se udála hlavně uvnitř těla telefonu. Změnila se architektura procesoru a také operační systém, kdy se v minulosti jednalo o GEOS, ale s příchodem 9210 byl představen přístroj s operačním systémem Symbian a byl tak po Sony Ericssonu druhým mobilním telefonem vybaveným tímto operačním systémem. Zejména značka Nokia, může operační systém Symbian být vděčný za svou velkou popularitu a v té době i za značnou dominanci na trhu. [37, 38]

Rok 2002 zaznamenal i premiéru dvou značek, kterými byly BlackBerry a Handspring, oba přístroje vybaveny operačním systémem Palm. Závodu o úspěch na poli chytrých telefonů se chtěla zúčastnit i značka HTC, která vstoupila na trh s mobilním telefonem u nás známým jako T-mobile MDA. Tím ovšem rozmach chytrých telefonů v roce 2002 nekončil. Bylo představeno několik příbuzných výše uvedených značek, které se od sebe lišily pouze v detailech, ať už se jednalo o mobilní zařízení Canary, které disponovalo klávesnicí bez dotykového displeje nebo zařízení společnosti HP s názvem Jarnada 928. [36, 38]



**Obrázek 3 - MDA T-mobile**  
Zdroj: [18]



Značky se začaly předhánět, která z nich přinese větší novinku v odvětví chytrých telefonů nebo nastaví novou laťku v podobě vyššího výkonu, než disponuje konkurence. Značka Sony Ericsson přišla s novým grafickým uživatelským rozhraním UIQ, Nokia oproti tomu představila vysouvací model 7650. ovšem bez dotykového prostředí. U modelu 7650 začínalo vycházet najevo, že už se nadále nejedná o složité přístroje určené pouze technicky nadaným lidem, ale o zařízení, které se svou ovladatelností přibližovalo obyčejným mobilním telefonům. [36]

Rok 2003 začal psát novou éru mobilních telefonů s příchodem Andyho Rubina, bývalého zaměstnance firmy Apple a zakladatele společnosti Android Inc, brzkého důležitého hráče na trhu chytrých telefonů popsaného v dalších kapitolách.

Firmy zkoušely prorazit na trh s provedením chytrých telefonů, které by se dalo označit za ne příliš klasické, ať už se jednalo o stylové věčko značky Motorola nebo Nokii N-Gage, zařízení, které připomínalo z většiny herní produkt než mobilní telefon. [36]

Nokia s příchodem roku 2005 už zcela ovládá trh s chytrými telefony a představuje modelovou řadu N. Kromě standardního modelu N70, prezentuje mobilní telefon s označením N90, který umožňoval připojení většího zoomu k fotoaparátu, model N91 zase sloužil jako mobilní přehrávač hudby, kterému umožňoval tuto funkci úložný prostor o velikosti 4 GB. Modelem E61 se oproti tomu firma Nokia snažila vytlačit konkurenci v podobě značky BlackBerry, protože právě tento model disponoval qwerty klávesnicí, která byla do této doby typická pro modely výrobce BlackBerry. Firma HTC se postupně odpojuje jako OEM výrobce a začíná prosazovat více svoji značku mobilních telefonů.

Z kraje roku 2007 byla ukončena celá řada spekulací a zrodila se událost, na kterou každý čekal a která také měla otrást celým světem chytrých telefonů a dominancí firmy Nokia na trhu, bylo tím představení vůbec prvního iPhone od firmy Apple. Při své premiéře Apple zrušil také jeden zažitý zvyk, kterým bylo ovládání dotykového mobilu pomocí stylusu, iPhone se ovládal pouze lidskými gesty. Nokia své postavení neuhájila ani představením modelu N95, který ukázkově reprezentoval kam až se technika mobilních telefonů dokázala dostat. Samotnému modelu mírně uškodily potíže se softwarem, které ze začátku uvedení měl a také to, že se jednalo pouze o bezdotykový displej. Tovární značka HTC odpovídá na uvedení iPhone svým novým modelem Touch s novým uživatelským prostředím Sense. [36, 37, 38]

Rozdělení trhu má pod kontrolou v Evropě a zbytku světa stále ještě firma Nokia, toto pravidlo se netýká pouze Severní Ameriky, kde je ve větším zastoupení chytrý telefon BlackBerry, ačkoliv se využívá primárně v pracovním prostředí, pomalu se začíná přeorientovávat na běžného zákazníka. V případě značky BlackBerry se s vydáním nových modelů nikdy neočekávalo větších pokroků, ale z většiny malých a dobře promyšlených kroků dle idey firmy, výjimku tvoří model Storm z roku 2008 s dotykovým displejem, do této doby disponovaly všechny zařízení pouze qwerty klávesnicí. [38]

Rok po uvedení prvního chytrého telefonu od firmy Apple přichází určitá nastavba ve formě iPhone 3G. S uvedením telefonu na trh přichází i představení obchodu pro aplikace App Store, jedná se o zjednodušení pro uživatele, kde aplikace na zařízení lze snadno najít, protože už nadále nemusí složitě prohledávat různé webové stránky, ale ke stažení aplikací a jejich následnou instalaci slouží jedno centrální místo. Nedlouho po uvedení představuje Android Inc. vlastněný už v té době firmou Google první chytrý telefon T-mobile G1. Samotné zařízení bylo pod taktovkou výroby společnosti HTC, které dosud využívalo služeb Windows Mobile a tak nepřímě přispělo k pomalému ústupu této firmy. Později ve stejném roce spolupráce pokračovala a výsledky byly demonstrovány na modelech Touch s přídávky Diamond, Pro a HD. Nokia připravila model pro přímou konkurenci Applu ve formě 5800. [36]



**Obrázek 4 - Nokia 5800XpressMusic**  
Zdroj: [19]

Ročník 2009 pokračuje v tradici tam, kde předešlý ročník skončil. Představeno bylo plno nových modelů s operačním systémem Android, včetně přístrojů pod taktovkou firmy HTC, modely Hero a Tattoo. Další novinkou bylo nahrazení operačního systému Palm OS systémem webOS v zařízení Palm Pre z důvodu nedostatků, který předešlý systém obsahoval, ze seznamu je možné zmínit absence podpory sítě 3G. V závěru roku přináší firma Nokia špičkový model N900 u kterého tomu také odpovídá cena. [37]

Bylo otázkou času, kdy si i firma Samsung začne uvědomovat důležitost trhu s chytrými telefony a věnuje mu svou pozornost. Představil proto svoje první zařízení s operačním systémem Android, model s názvem i7500 Galaxy, měsíc poté i vlastní operační systém Bada, který byl obsažen do mobilního telefonu Samsung Wave na začátku roku 2010. [37, 38]

Rok 2010 znamenal téměř jistý konec operačního systému Windows Mobile, největší odběratel v podobě firmy HTC přešel na operační systém Android a definitivně to potvrdil i vydáním modelu Google Nexus One. Firma Samsung se začíná rozhlížet po trhu s chytrými telefony a neurčuje si malé cíle, chce sesadit z nejvyšších příček finskou Nokii. Přímou vyzývá i konkurenta Apple, který za krátkou dobu nabyl velké popularity a s velkou noblesou představuje svoji vlajkovou loď, model Galaxy S. Apple do boje zapojil i právníky, kteří tvrdí, že firma Samsung přímo kopíruje jejich model iPhone. [38]

Koncem roku 2010 se stal šéfem Nokie vysoký představitel ve vedení u společnosti Microsoft, vše tak napovídalo spojení těchto dvou firem a možného vzkříšení operačního systému Windows Mobile na zařízeních od Nokie. Ve stejném roce došlo také ke spojení firem HP, výrobce počítačů a firmy Palm. Toto spolenectví ale nepřineslo úspěch. Novinky vydané pod společností HP, ale s operačním systémem webOS od už v té době neexistující firmy Palm se na trhu neujaly. Systém obsahoval několik důležitých softwarových chyb a celý tento projekt byl velmi ztrátový a v roce 2011 ukončila společnost HP své působení na trhu chytrých telefonů a to znamenalo konec i pro Palm a jeho operační systém webOS.

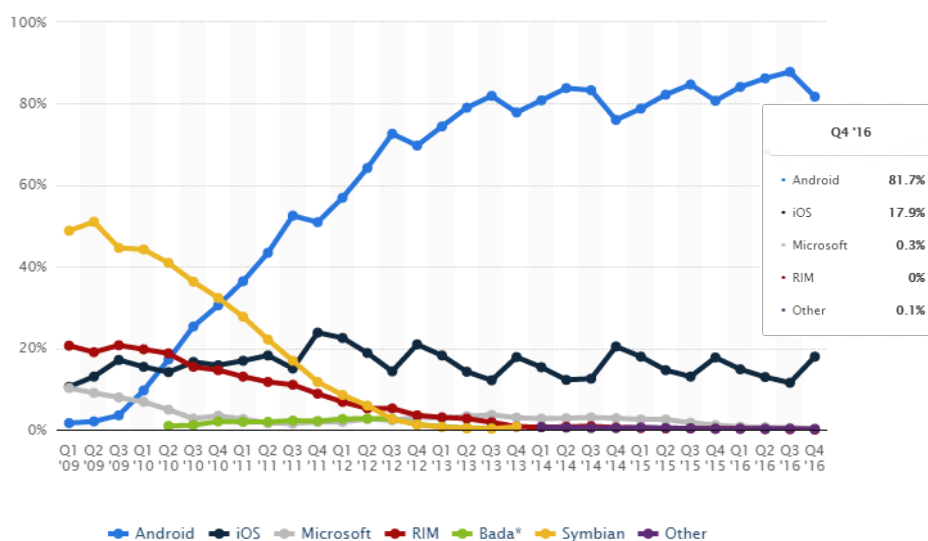
Společnost Nokia v létě 2011 představuje první smartphone bez Symbianu, model N9. Jednalo se spíše o ukázkou ambicí vývojářů, že stále disponují talentem vyrobit kvalitní produkt. Jelikož ještě před uvedením bylo oznámeno právě partnerství se společností Microsoft, bylo zřejmé, že se bude jednat o jediný model a nic podobného následovat nebude. . [38]

Koncem roku jsou představeny nové modely se vzkříšeným operačním systémem Windows Mobile, pojmenovaným nově Windows Phone. Systém se nachází ve všech nových modelových řadách Nokia. Symbian odešel pod poradenskou společnost Accenture a s firmou Nokia nemá nic společného, krom historie. [38]

Od té doby se na trhu příliš situace nezměnila. Na trhu jsou stále stejní hráči a bojuje se o to, kdo bude mít větší podíl na trhu. V současné době jsou hlavními konkurenty Samsung se svým OS Android a Apple se svým iOS, kteří dominují trhu a nikdo jiný toto rozložení zatím nenarušil. Historie ukazuje, že kdokoliv, i ten neúspěšnější na trhu může záhy skončit a tak je budoucnost otevřená i pro všechny potenciální konkurenty v podobě HTC nebo jiných čínských značek vydobývajících si svoje pozice na trhu, včetně pomalu vracející se Nokie, která nedávno představila legendární model 3110 s upraveným softwarovým vybavením a hardwarovým vzhledem, ale stále vycházející ze svého úspěchu před několika lety. [38]

### 3.2 Android

Mobilní operační systém Android je nejvíce užívaný OS na trhu, čítající přes 80% všech vyrobených chytrých telefonů. Otevřená platforma tohoto systému je přístupna všem bez jakýchkoliv poplatků, nízká náročnost na výkon hardwaru umožňuje využití na různá zařízení od mobilních telefonů až po různé druhy domácí elektroniky.



**Obrázek 5 - Prodaná mobilní zařízení , rozdělení dle OS**  
Zdroj: [20]

### 3.2.1 Historie

Historie operačního systému Android se začíná psát v roce 2003, kdy u zrození značky stály Rich Miner, Nick Sears, Chris Whit a nejvíce známý Andy Rubin. Prvotní zájmem bylo vytvoření operačního systému pro digitální fotoaparáty, tento nápad brzy společnost opustila z důvodu malého potenciálu na uchycení, způsobené velikostí trhu. Společnost se poté začala orientovat na mobilní telefony, ale navzdory velkým zkušenostem předních představitelů společnosti s předešlým vedením velkých firem, došly peníze. [1, 4]

V polovině roku 2005 odkoupil skomírající společnost softwarový gigant Google. Jeden z hlavních představitelů původní společnosti, Andy Rubin, byl dosazen do vedení části firmy, která se zabývala mobilní technologií a začal pracovat na operačním systému na linuxovém jádře. Ve třetím čtvrtletí roku 2007 získala firma hned několik patentů na mobilní zařízení a očekávalo se, kdy firma poprvé vstoupí na trh proti konkurentům, kterými byly do té doby pouze Windows Mobile a Symbian. Na začátku listopadu, kdy fanoušci podleli spekulacím a netrpělivě očekávali příchod nového mobilního telefonu od Googlu, vydala firma tiskové vyjádření prostřednictvím Andyho Rubina, které popírá všechny spekulace o vydání nového telefonu, dále připojuje oznámení, které nikdo nečekal, a tím bylo vytvoření určité skupiny, která bude sdružovat různé zainteresované osoby zastřešující oblast mobilních telefonů. Jednalo se o 34 členů z řad výrobců mobilních zařízení, mobilních operátorů a developerů. Úkolem bylo prosadit operační systém Android jako jediný na trhu a zajistit si tak v určitém smyslu slova monopol, navzdory už rozdělenému trhu konkurenty. V současné době se sdružení skládá z 84 členů, kteří jsou zavázáni nepoužívat ve svých mobilních zařízeních jiný operační systém než je Android. [3]



Obrázek 6 - Někteří členové Open Headset Alliance  
Zdroj: [21]

V ten samý den, jakým bylo oznámení o vytvoření aliance, představil Google nově vzniklou platformu pro mobilní zařízení – Android. Vzhledem k faktu, že se jednalo o open source platformu, byly také uvolněny vývojové nástroje a to zcela zdarma i když mnozí systém přirovnávali k Windows Mobile, který se v té době od Androidu vzhledově příliš nelišil. Zákazníci si však museli ještě rok od oznámení počkat, než byl představen i první mobilní telefon vybavený tímto operačním systémem, HTC Dream s procesorem o taktu 528 Mhz, operační paměti o velikosti 192 MB a vnitřním úložištěm 256 MB. [3]



Obrázek 7 - HTC Dream ( T-mobile G1)  
Zdroj: [15]

### 3.2.2 Vývoj systému

Jádro systému Android je postaveno na Linuxové architektuře s úpravami, které umožňují využití na mobilních zařízeních. Některé části byly plně přizpůsobeny, jiné zase zcela zablokovány z důvodu funkční kompatibility. S tímto postupem nesouhlasí někteří zastánci čistě Linuxového přístupu, protože se v jádru věci nejedná o standardně používaný přístup a více prosazují spolupráci s nezávislými vývojáři. Společnost Google nepřišla jako první s myšlenkou využití právě Linuxového jádra, ale zásluhou jednoduchého uživatelského rozhraní a masivní podporou mnoha programů a her jako jediná uspěla a dokázala se postavit hlavním konkurentům, Windows Mobile a iOS. [1]

Většina členů aliance vychází ze základní verze systému, nicméně si pro jednotlivá zařízení systém upravují, příkladem může být HTC a úprava s názvem Sense. Najdou se ale pořád i nadšenci, kteří vytváří vlastní verze nezávisle na oficiálních updatech, nejznámější Cyanogen Mod. [2]

Vývojáři starající se o zdokonalování jednotlivě vydaných verzí, přizpůsobují systém k integrování nových funkcí a možnost využívání nejmodernějších technologií v nově vydaných mobilních zařízeních. V prvních verzích bylo velmi obtížné sestavit operační systém tak, aby po přidání nových funkcí nebylo nutné přepisovat celý kód systému. Co je dnes samozřejmostí, kdy prakticky každá aplikace může komunikovat s jakoukoliv částí v zařízení, dříve nebylo možné. Celý návrh nejvíce připomíná stavebnici, kdy přistavěním dalších dílů, není nutné bourat původní stavbu, dochází tak k čisté nástavbě v podobě novějších verzí. [1, 2]

Google si uvědomoval svoji pozici nového hráče na trhu a tak bylo vydání první verze Androidu pod silným tlakem. V listopadu 2007 byl uvolněn první verze balíček nástrojů pro vývojáře ve formě SDK, u února 2008 měli přední výrobci k dispozici náhledy s tímto operačním systémem a v září téhož roku došlo k vydání prvního telefonu s operačním systémem Android. [3]

### 3.2.2.1 Betaverze systému

Dříve než byly uvolněny plnohodnotné verze operačního systému Android, bylo vydáno hned několik beta verzí, stejně jako tomu je dnes u vývoje např. počítačových her. První beta verze byla testována pouze na emulátoru založeném na prototypu telefonu od společnosti HTC s názvem Sooner. Zařízení bylo často srovnáváno a také nazýváno pouhou kopií telefonu od BlackBerry primárně kvůli zvolenému designu, kdy telefonu dominovala qwerty klávesnice a nízký, za to široký displej. Samotný telefon se však nikdy nedostal do prodeje a vývojáři zůstali pouze u prototypu. [6]

Následující beta verze už opustila vzhled připomínající BlackBerry a samotný emulátor více připomínal zařízení, jakým směrem se ubírá Android i dnes.

Poslední beta už byla prakticky k nerozeznání od finální verze oficiálně vydané Android a označení Android 0.9 tomu také napovídalo. Umožňovala v první řadě instalaci více aplikací, kdy předešlá verze se zastavila na čísle 21 a také úpravy domácí obrazovky.

### 3.2.3 Vývojové verze

Vydání jednotlivých verzí systému vedle nových funkcí přinášelo i opravdu bezpečnostních závad na systému. Pokud se nachází v seznamu podporovaných zařízení přístroj uživatele, byl upozorněn s možností stažení aktualizace zdarma. Každý název nové verze je pojmenovaný podle anglického názvu některého ze sladkých zákusků a zároveň dodržuje posloupnost abecedy, logo je vytvářeno prolnutím hlavního loga Androidu, tedy robota do daného zákusku. [1, 2, 3, 4, 5]

Jednotlivé verze Androidu obsahují různé specifické nároky na hardware. Každá vydaná aplikace na Android obsahuje tzv. číslo API z důvodu rozeznání pro jakou verzi je aplikace vhodná a plně kompatibilní. Následující kapitola představuje hlavní změny, které jednotlivé verze přinesly.





**Obrázek 8 - Data vydání jednotlivých verzí a jejich loga**  
Zdroj: [22] (s úpravami autora práce)

### 3.2.3.1 Android 1.X

První oficiálně vydaná verze pro Android byla určena především pro dříve zmíněný HTC Dream. Obsahem byl procesor o taktu 528 MHz s jedním jádrem a 192 MB operační paměti RAM. Oproti svému největšímu konkurentovi iPhone, který se snažil prosadit strategii kázající co nejméně hardware tlačítek se lišil právě v počtu tlačítek, které tělo telefonu obsahovalo, nepočítajíc výsuvnou klávesnici qwerty,

Operační systém nabízel aplikace od společnosti Google včetně synchronizace pro kalendář nebo e-mail. Jedna z hlavních aplikací byly mapy, kdy bylo umožněno uživateli prohlížet místa zájmu pomocí panoramatických fotek, funkce byla již dříve známa jako Google Street View. Důležitou součástí byl i webový prohlížeč, nicméně většina navštívených stránek nebyla responzivní pro mobilní zařízení. [1, 2, 3, 4, 5]

S vydáním první verze si odbyl premiéru i Google Market, dnes Google Play, který sloužil jako centrální místo pro stažení dodatečných aplikací nebo aktualizaci těch stávajících. Celkově systém nevybočoval ze zaběhnutého standardu, který nastavili konkurenti. K dispozici byl také fotoaparát, WiFi, průzkumník souborů nebo Bluetooth.

Jednotlivé verze s číslem za 1, z většiny opravovaly chyby a aktualizovaly rozhraní API na novější verze. Větších změn se událo až po vydání aktualizace 1.5, kdy samotný uživatel poznal na první pohled rozdíl v podobě změny grafického rozhraní všech aplikací od Google. Bylo umožněno také nahrávat skrze aplikaci YouTube videa přímo ze zařízení.

Přidány byly alternativní softwarové klávesnice, které v pozdější době nabízely vybrat si výrobce chytrých telefonů ze dvou variant, hardwarovou klávesnici mohli tak úplně odstranit a přiblížit konstrukci telefonu více k high-tech vzhledu.

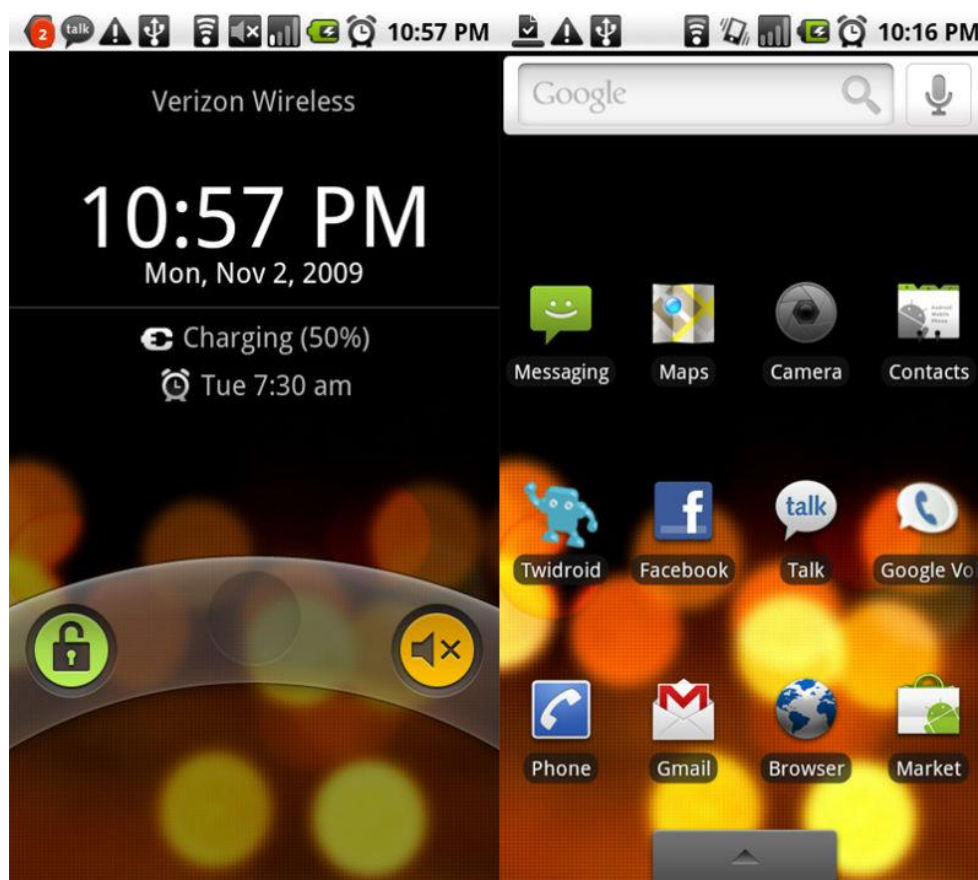
Šanci ihned využila společnost HTC a vydala tak po sobě již druhý telefon s čistě Android operačním systémem. Přidány byly různé animace nebo možnost větší variability v úpravě fotek pořízených fotoaparátem. Novinkou byla i možnost úpravy domácí obrazovky pomocí widgetů a znázornění stavu baterie, podle které mohl uživatel přispět k její výdrži, dále pak vylepšené vyhledávání hlasem. Nakonec dostalo změny i v základu systému s podporou CDMA sítí a rozšířilo tak své portfolio na všechny operátory.

### 3.2.3.2 Android 2.X

Měsíc a půl od vydání poslední aktualizace verze 1, vydal Google verzi s označením 2.0 a dal tak najevo, že se s ním musí počítat. Na poli hardwaru byla vydána novinka v podobě mobilního zařízení Nexus One, nicméně pokud uživatel neměl trvalé bydliště na území USA, byla možnost telefon získat velmi nízká. Zároveň přichází i Motorola Droid a společnost Motorola musela zaplatit licenční poplatek Lucasfilm.

Softwarově byla aktualizace velmi povedená, ačkoliv se jednalo o rozdíl měsíce a půl mezi přechodem verzí, přinesla aktualizace i přes všechny pochybnosti důležité změny. Zdokonalena byla synchronizace, do které bylo možné připojit více účtů, podpora Bluetooth 2.1. a integrace nových funkcí pro fotoaparát včetně digitálního zoomu. Zajímavou novinkou byl webový prohlížeč s podporou jazyka HTML5 a geolokace.

První vylepšení verze 2.0 přinesla dlouho očekávanou změnu v práci s úložištěm a jednotlivými aplikacemi. Byla zpřístupněna možnost instalovat aplikace na externí paměťové úložiště nebo přesouvat aplikace mezi interní paměť a paměťovou kartu. Další novinkou byla možnost vytvoření tzv. Wifi hotspotu, kdy jeden uživatel s přístupem na internetovou síť pomocí Wifi signálu sdílel tento přístup veřejně, k tomu se ještě přidalo sdílení internetu pomocí USB kabelu, který byl připojen k počítači. [1, 2, 3, 4, 5]



Obrázek 9 - Vzhled zamknuté a domácí obrazovky verze 2.0

Zdroj: [23]

Android také s aktualizací přinesl integrovanou GPS navigaci, hodnoty akcií samostatných výrobců zabývající se tímto odvětvím se okamžitě propadly o několik desítek procent, lidé si uvědomovali, že najednou mohou vlastnit několik zařízení v jednom a nemusí investovat do ne vždy levných GPS navigací. Mezi další viditelné změny se přidaly pohybuující se tapety, jejich velká nevýhoda spočívala v náročnosti jak na hardware, tak i na kapacitu baterie. [1, 2, 3, 4, 5]

Pro „meziverzi“ operačního systému s číslem 2.2 byla novinkou aplikace Flash Player od společnosti Adobe. Opět ale byla nevýhodou jeho velká náročnost a to nejenom na mobilních zařízeních, ale problém s ním měly i stolní počítače.

V prosinci 2010 byla představena další aktualizace. Změnu zaznamenali uživatelé po pouhém spuštění, grafická podoba uživatelského rozhraní prošla změnou za účelem větší rychlosti systému.

Aktualizace přinesla změnu i pro vývojářskou část populace v podobě JIT, jednalo se o vylepšení kompilace pro Dalvik Virtual Machine, který obstarává běh jednotlivých aplikací v Androidu, vyřízení operací běžících na pozadí bylo až 5x rychlejší. Přibyla také nová funkce NFC, která rozšiřuje možnosti využití zařízení jako platební karty či jízdenky do metra pouhým přiložením mobilu ke čtečce. [1, 2, 3, 4, 5]

### 3.2.3.3 Android 3.X

Verze vydaná v prvním čtvrtletí 2010 byla velice specifická, protože se nejednalo o aktualizaci pro mobilní zařízení, ale cílené zaměření především na tablety z důvodu uvedení konkurence Applu a jeho iPadu.

Inovacemi prošlo hlavně rozlišení obrazovky i jednotlivých aplikací pro plnou kompatibilitu na velké displeje. S tím přišla i nutná změna softwarové klávesnice, z důvodu absence jakýchkoliv hardware tlačítek na zařízení a úvodní obrazovky, kde byly přidány nové widgety a změněn způsob jejich úprav. [1, 2, 3, 4, 5]

Verze s sebou přinesla i spousty nedostatků a nedodělků. Především mizerná stabilita, chybějící podpora Adobe Flash player a nepřístupnost ve formě open-source zavinily nízkou popularitu a rychlý konec podpory.

### 3.2.3.4 Android 4.X

Na řadu přišla nová verze aktualizace a snaha Androidu napravit reputaci po nepřilíh populární verzi 3.0. Verze nebyla rozdělena na část určenou pro tablety a část pro mobilní telefony a vrací se zpět k základu open-source. Jako první ho využila společnost Samsung v modelu Galaxy Nexus. Z předešlé verze bylo přeneseno pouze nastavení, které počítalo pouze se softwarovými tlačítky. Hlavní změny se i v této aktualizaci týkali grafického rozhraní. Nejednalo se pouze o menší úpravy a vylepšení, ale o celkové přepracování, které bylo naposled viděno u verze 2.0.

Kromě grafiky se vylepšení dostalo i technologii Android Beam, značící přenos dat na krátkou vzdálenost. Zdokonalení se týkalo i ovládání hlasem resp. jeho rozpoznávání, stejně tak vylepšení fotoaparátu pro využití zabezpečení, pomocí kterého uživatel mohl využít vlastní podobiznu, obličej, k odemčení telefonu.

Mezi jednotlivými aplikacemi, které byly spuštěny bylo možné snadné přepínání. I přes přítomnost, přímo se orientujících aplikací na Google obchodě pro úpravy fotek, byly přidány další možnosti jejich editace přímo v telefonu prostřednictvím základní aplikace fotoaparátu. [1, 2, 3, 4, 5]

První průběžná aktualizace stavající verze nepřinesla příliš změn z důvodu, že jednotlivé aplikace od společnosti Google byly průběžně samostatně aktualizovány bez ohledu na oficiální termín vydání celkové aktualizace systému. Vylepšení se dostalo především samotnému jádru systému pomocí Project Butter, který obstarával rychlejší a plynulejší vykreslování obrazu a byla tak zmenšená náročnost na využití procesoru, celá tato operace pak měla i pozitivní dopad na využití baterie v zařízení.

Součástí nové verze byla aplikace, která se zde poprvé objevila jako nový přírůstek do integrovaných softwarových součástí s pojmenováním Google Now. Kdy pomocí zjištění aktuální polohy vyhledal mobilní telefon nejbližší bary a restaurace, aplikace nesloužila pouze pro zábavu, ale mohla uživateli přinést i jinou užitou hodnotu v podobě aktuálního počasí nebo inteligentní navigace, která se řídila dopravním zpravodajstvím a dopravila vozidlo i s cestujícími do cíle bez zbytečných dopravních komplikací. Změnami prošla i notifikační lišta umožňující přístup do nejvíce používaných složek nastavení. S uvedením chytrých hodinek na trh byla přidána do systému jejich podpora a funkce pro zajištění přenosu dat i při malém množství energie. [1, 2, 3, 4, 5]

Aktualizace 4.4 byla původně označená jako Key Lime Pie. Ale sladkost, která byla velice oblíbená mezi vývojáři společnosti Google zajistila přejmenování na KitKat. Příjemnou změnou byla náročnost celého systému směrem dolů. Mobilní zařízení musely splňovat minimální požadavky paměti RAM pouze 512 MB, dříve 1024 MB.

Stejně tak jako tomu bylo o předešlých verzích i tato sebou přinesla grafické změny, jednalo se o drobnosti v podobě notifikační lišty nebo průhlednosti ikon aplikací. Byla přidána podpora pro nové využití senzorů pomocí aplikací např. různé měřicí nástroje nebo krokoměr a s trochou překvapení i podpora pro infračervený senzor, který byl na ústupu po zavedení a následném vylepšení Bluetooth technologie.

### 3.2.3.5 Android 5.X

Aktualizace vydaná roku 2014 pod anglickým názvem Lollipop a českým Lízátko, sjednocovala grafické rozhraní na všech zařízeních. Společnost chtěla dosáhnout toho, že grafický vzhled bude na všech zařízeních stejný neohledě na rozlišení daného displeje nebo zda se jedná o mobilní telefon či tablet. Rozšířili se také možnosti využití u NFC technologie v podobě obnovy dat. Pokud nový i starý telefon uživatele obsahuje tuto technologii, pak stačí telefony k sobě přiložit. V novém telefonu bude tak kompletní nastavení uživatelského rozhraní, včetně aplikací, které byly uloženy ve starém zařízení.

Zajímavostí bylo přidání úsporného režimu ke zvýšení výdrži baterie a v neposlední řadě také profil host, pro nastavení sdílení obsahu telefonu při využití daného zařízení více osobami. Dále byla vylepšena podpora dvou SIM karet, v případě aplikace Číselník se jednalo o spuštění dvou grafických rozhraní, každé mělo jinou barvu a uživatel tak bezpečně poznal, jako SIM kartu využívá.

Notifikační lišty zaznamenaly dalších změn a nabízejí více možností, jak s nimi pracovat, dotykem je možné otevřít příslušnou aplikaci, tahem do strany zase upozornění zase vymazat nebo celou lištu posunout směrem nahoru a odmítnout tak všechny došlé notifikace. [1, 2, 3, 4, 5]

Přizpůsobení a samotný přístup do rychlého nastavení zaznamenal také změny. Skrze zabezpečený telefon lze ovládat některé funkce, jakými jsou např. Wifi nebo Bluetooth, dlouhým přidržením je možné ikony v rychlém nastavení odstranit.

### 3.2.3.6 Android 6.X

Aktualizace se dočkala mnoha klíčových vylepšení, které z části reagovaly na připomínky uživatelů zařízení. RAM manager je přesnější a nabízí širší úhel pohledu na využívání paměti a baterie jednotlivými aplikacemi a uživatel má tak možnost se sám rozhodnout, kterou aplikaci běžící na pozadí ukončí a přispěje tak k většímu výkonu svého zařízení. [1, 2, 3, 4, 5]

Zabezpečení udělalo krok vpřed a změnil se způsob řešení udělování oprávnění aplikacím. Dosud žádaly aplikace během instalace povolení k částem telefonu, které ze své podstaty nebylo nutné k plné funkčnosti využívat např. aplikace na vyhledávání spojů, žádala o povolení přístupu ke galerii.

Aplikace bude žádat o zpřístupnění jednotlivých komponent v případě, že je opravdu bude pro své operace potřebovat a systém zobrazí varovné upozornění o souhlasu.

Novinky zaznamenaly i výrobci chytrých telefonů s podporou čtení otisků prstů a tak nemusí tuto funkci programovat zvlášť. Samotné připojení USB kabelu přišlo se změnou konektivity, resp. novým typem kabelu, kdy není podstatné, jakou stranou je kabel k telefonu připojený, protože je reverzibilní. [1, 2, 3, 4, 5]

### 3.2.3.7 Android 7.X

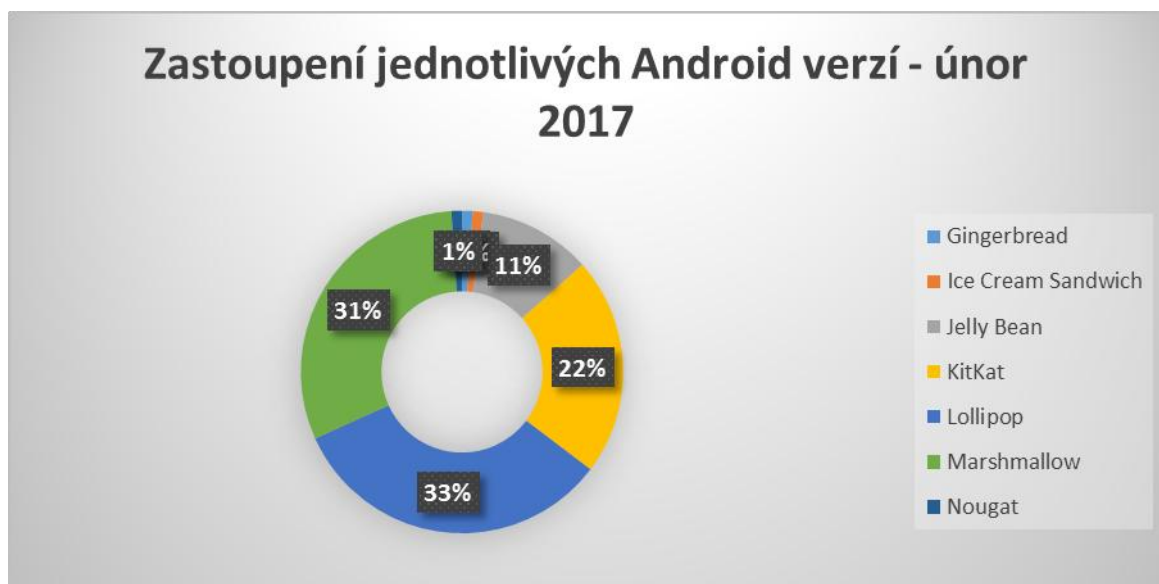
Poslední velká vydaná aktualizace přichází s integrací API Vulkan, který zajistí vyšší grafický výkon pomocí přístupu systému ke grafickému procesoru. Telefony Samsung S7 toto vylepšení již obsahovaly, ale Google integruje tuto funkcionalitu přímo do systému, takže je dostupná pro všechny telefony s danou verzí operačního systému. S rychlostí systému pomůže také nově vylepšený JIT kompilér, podporující rychlejší instalaci souboru z důvodu zmenšení jejich datových velikostí.

Technologicky se dostává do popředí i virtuální realita. Telefony obsahující tento systém budou nativně podporovat tuto technologii.

Praktické vylepšení obsahuje noční režim s redukcí modrého světla, šetří tak lidský zrak během noci. Za zmínku také stojí vyřešení systémových aktualizací, kdy si telefon sám stáhne a nainstaluje danou aktualizaci a uživatelé poté pouze upozorní na nutnost restartování telefonu. [1, 2, 3, 4, 5]

### 3.2.3.8 Zastoupení jednotlivých verzí na trhu

V předešlé kapitole byly nastíněny všechny důležité změny, které jednotlivé aktualizace systému Android přinesly. S tím jak se mění výdrž mobilních telefonů po technické stránce u uživatelů se také mění zastoupení jednotlivých verzí na trhu. Proto se některé starší verze dnes již prakticky nepoužívají, k tomuto faktu přispěl i důvod, že byl odepřen starším verzím než je 2.2 přístup na obchod Google Play, výjimku tvoří ještě aktualizace 3.0, která nebyla úspěšná ani na začátku svého uvedení na trh. [9]



**Obrázek 10 - Zastoupení verzí systému mezi uživateli**  
Zdroj: vlastní zpracování

Data z grafu vycházejí dle počtu přístupů mobilních zařízení s daným operačním systémem na Google Play. Výsledkem je zjištění, že nejvíce využívaná verze systému s v mobilních zařízeních s Android OS stala aktualizace s názvem Lollipop s datem vydání listopad 2014 a pouze 2% náskokem před verzí Marshmallow. [9]

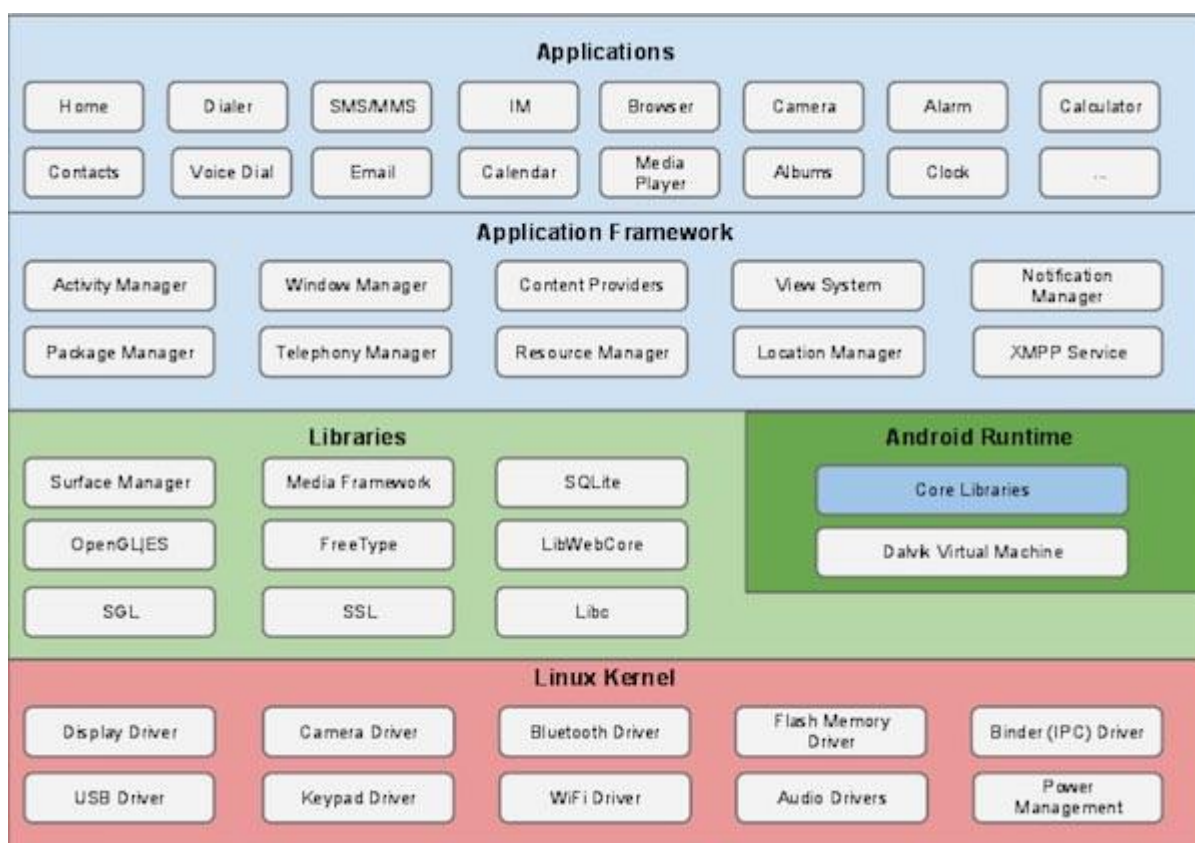
### 3.2.4 Architektura

Jak bylo dříve zmíněno, základy celého systému Android postavené na Linuxovém jádře upraveném dle specifikací mobilních zařízení. [5]

Celý operační systém Android se skládá z pěti hlavních vrstev. Každá vrstva obsahuje několik komponent, pracuje jako samostatný celek a spravuje oblast, která jí byla přidělena, vrstvy spolu však navzájem komunikují. Kompletní operační systém je uložen ve vnitřní operační paměti ROM. Přístup do této části paměti obstarává Boodloader, který také zastává pozici BIOSu u standartních počítačů.

Programy jsou psány v jazyce Java a poté komprimovány speciálním virtuálním strojem Dalvik., společnost Google nemusí tak platit licenční poplatky za využívání programovacího jazyku Java.Open-source zajišťuje výrobcům možné úpravy nejenom jádra, ale i zbylých částí operačního systému a dostatečně se tak odlišit od konkurenčních značek i když sdílí stejný operační systém..[5]





Obrázek 11 - Znárodnění architektury OS Android

Zdroj: [24]

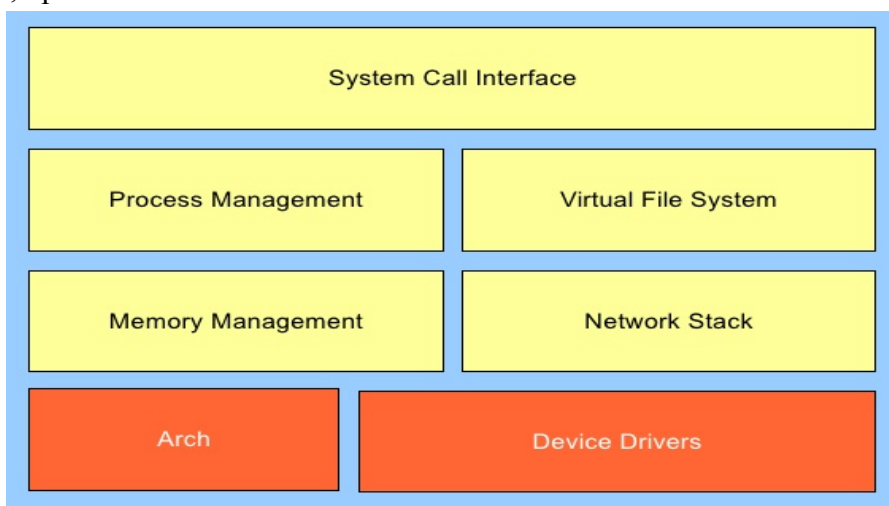
Na obrázku číslo 11. je graficky znárodnění rozdění vrstev a jejich součásti, které spravují. Nejnížší vrstva obsahuje jádro, kam běžný uživatel nemá práva přístupu. Nejvyšší vrstva oproti tomu je vrstva aplikací, které uživatel osobně nainstaloval nebo se jedná o interní aplikace dodávané Googlem. [6]

### 3.2.4.1 Zavadeč

Jak už bylo řečeno, jedná se o samostatně oddělený program od operačního systému sloužící k jeho samotnému spuštění, zavedení. Je tedy možné pomocí zavadečů k tomu určených, obměnit verzi operačního systému i přes absenci v podporovaných zařízeních nebo změnit operační systém úplně. Dalšími jeho úlohami může být v případě aktualizace, asistence při reinstalaci nebo kontrolování funkčnosti hardware komponent. Zpřístupňuje speciální režim pro zapisování do paměti ROM, zajímavostí je, že přímo přes tuto část se provádí blokáce užití, pokud by chtěl uživatel využít služeb jiného operátora, než u kterého zařízení zakoupil. Zpřístupnit ho lze z většiny pomocí kombinace kláves, která je rozdílná, dle toho, o jaké zařízení se jedná. [7]

### 3.2.4.2 Linux Kernel

Nejnižší a zároveň nejdůležitější vrstvou je Linux Kernel, jádro verze 2.6, po vydání Androidu 4.0 běží systém na verzi jádra 3.0. Příčinou využití jádra od Linuxu byla jeho snadná přenositelnost mezi různými druhy zařízení a společná vlastnost pro obě společnosti, open source.



**Obrázek 12 - Vnitřní architektura Kernel**

Zdroj: [25]

Samozřejmě prošlo jádro úpravami od společnosti Google, nahrazena např. byla knihovna GNU, knihovnou, kterou vytvořil Google přímo pro Android. Výhodou je její velikost, protože je přizpůsobena pro procesory s malou frekvencí. Má na starost správu procesů, které jsou spuštěny jednotlivě, odděleně od sebe a je tím tak zajištěna stabilita systému a základní bezpečnost, neboť uděluje oprávnění jednotlivým aplikacím, dále pak správa paměti a další nezbytné základní funkce, jedna z hlavních funkcí je také zajištění komunikace mezi hardwarem a softwarem, znázorňující i obrázek architektury. [7]

### 3.2.4.3 Knihovny

Další vrstvou v architektuře OS Android jsou knihovny, napsány programovacím jazykem C/C++ a umožňují pracovat mobilním zařízením s různými typy dat a jsou základní stavební kámen pro aplikační rámeček. Knihovny jsou zodpovědné za výpočty složitějších operací, jako je vykreslování grafiky nebo práce s mediálními soubory. [6]

Příklady některých knihoven s jejich specifikacemi jsou uvedeny níže:

- Surface manager – vrstvení jednotlivých oken grafiky, k výslednému obrazu, které je uživatel schopen vidět
- OpenGL/SGL – Open GL zajišťuje vykreslení 3D grafiky a SGL vykreslení 2D grafiky v podobě textu nebo základních geometrických tvarů
- Media Framework – knihovna mediálních kodeků sloužící k přehrávání různých videí a zobrazování obrázků v mobilním zařízení
- FreeType – vykreslování vektorového a bitmapového stylu písma
- SSL – ochrana pro komunikaci v síti, šifrování přenosu dat
- SQLite – základní databáze k ukládání dat s přístupem všech aplikací
- Webkit – jádro webového prohlížeče pro HTML a JavaScript
- Libc – knihovna se základem v Unixu se změnami potřebných pro chod v Linuxu poskytující engine webovým prohlížečům

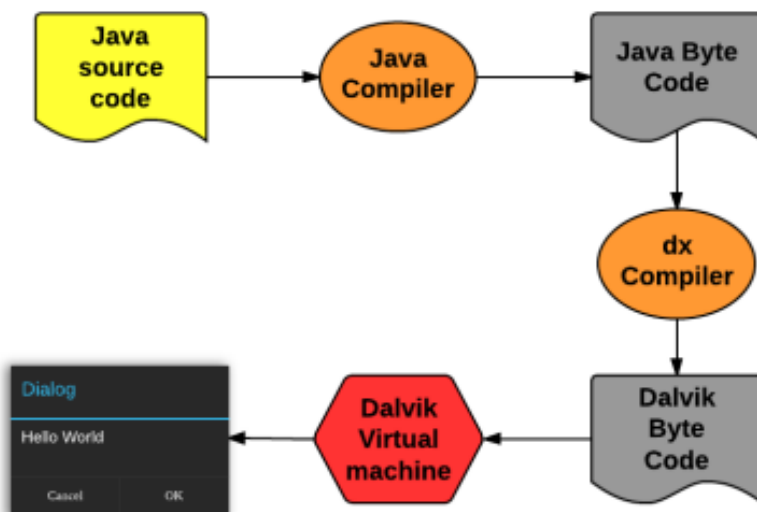
Obsahem jsou také některé API knihovny používané při vývoji aplikací:

- android.os – knihovny pro přístup k základním funkcím systému
- android.webkit – knihovna, obsahující nástroje pro práci ve webovém prostředí
- android.telephony – knihovny pro komunikaci
- android.widget – knihovny pro ovládání tlačítek

#### 3.2.4.4 Android Runtime

Vrstva, na které běží veškeré aplikace, obsahující Dalvik Virtual Machine a základní knihovny Java. Většina aplikací je napsána v Java kódu a vzhledem k placení licence při využívání Java Virtual Machine (neplatí pro Java knihovny, ty jsou volně šiřitelné), byl naprogramován DVM, který je se svými nízkými nároky na procesor a baterii lépe optimalizovaný na mobilní zařízení a také Google není nucen platit onu licenci firmě Oracle. [5, 6, 7]

Programy jsou napsány v Java kód, převedeny na Java byte kód, kompilovány Dalvikem do DEX souborů a odeslány na DVM, který převede program do přímo spustitelné formy na mobilním zařízením.



Obrázek 13 - Průběh kompilace Java kódu  
Zdroj: [26]

#### 3.2.4.5 Aplikační rámec

Vrstva, se kterou vývojáři aplikací pro operační systém Android nejvíce pracují. Knihovny jsou napsány v Java jazyce a jedná se o základní kameny každé aplikace a rozšiřují možnosti jejího využití. Níže jsou uvedeny nejdůležitější bloky, se kterými může vývojář pracovat. [5, 6, 7]

- Activity Manager – nejdůležitější, obstarává životní cyklus aplikace
- Window Manager – řídí výstupy obrazových oken resp. jejich vykreslování
- Content Providers – sdílení obsahu dat určité aplikace pro jinou aplikaci
- View Sytem – obsahuje prvky uživatelského rozhraní např. tlačítka
- Package Manager – možnost zjištění aplikačních balíčků nacházející se v zařízení a sdělení informace na vyžádání aplikaci
- Telephony Manager – spravuje telefonní služby

- Resource Manager – správa zdrojů různých formátů využívaných v aplikaci
- Location Manager – zjišťování polohy dle GPS nebo signálu mobilní sítě
- Notification Manager – správa notifikační lišty

#### 3.2.4.6 Aplikace

Nejvyšší a také poslední vrstvu, kterou jako jedinou běžný uživatel může vidět, představují aplikace. Jsou zde uloženy aplikace integrované společností Google a jejími vývojáři, které nejdou běžným způsobem odinstalovat (kalendář, kontakty, webový prohlížeč) a také aplikace nainstalované samotným uživatelem pro vlastní přizpůsobení telefonu. [5, 6, 7]

#### 3.2.5 Distribuce

Vývojář má dvě možnosti, jak nově vytvořenou aplikaci publikovat veřejnosti. Zveřejnit aplikaci na neoficiálních zdrojích, které k tomu nejsou přímo určené nebo využít služeb obchodu Google Play. Ještě před samotným vznikem obchodu se jednalo o standardní postup, ale v dnešní době zveřejnění aplikace na jiném místě než je Google Play se nepraktikuje. Existují stále neoficiální zdroje v podobě např. Black Market, stažením aplikace prostřednictvím tohoto zprostředkovatele, uživatel riskuje naložením bezpečnosti svého zařízení. Vytvořená aplikace obsahuje koncovku apk, samotný zdroj aplikace obsahuje mimo jiné také informace o kompatibilitě jednotlivých zařízení dle specifikací výkonu a požadovanou verzi API. [9]

### 3.2.5.1 Google Play

Důvodem způsobující popularitu chytrých telefonů, jsou aplikace. Umožňují široké rozšíření základních funkcí a mohou zastupovat svým potenciálním využitím více zařízení než je obyčejný telefon. Google toto zprostředkovává prostřednictvím obchodu Google Play. Jednalo se o reakci na publikování obchodu s aplikacemi u konkurence ve formě Apple Store. [9]

Dnes již není uživatel, vlastnící chytrý telefon, který by obsahoval pouze přednainstalované aplikace, mezi kterými je i obchod. Služba Google Play reprezentuje snadný přístup k aplikacím všeho druhu. Na začátku po vytváření nabízel pouze samotné aplikace, uplynutím doby byla nabídka rozšířena o hudbu, videa a filmy resp. nabídkou o jejich vypůjčení.

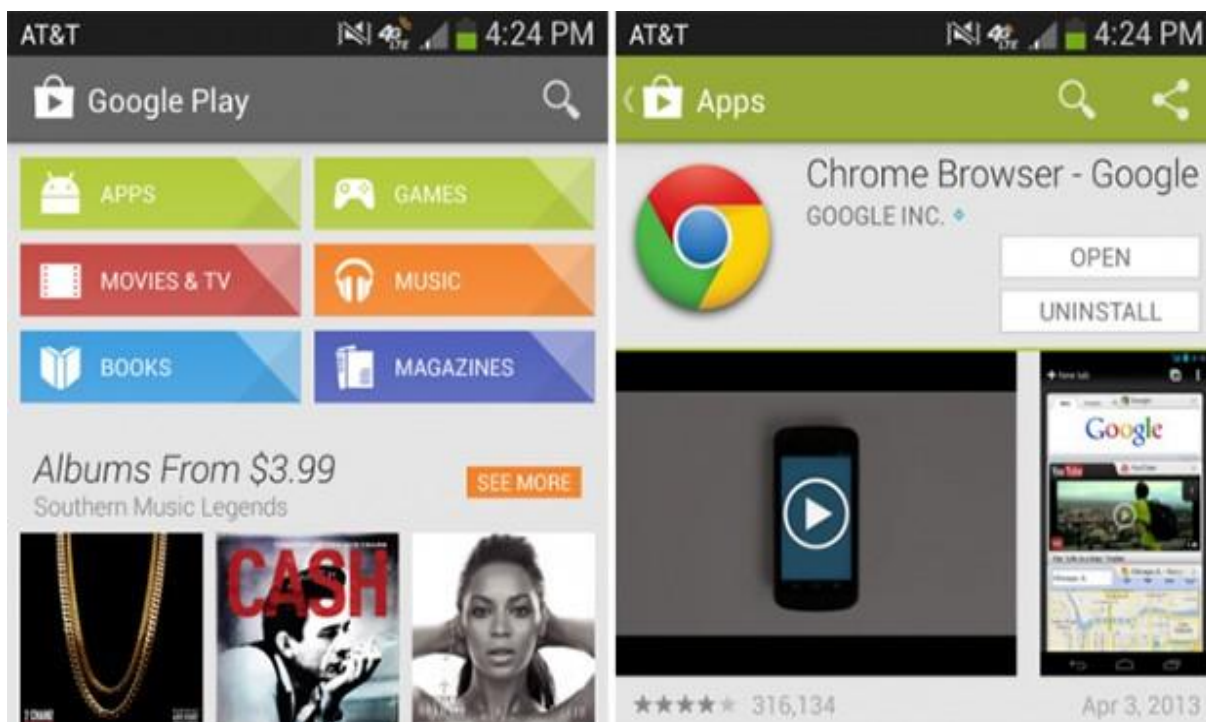
Základní kategorie jsou rozděleny na placené a neplacené aplikace, jednou z podmínek využívání služby je založení účtu, které je možné propojit s bankovním účtem a v případě placené aplikace je tak stažení aplikace stejného komfortu jako neplacené.

Výhodami jsou celková přehlednost obchodu a snadného vyhledávání. Jak již bylo řečeno, aktualizace se provádí automaticky, bez vyzvání uživatele k jejímu schválení nebo s připomínkou zveřejnění nové verze, odvíjí se od nastavení telefonu a upozornění. [9]

Nevýhodou je zde nebezpečí nainstalování škodlivého softwaru, na rozdíl od Applu a Microsoftu neprocházejí aplikace složitým okruhem schvalování a je možné je rovnou nahrát do obchodu. Částečné ošetření bezpečnostní situace řeší jednoduchý program od Google na kontrolu aplikací, stále je ale nutné být ostražitý při instalaci neznámých aplikací.

Pokud uživatel přistupuje k obchodu, jako vývojář má k dispozici speciální účet po zaplacení vstupního poplatku v přepočtu 500 Kč. Na základě žádosti byla Česká republika v roce 2012 zařazena mezi země, které si mohou otevřít vývojářské účty. Důvod byl především v dobré pověsti vývojářů v naší zemi. Po registraci speciálního účtu je zpřístupněna možnost publikování vlastní vytvořené aplikace, pokud se jedná o placenou aplikaci je vývojář vyzván k dodatečné registraci do Google Checkout služby související s převodem peněz. [9]

V případě přihlášení pod standardním uživatelským účtem se zobrazené aplikace mohou měnit v závislosti na zemi, ze které uživatel pochází, dle právních norem schválených v daném státě. Obsah obchodu zobrazovaný návštěvníkovi může souviset i s mobilním zařízením, přes které do obchodu přistupuje. V případě zjištěné nekompatibility s přístrojem uživateli tato aplikace není ani zobrazena v nabídce.



Obrázek 14 - Vzhled Google Play

Zdroj: [27]

Po zvolení určité aplikace je uživateli zobrazen její profil, jehož obsah spravuje sám vývojář. Jedná se o logo, stručný popis charakterizující účel a způsob využití aplikace a dále fotky, v některých případech i videa, zobrazující vzhled aplikace. V dolní části je také uvedena statistika počtu stažení a následně hodnocení jednotlivých uživatelů v podobě přidělení hvězdiček a krátkého slovního odůvodnění.

Společnost Google přidala dvě nové služby související s obchodem. První služba z roku 2012 nabízí přístup k distribuci soukromých aplikací v rámci společnosti. Zaměstnanec se přihlásí do obchodu pod firemním účtem a ihned se zobrazí nabídka aplikací registrována pod danou doménou. Jedná se o způsob šíření aplikací pro interní využití, jednotlivé aplikace nemohou být zveřejněny na běžném obchodě a soukromém kanále zároveň. [9]

Druhá novinka umožňuje vyzkoušení placené aplikace. Uživatel aplikaci stáhne, vyzkouší, a pokud nenaplní očekávání nebo nesplňuje cíle, které jsou popsány na profilu aplikace, může aplikaci do 15 minut vrátit. I tato možnost bývá zneužívána hackery, kteří aplikaci stáhnou z obchodu, prostřednictvím speciálního programu aplikaci zkopírují, následně vrátí a po odstranění zabezpečení volně distribuují. Statistiky u některých aplikací poté zobrazují registrování licence např. ve sto případech, i když k samotnému zakoupení aplikace došlo pouze v 10. [9]

### **3.2.6 Bezpečnost**

Dostatečné zabezpečení všech operačních systémů na mobilních telefonech je přední prioritou všech výrobců těchto zařízení. Vysoká popularita Androidu zajistila i nelichotivé prvenství v počtu hackerských útoků na mobilní telefony vybavené jejich operačním systémem, situaci nezlehčuje i důvod rozšířenosti Androidu a to platforma open-source. Statistiky z roku 2010 uvádějí počty hackerských útoků na systém Android ve 24 % případů, nyní se čísla odhadují okolo 97% ze všech malware útoků mířících na mobilní zařízení. [8]

Útoky se mohou rozdělit na dvě skupiny, první skupinou je hackerský útok za účelem peněžité odměny, v tomto případě se většinou jedná o prolomení ochrany telefonu a odeslání prémiové SMS, druhá skupina cílí na získání citlivých informací o uživateli, hackeři přitom mohou využít nejenom data uložená v mobilu, ale i interní aplikace, pomocí kterých, odposlouchávají uživatele nebo pořídí fotografie bez jeho souhlasu.

Prvotní stav každé aplikace je, bez jakýchkoliv možností ovlivnit činnost systému. Každá operace prováděna aplikací v systému, musí být přesně definována a k provedení takové operace následně žádá aplikace systém při instalaci o celý seznam různých oprávnění. Každá aplikace musí být také podepsána bezpečnostním certifikátem ověřující totožnost autora. [8]

S každou aktualizací přichází i seznam oprav bezpečnostních trhlin v operačním systému. Ale jsou to právě uživatelé, kteří instalují aplikace ze zdrojů, které nejsou oficiálně doporučené společností Android, jediné takové bezpečné místo je Google Play.



Uživateli je k dispozici i tzv. aplikace ověření fungující na principu odeslání dotazu na danou aplikaci do společnosti Google, která se postará o identifikaci a vyhodnocení, zda jde o bezpečnou aplikaci. K zabezpečení slouží stejně tak jako na osobních počítačích, antivirové systémy. Obsahují část totožnou s funkcí aplikace ověření, ale také kontrolují požadavky oprávnění na operace jednotlivých aplikací, pokud např. aplikace na počasí požádá o oprávnění k přístupu do telefonního seznamu, vyhodnotí antivir aplikaci jako rizikovou a upozorní na to také uživatele s doporučením, jak se v takovém případě zachovat. [8]

Zde jsou doporučená pravidla, která by měl každý uživatel dodržovat a přispět tak k větší ochraně svého zařízení:

- Zabezpečit telefon heslem, gestem nebo otiskem prstu
- Aktivovat službu automatického zamykání
- Instalace některé z bezpečnostních aplikací
- Kontrolovat práva požadovaná aplikacemi
- Využívat bezpečných zdrojů pro stahování aplikací
- Instalovat vydané aktualizace
- Deaktivovat automatické připojení na Wifi síť
- Vypínat funkce Bluetooth a NFC ihned po skončení práce s nimi
- Neotvírat přílohy nevyžádaných e-mailů

### **3.3 Aplikace**

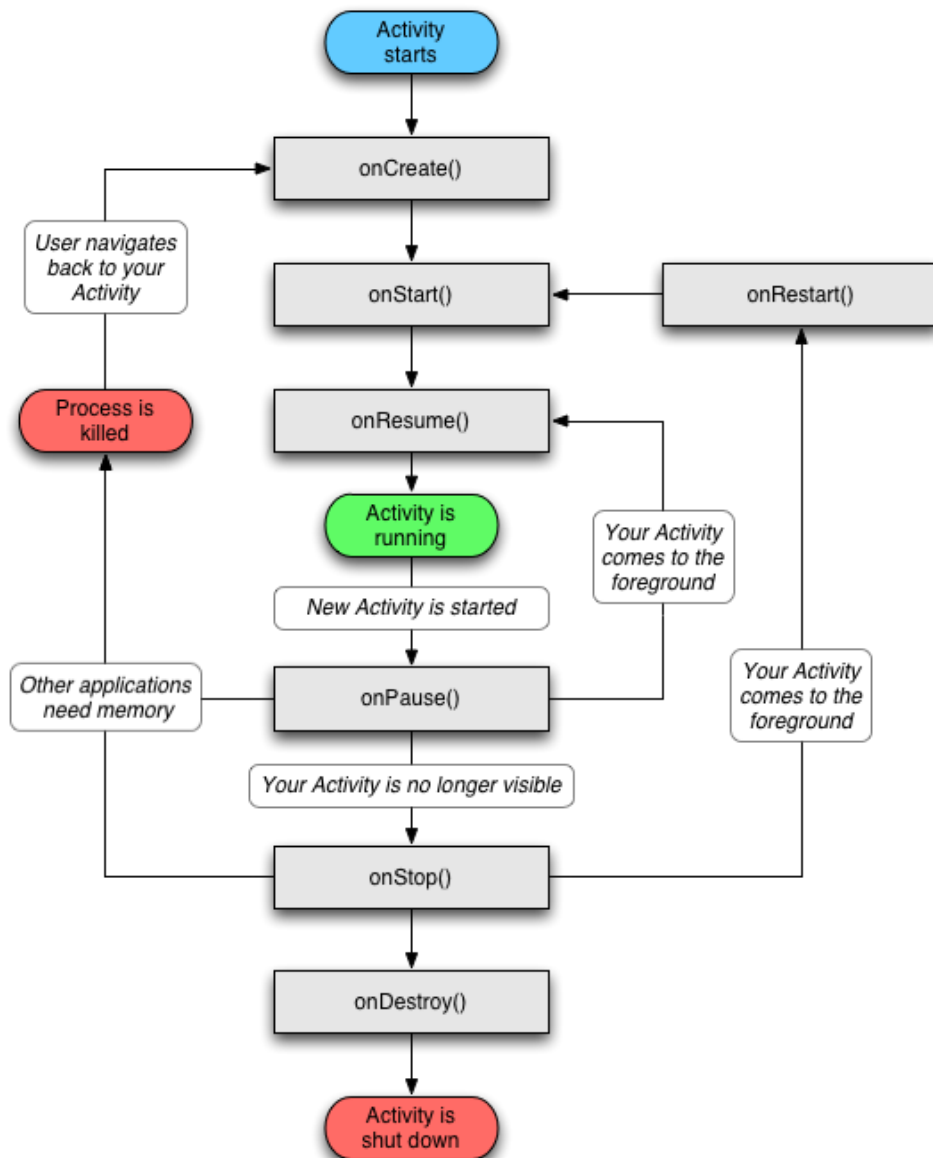
Stejně jako operační systém Android, tak i každá aplikace určená pro tento systém má určitou strukturu skládající se z různých komponent. Pět bloků tvoří základ a jsou blíže popsány v XML souboru Android Manifest, nemusí být všechny aplikací využívány. Architektura aplikací zpřístupňuje využití komponent jiných aplikací za předpokladu, že to sami umožňují. [4, 7, 9]

### **3.3.1 Android Manifest**

Obsahuje informace, jaká oprávnění aplikace potřebuje pro své fungování a definuje tak všechny přístupy k informacím, které bude využívat a také, jaká minimální verze systému musí být nainstalována na mobilním zařízení pro spuštění aplikace.

### **3.3.2 Activity**

Základ pro každou aplikaci. Zároveň je základem uživatelského rozhraní. Prakticky si lze aktivity představit jako analogové okno sloužící k interakci s uživatelem, kromě grafiky obsahuje i řídicí logiku. Aplikaci vždy tvoří více aktivit, ale jenom jedna je označena za hlavní. Vždy je aktivní pouze jedna a mohou se navzájem volat. Příkladem může být webová stránka, jedna aktivita bude zobrazovat úvodní stránku, v případě rozkliknutí záložky je uživatel odkázán na novou stránku a je vyvolána další aktivita. Aktivity tak spolupracují, ale jsou na sobě nezávislé. Dohledem nad řízením aktivit je Activity Manager, spravující celý životní cyklus za účelem neplýtvání výkonu. [4, 7, 9]



Obrázek 15 - Životní cyklus Activity  
Zdroj: [28]

Activity manager řídící životní cyklus a využívá následujících stavů:

- Activity starts – spuštění aktivity
- Activity is running – aktivity již zobrazená uživateli na telefonu
- Process is killed – nestandardně ukončená aktivity (např. nedostatek paměti)
- Activity is shut down – standardně ukončená aktivity

Mezi jednotlivými stavy dochází k vyvolání různých metod:

**onCreate()** – metoda volána při pouhém vytvoření Activity, tvoří uživatelské rozhraní a nachází se zde kód, který se má zpracovat po jejím spuštění

**onStart()** – je metoda volána ještě než se samotná aktivita zobrazí uživateli, pokud je aktivita na popředí volá se metoda **onResume()**, pokud jde o aktivitu na pozadí, volá se metoda **onStop()**

**onRestart()** – metoda volána v případě zastavení aktivity a jejím následném spuštění

**onResume()** – volána před zahájením interakce s uživatelem po metodě **onPause()**,

**onPause()** – metoda volána při změně aktivity, nemusí nutně znamenat i její ukončení

**onStop()** – volána v případě aktivity běžící na pozadí, metoda může být zavolána z důvodu ukončení aktivity nebo pokračování jiné, když aktivita není viditelná pro uživatele, po zavolání této metody je volána metoda **onRestart()** v případě obnovy aktivity nebo metoda **onDestroy()** při plném ukončení

**onDestroy()** – poslední metoda volána před ukončením životního cyklu aktivity

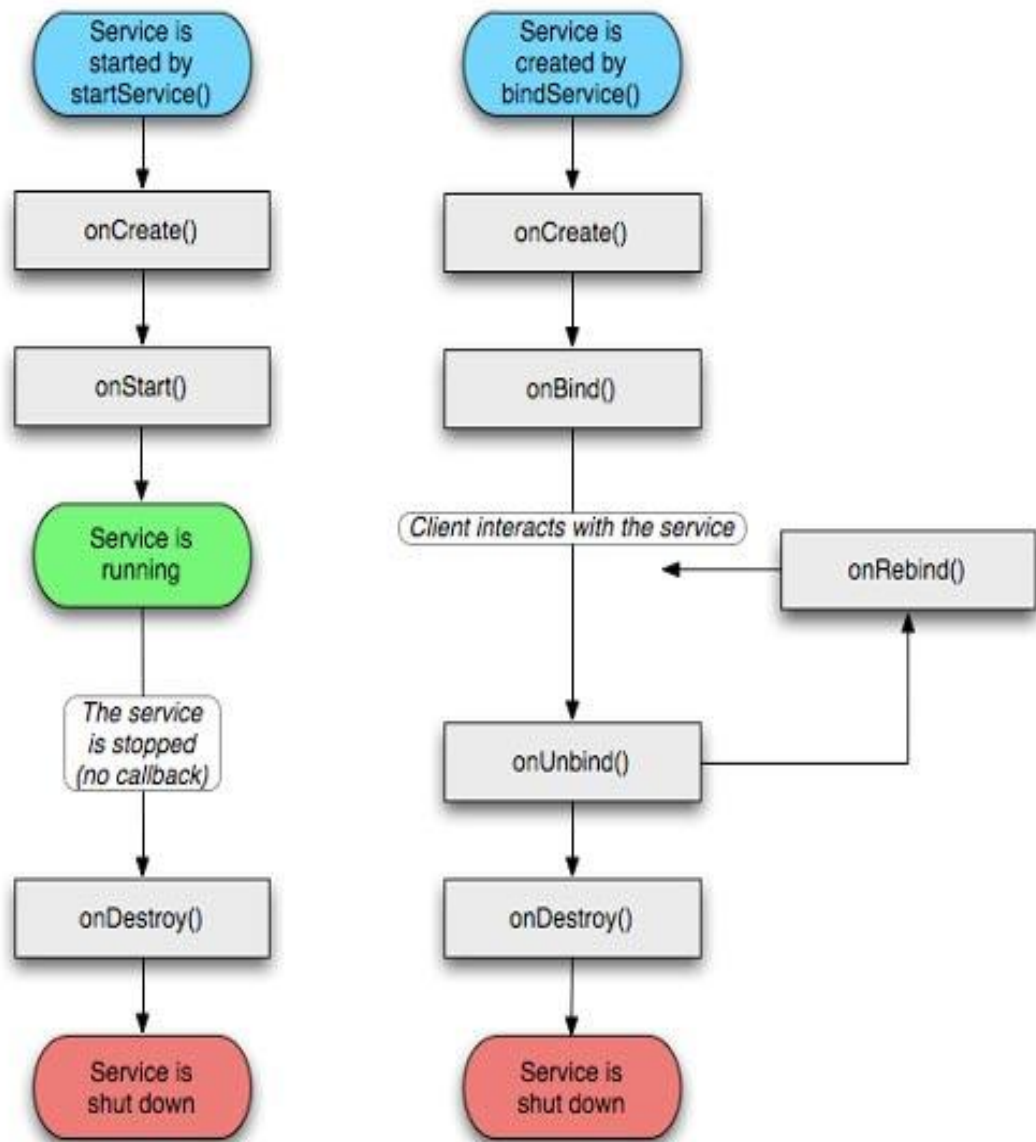
### 3.3.3 Services

Procesy běžící na pozadí bez uživatelského rozhraní s větší prioritou, než aktivity. I když mohou vykonávat stejnou funkci jako aktivity, užívají se v případě nutnosti provádět danou činnost delší časový úsek, aniž by bylo potřeby je zobrazovat uživateli na hlavní obrazovce. [4, 7, 9]

Tuto možnost nabízí přehrávání hudby, kdy po zavření aplikace a zamčení telefonu hudba stále nehraje, výjimky se mohou týkat některých aplikací určených k přehrávání hudby s nutností připojení k síti. [4, 7, 9]

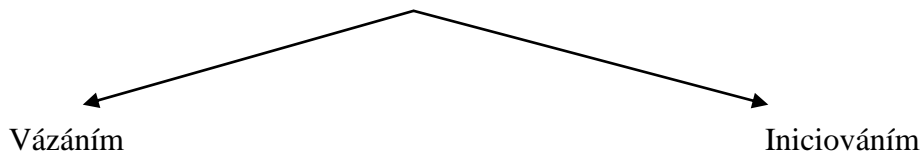
Životní cyklus se od životního cyklu aktivity velmi liší, jsou pouze tři výchozí stavy:

- **Starting** –zavolání komponenty
- **Running** – komponenta běžící na pozadí
- **Destroy** – ukončení komponenty, samostatně nebo prostřednictvím klienta



Obrázek 16 - Životní cyklus Services  
Zdroj: [29]

Komponenta služby může vzniknout dvěma způsoby:



**Vázaná:** služba zavolána metodou `bindService()` a na rozdíl od iniciované běží po dobu, po kterou je vázána na některou komponentu za účelu získání určitého výsledku.

**Iniciovaná:** služba zavolaná metodou `startService()`, běží na pozadí po nedefinovaný čas i v případě ukončení předešlé komponenty, která jí zavolala. Služba se zastaví sama po dokončení operace, kvůli které byla zavolána a nevrací výsledek volající komponentě.

Mezi třemi výchozími stavy dochází k volání následujících metod:

**onCreate ():** po prvním vytvoření služby je metoda zavolána a probíhá nastavení před voláním dalších metod `onBind()` nebo `onStart()`, dle toho zda se jedná o vázanou službu nebo iniciovanou[4, 7, 9]

**onBind():** metoda zavolaná v případě, že se jedná o službu vázanou na komponentu `bindService()`, komunikace klientů probíhá přes rozhraní `IBinder`, které je nutné definovat hodnotou, pokud službu není v úmyslu vázat je odesílána prázdná hodnota

**onStartCommand():** metoda zavolaná v případě, že se jedná o iniciační službu `startService()` a po skončení metody je zavolána `stopSelf()` metoda pro její ukončení, v případě nezavolání metody by služba běžela nekonečně dlouho nebo je zde možnost zavolat metodu `stopService()`, která je vyvolaná některou komponentou[4, 7, 9]

**onDestroy():** metoda je zavolána při ukončení služby

### 3.3.4 Intent

Intent, český překlad záměry, jsou druhem zpráv zasílané uvnitř systému mezi hlavními bloky za účelem informovat aplikace o událostech. Android disponuje výhodou, která se při těchto operacích využívá, jedná se spuštění komponenty jiné aplikace. Pokud vývojář bude potřebovat pro svou aplikaci např. zaměření polohy, nemusí vytvářet speciální aktivity, ale může využít záměru a spustit aktivity jiné aplikace, která zaměření polohy již obsahuje. [4, 7, 9]

Rozdělují se na dva druhy:

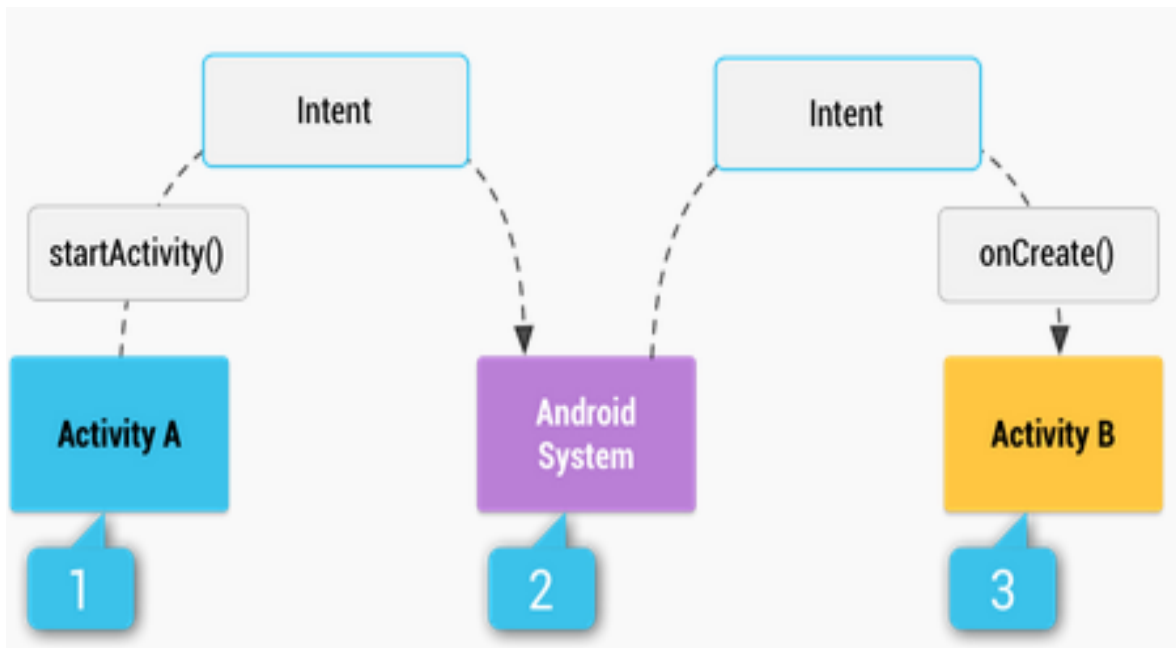
- Implicitní – určen pouze příjemce, využívána komponenta cizí aplikace
- Explicitní – určen příjemce, cílová komponenta, užívána pro vnitřní komunikace systému

Objekt záměru obsahuje vždy kategorie informací, které ho přesně definují, obsahem je:

- Jméno – pokud není dohledatelné, Android využije jiné informace k identifikování komponenty
- Kategorie – o jaký druh komponenty se jedná
- Akce – informuje, jaká akce se provede
- Vlajky – jak aktivitu zahájit
- Data – informace zpřesňující jaká data jsou objektem zájmu
- Dodatečné informace – obsahuje ostatní hodnoty sloužící k identifikaci

Na následujícím obrázku je znázorněna, jakým způsobem probíhá práce s implicitním záměrem:

1. Aktivita A vytvoří záměr s popisem akce a předá ji metodě `startActivity()`.
2. Systém Android vyhledá všechny dostupné aplikace, které splňují kritéria odpovídající záměru, provádí se přes filtr záměrů.
3. V případě shody, spustí systém odpovídající aktivitu (aktivita B) vyvoláním jeho metody `OnCreate ()` a dojde k předání záměru.



Obrázek 17 - Příklad implicitního záměru  
Zdroj: [30]

### 3.3.5 Content providers

V systému není stejné místo pro ukládání nebo načítání dat umožňující přístup pro všechny aplikace v zařízení. Od toho slouží Content provider (poskytovatel obsahu) sloužící ke sdílení dat mezi aplikacemi, uživatel například může mít nainstalováno více aplikací zabývajících se přehráváním hudby, ale ve všech aplikacích budou přítomné stejné písničky. Uvnitř systému se nachází soubor Android Provider, kde jsou zapsáni jednotliví poskytovatelé a druhy dat se kterými mohou pracovat. [4, 7, 9]

Vývojář má možnost vytvoření vlastního poskytovatele za cílem sdílení dat, která sám vytvořil.



Poskytovatelé dat se uvádějí v činnost standardními příkazy známých z programování databází, zde jsou příklady některých příkazů:

- onCreate(): inicializace Content Provideru
- insert(): vložení nového řádku do Content Provideru, argumenty se určí cílový řádek v cílové tabulce
- update(): aktualizace stávajících řádků Content Provideru, argumenty se určí cílová tabulka
- delete(): odstranění řádky z Content Provideru, argumenty se určí řádky v dané tabulce určené ke smazání
- query(): získávání dat z Content Provideru, argumenty se vybírá konkrétní řádek v konkrétní tabulce

### **3.3.6 Broadcast Receivers**

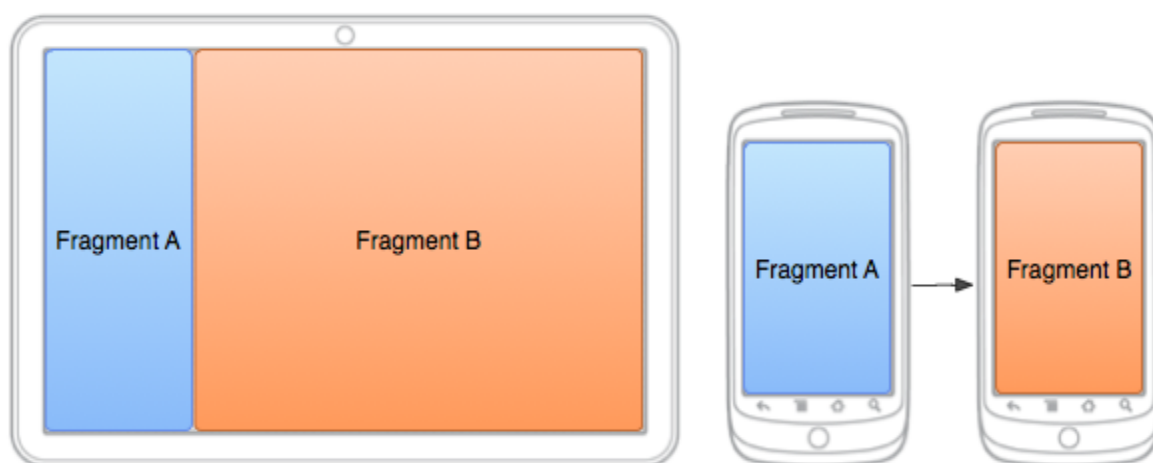
Součástí bez grafického vzhledu, která sleduje systémem vysílaná oznámení o událostech, pokud dojde ke změně systému nebo poruše. Reakce vysílače závisí na druhu a příčině oznámení, operací poté může být výpis na stavový řádek i spuštění komponenty. Vývojář má možnost využití vysílače systému nebo naprogramovat vlastní. [4, 7, 9]

Události prochází filtrem a jsou přesně definovány ve výše zmíněném souboru AndroidManifest.xml každé aplikace a jakým způsobem má daná třída reagovat. Příkladem může být upozornění o doručené sms zprávě nebo informace o zmeškaném hovoru. [4, 7, 9]

### **3.3.7 Fragmenty**

Není zařazen mezi základní stavební kameny aplikace, jsou součástí aktivit a reprezentují jeho chování. Jedna aktivita může obsahovat více fragmentů a umožňuje jejich vzájemné kombinování nebo je zde i možnost zobrazení všech fragmentů vedle sebe, výhodou je současné manipulování s větším množstvím fragmentů a ukládání těchto fragmentů do zásobníku, v případě návratu v operaci, dochází k návratu transakce fragmentu. Další výhodou spočívá ve využití jednoho fragmentu ve více aktivitách.

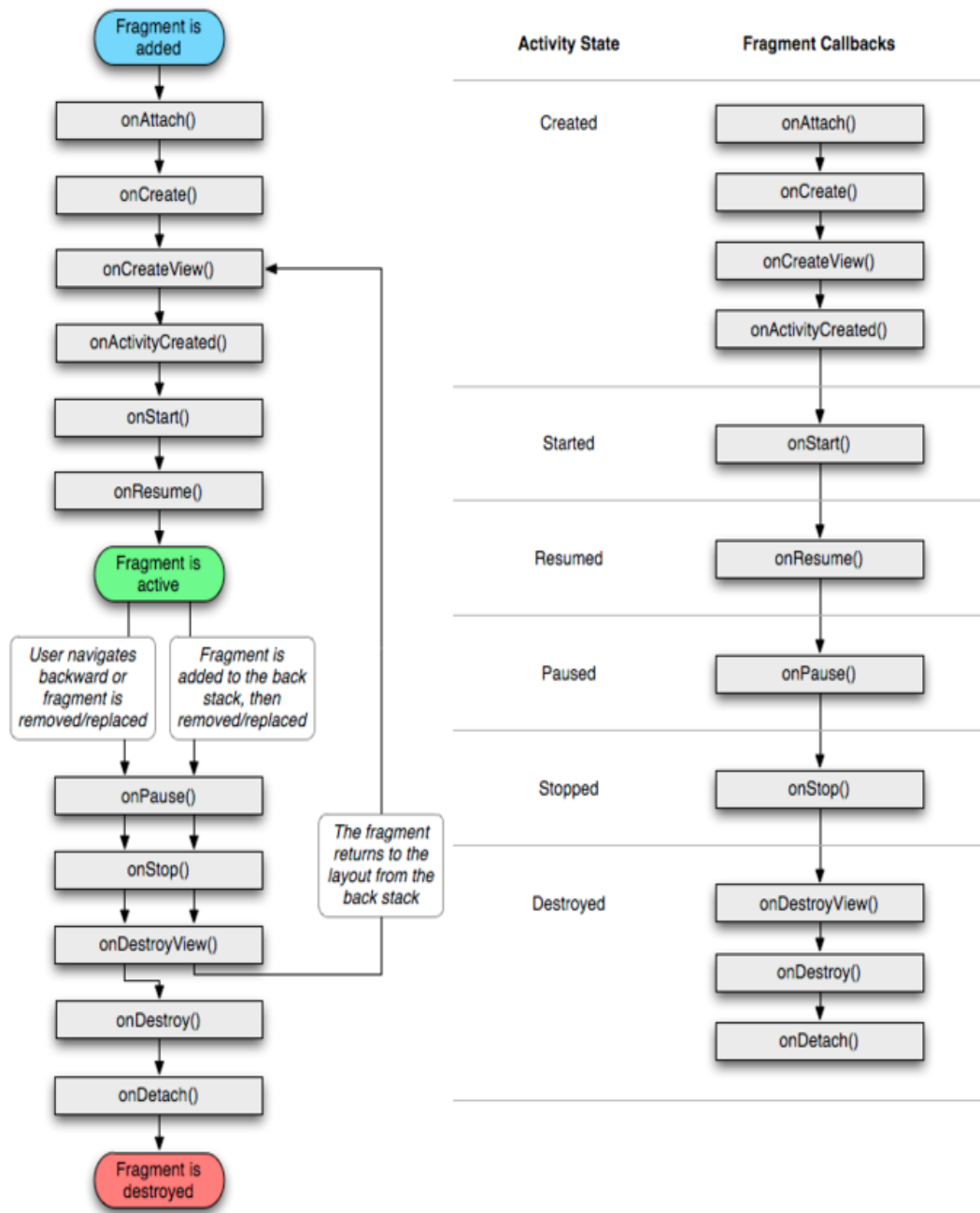
Užívá se při vytvoření responzivní aplikace pro tablety a mobilní zařízení. Obsahuje vlastní vzhled a vlastní životní cyklus, ale působí na něj životní cyklus aktivity. V případě zastavení vedoucí aktivity je zastaven i fragment a nemůže pokračovat ve své činnosti resp. všechny fragmenty, které jsou obsahem aktivity. Fragments jsou součástí systému od nepopulární verze 3.0, která byla vytvořena přímo pro tablety jako optimalizace pro podporu dynamiky uživatelského rozhraní. [4, 7, 9]



**Obrázek 18 - Využití fragmentů na tabletu a mobilním zařízení**  
Zdroj: [31]

- `onAttach()` – metoda volána při předávání argumentu od aktivity, dochází k propojení
- `onCreate ()` – vytváření fragmentu, obsahuje části pro zachování i po zastavení fragmentu
- `onActivityCreated()` metoda volána po dokončení metody `onCreate()` příslušnou aktivitou pod kterou patří
- `onCreateView()` – vykresluje fragment v rámci uživatelského rozhraní
- `onPause ()` – metoda je volána při změně fragmentu nebo změně aktivity, probíhá ukládání dat

Na obrázku umístěném pod tímto textem lze vidět rozdíl mezi životním cyklem fragmentu a životním cyklem aktivity, rozdíl je pouze při vytvoření a zničení.



Obrázek 19 - Porovnání životního cyklu aktivity a fragmentu  
Zdroj: [32]

## **3.4 Vývoj na Android**

### **3.4.1 Programovací jazyk**

Programovací jazyk Java vyvinut společností Sun Microsystems a představen roku 1995 patří mezi nejvíce využívané jazyky na světě v oblasti IT. Jde o objektově orientovaný jazyk a využívána je primárně jeho přenositelnost na různé platformy, které využila i společnost Google a jazyk Java jako základ pro tvorbu svého operačního systému. Java je využívána Androidem hlavně kvůli syntaxi, knihovny používá Android své vlastní a i přes využívání Java jazyku při programování je větší část jádra systému napsána v programovacím jazyce C. [1, 5]

Vývojáři mají možnost využívat i jiných programovacích jazyků dle svých preferencí. Google umožňuje psaní v nativním jazyce C/C++ pomocí balíčku NDK, nevýhodou však je větší obtížnost vývoje, výhoda se pak skrývá ve výkonu, kdy aplikace naprogramovaná v nativním jazyce je méně náročná na procesor. Prostřednictvím jiného nástroje od společnosti Google, SL4A, lze vytvářet skripty pro API rozhraní pomocí různých programovacích jazyků např. JRuby nebo Python. [1, 5]

Pokud se vývojář rozhodne pro vytvoření aplikace na operační systém Android, je tedy nutná znalost jazyku Java. I když, jak dříve bylo zmíněno, dochází uvnitř systému ke kompilaci, aby nebyly porušeny licenční podmínky Oracle.

### **3.4.2 Vývojové nástroje**

V případě vývoje aplikací pro operační systém Android se využívá tzv. softwarový balíček SDK, který obsahuje plugin ADT, Android Development Tools.

Tento balíček představuje sadu nástrojů, Java knihoven, dokumentace a tutoriálu, které slouží jako prostředek pro pochopení celého principu vývoje.

ADT obsahuje Android SDK Manager, který je odpovědný za stažení všech souborů, potřebných pro zvolenou verzi Androidu. [1, 5]

To znamená, stažení soborů, které souvisí s jednotlivými verzemi, na kterých chceme, aby vytvořená aplikace fungovala. Výhodou je, že samotné vývojové prostředí kontroluje jednotlivě volané metody a zda jsou dostupné pro verze, které vývojář zvolil.

Další částí SDK je Android Virtual Device. Jedná se o možnost vytvoření virtuálního zařízení, na které se otestuje vytvořená aplikace. Nabídka možností konfigurace je široká a vývojář má tak možnost otestovat aplikaci na různých zařízeních s různou hardwarovou výbavou a zjistit tak, jak se aplikace chová a zároveň nemusí být vlastníkem několika mobilních zařízení. [1, 5]

Pomocí jiné části, Android Device Manageru, lze na zařízení vytvořeném v AVD simulovat příchod SMS, pro otestování např. aplikace vytvořené pro běh na pozadí, spravující notifikační lištu a jak se bude příchozí sms zpráva zobrazovat.

Emulátor však trpí také velkou nevýhodou, v podobě nepoužitelnosti na hardwarově méně vybavených zařízeních. Řešení nabízí emulátor GenyMotion, který je méně náročný na výkon, ale zde se jedná o placenou službu, neplacená verze je ochuzena o některé důležité části v případě vývoje specifických aplikací, předešlý příklad na simulování příchodu SMS zprávy by v tomto případě nebyl možný.

Aplikace jsou většinou vyvinuty v Java programovacím jazyce a mimo SDK lze využít i další vývojová prostředí. [1, 9]

Nástroje lze rozdělit na dvě skupiny:

- SDK nástroje – potřebné bez ohledu na verzi
- Přímé nástroje platformy – na verzi Androidu v tomto případě záleží a podporují nejnovější funkce

Nástroje v podobě SDK mohou být přímo součástí některého z vývojových prostředí pomocí specifického pluginu, který přidává další nástroje. Může se jednat o oficiálně podporované vývojové prostředí společností Google, kterým dříve bylo Eclipse. Toho nyní vystřídalo Android Studio IDE, a stejně jako systém samotný je open-source nebo také neoficiální prostředí NetBeans. [4]

Některé druhy nástrojů používané vývojáři jsou představeny v následující kapitole.

### 3.4.2.1 Eclipse

V roce 2001 bylo u založení 9 firem, původně se jednalo o pouhý projekt firmy IBM, v roce 2003 už počet stoupl na 80 různých, většinou technologicky zaměřených společností o rok později se vznikla nezisková organizace Eclipse Foundation a program se tak stal dostupný široké veřejnosti bez jakýchkoliv poplatků. [6]

Dříve doporučovaný společností Google program pro vývoj aplikací na platformu Android, dnes pouze jeden z možných vývojových nástrojů. Prostředí Eclipse si uživatel může přizpůsobit ke své práci po stažení Android SDK společně s emulátorem a přidáním tak dodatečných funkcí. Okno programu podléhá základnímu rozdělení viděných u většiny programovacích nástrojů. Uprostřed je místo pro psaní kódu, nalevo navigace a prohlížeč nainstalovaných balíčků, vpravo nabídka upravení jednotlivých parametrů a dolní částí programu reporting chyb. [4]

Začínajícím vývojářům je doporučeno společností Google, stažení a nainstalování balíčku se všemi základními SDK a vývojovými nástroji označené jako balíček ADT.

Obsah ADT Bundle:

- Android SDK
- Eclipse
- ADT plugin
- Poslední vydaná verze platformy Android
- Poslední vydaná verze emulátoru od Androidu

ADT plugin rozšiřuje funkce Eclipse, aby vývojáři mohli zřídit nové projekty pro Android, vytvářet uživatelské rozhraní aplikace, přidávat balíčky založené na Android Framework API, ladit své aplikace využitím Android SDK a exportovat (podepsané nebo nepodepsané).apk soubory k distribuci svých aplikací. ADT je oficiálně zastaralý již od konce roku 2015. Nyní se Google zaměřuje na Android Studio jako oficiální Android IDE.

Právě na rozšíření funkcí pomocí pluginů Eclipse zakládá. Po instalaci obsahuje pouze prostředky pro vývoj s použitím standardní Javy, ale neobsahuje např. prostředek pro prezentaci grafického vzhledu uživatelského rozhraní a od toho tu jsou dodatečné pluginy.

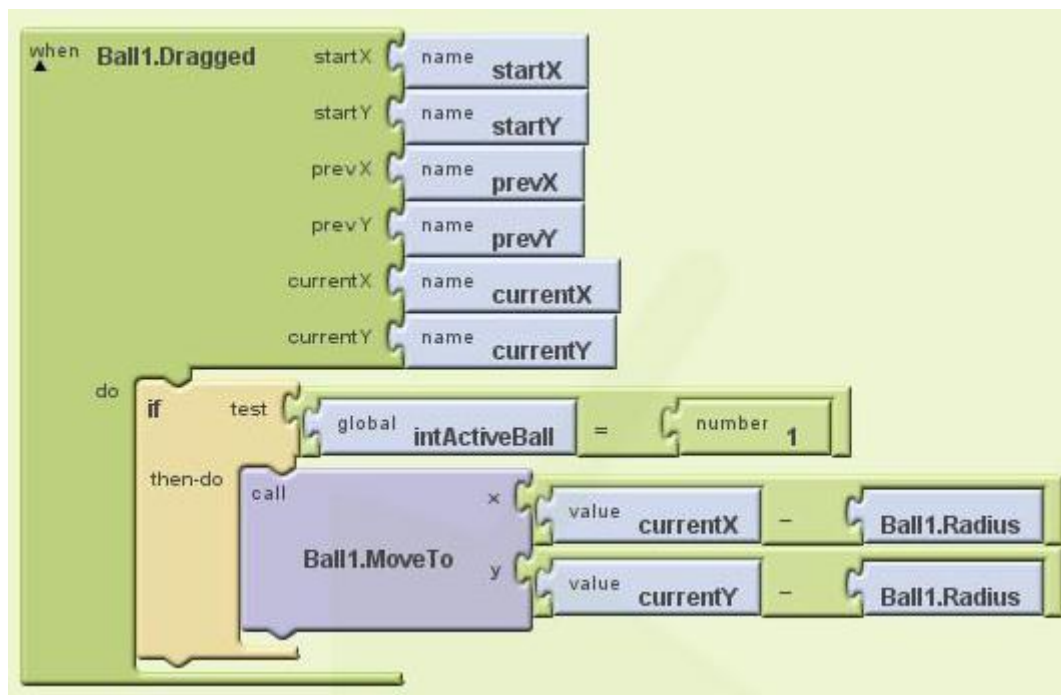
### 3.4.2.2 AppInventor

AppInventor je na první pohled netypické open-source vývojové prostředí. V tomto případě není přítomné žádné základní rozložení ovládacích prvků, jako tomu bylo u Eclipse. K vývoji aplikace využívá především grafického prostředí, skládání do sebe pasujících barevných bloků. Schválně je zmíněno pasujících, protože některé bloky nejde spojit a vývojář je tak upozorněn, že některé bloky, představující komponenty nelze kombinovat. Uživatel nemusí znát žádný z programovacích jazyků, a proto je nástroj vhodný k výuce pro začátečníky, ale s omezeným počtem komponent ( je stále ve vývoji).

Za vznikem roku 2010 stála společnost Google, která ale vývoj zastavila a vývoj prostředí ve formě projektu předala Institutu Technologie v americkém městě Cambridge. Program funguje ve webovém prostředí, běží na vzdáleném cloudu a je nutné mít založený účet od Google. Oproti tomu je k němu možné přistupovat odkudkoliv a nezáleží ani na operačním systému, který využívá vývojář, potřebuje pouze Javu a žádné jiné ovladače.

App Inventor se skládá ze tří částí:

- Designer – určuje základní možnosti aplikace pomocí komponent
- Blocks Editor – jak mají komponenty reagovat na předem definované akce
- Emulátor – k zobrazení aplikace, napsaný kód se okamžitě projeví



Obrázek 20 - Ukázka AppInventor  
Zdroj: [33]

### 3.4.2.3 Android Studio

Android Studio je založené na IntelliJ IDEA, tedy na komerčně úspěšném vývojovém prostředí pro jazyk Java. Google ve spolupráci s JetBrains projekt oficiálně představil 16. května 2013 na své vlastní konferenci. Už od začátku představení se jednalo o open-source, ale více jak dva roky bylo Android Studio ve zkušebním režimu, nicméně pořád přístupné všem uživatelům. [1]

Nejlépe lze definovat Android Studio v porovnání s jeho předchůdcem Eclipse, dříve i číslo jedna pro společnost Google, tedy do té doby než se objevilo Android Studio. Na rozdíl od Eclipse není nutné instalovat dodatečné pluginy, instalace samotného programu se tak stává rychlejší a rychlejší je program při práci. Android Studio je dobře optimalizované a přizpůsobené na vývoj aplikací, méně náročnější na operační paměť než Eclipse. [4]

Jak znázorňuje tabulka, minimální požadavky na tvorbu aplikace jsou pro operační systém Windows a Mac OS totožné s očekávanou výjimkou použitého OS:

Operační systém	Microsoft® Windows® 7/8/10 (32 nebo 64-bit) / Mac® OS X® 10.8.5 +
Operační paměť	minimum 3 GB RAM doporučeno 8 GB RAM doporučené + 1 GB RAM na emulátor
Velikost HDD	2 GB minimum volného místa 4 GB doporučené (500 MB pro IDE + 1.5 GB pro Android SDK a emulátor)
Rozlišení obrazovky	minimum 1280 x 800
Ostatní požadavky	pro rychlejší emulátor je doporučeno procesor Intel® s podporou Intel® VT-x nebo Intel® EM64T (Intel® 64)

**Tabulka 1 - Tabulka minimálních požadavků**

Zdroj: vlastní zpracování

Více o vývojovém prostředí Android Studia je napsáno v praktické části této práce.



## 3.5 Konkurence

Prodeje smartphonů porazily poprvé na začátku roku 2013 statistiky prodeje telefonů bez operačního systému. Smartphony v současné době ovládají trh s telefony, ale děje se tak pouze ve vyspělých zemích, v ostatních částech světa mluví statistiky stále v neprospěch chytrých telefonů. Momentálně je trh rozdělen mezi tři velké společnosti. Vstup nového konkurenta do odvětví je tak velice obtížný. Nejpopulárnější operační systém byl již představen, v následující kapitole budou představeni dva jeho největší konkurenti. [10, 11, 12]

### 3.5.1 iOS

Firma Apple uvedla svůj první operační systém zároveň i s uvedením prvního chytrého telefonu, stalo se tak na konferenci v roce 2007. Původně se mělo jednat o operační systém pouze pro mobilní zařízení, později byl ale rozšířen i na ostatní produkty společnosti Apple. [10]

Architekturu založenou na OS Darwin, stejně tak vydanou společností Apple a napsanou v programovacím jazyku C tvoří čtyři vrstvy:

- Media layer
- Core Service layer
- Core OS layer.
- Cocoa Touch player

Poslední jmenovaná vrstva způsobila revoluci v práci s operačním systémem, kdy jej bylo možné ovládat pomocí gest.

Jednoduchá architektura s malou náročností na operační paměť, okolo 480 MB, si brzy získala své příznivce. Po uplynutí tří let bylo na iOS vydáno více než 195 000 aplikací s celkovým počtem stažení okolo 4 miliard. Je to již skoro 10 let od vydání prvního iPhone a za tu dobu si vybuodovala a každým dnem upevňuje společnost Apple své místo mezi úspěšnými výrobci mobilních telefonů a náleží jí přes 20 % prodeje chytrých telefonů po celém světě. [11]

Od uvedení první generace zařízení iPhone uběhlo již více než 4 roky. Během nich systém získal důležité postavení na trhu mobilních operačních systémů a je použit i v ostatních produktech firmy Apple jako je tablet iPad, přehrávač iPod Touch či Apple TV. iOS náleží 24% světových prodejů chytrých telefonů. [11]

První verze neměla po určitou dobu ani vlastní jméno, to se změnilo s vydáním vlastních SDK nástrojů společností Apple a systém byl pojmenován iPhone OS a v roce 2010 došlo na přejmenování na iOS. Operační systém je využíván na zařízeních vydané výhradně společností Apple, neexistuje žádné partnerství s jiným výrobcem, jako to je v případě Androidu. [11]

Stejně jako tomu je u Androidu, má i Apple jediné oficiální místo pro stahování aplikací s názvem AppStore. Oficiálně doporučeným vývojovým nástrojem je Xcode od Applu je však dostupný pouze na Mac, pro operační systém Windows je tu neoficiální Xamarin. Pokud chce uživatel vydávat aplikace na zařízení od firmy Apple, musí se nejdříve registrovat a po schválení zaplatit roční poplatek 99 dolarů. [10, 11]



**Obrázek 21 - První vs. poslední iPhone**  
Zdroj: vlastní zpracování

### 3.5.2 Windows Phone

Operační systém společnosti Microsoft a nástupce Windows Mobile, založený na platformě Windows CE a naprogramovaný v jazyce C++. Určený je na rozdíl od svých konkurentů pouze na chytré telefony, tedy do doby, než bude vydán Windows 10, schválně bez přízviska Phone, protože se jedná o multiplatformní systém s pozměněnou architekturou. V době psaní práce nebylo možné zjistit více informací, o jaké změny se jedná. [12]

Mobilní verze systému Windows má dlouhou historii, ale první snažení ve formě Windows Mobile neskočilo pro firmu šťastně. Microsoft se k tomu přičinil sám, když chtěl po společnostech HTC a Samsung, kteří měli zájem systém ve svých zařízeních využívat, licenční poplatky za každý prodaný kus mobilního zařízení. O návrat mezi mobilní operační systémy se firma začala snažit v roce 2004, ale už během vývoje se objevily komplikace a projekt musel být pozastaven. Uvnitř firmy došlo k personálním změnám a než projekt mohl znova odstartovat, byl obměněn kompletní vývojářský tým, nová verze operačního systému pro mobilní zařízení od společnosti Windows byla představena ke konci roku 2010. Hlavními prvky grafického rozhraní jsou dlaždice, celý systém řešení vzhledu se pak nazývá Metro a je přímo odvozen od verze na stolní počítače. Microsoft má prvenství, které se jeho konkurentům jen tak nepodaří vzít. Problém související s náročností systému Android a iOS na operační paměť je známa, oba systémy využívají několik stovek MB, oproti tomu, Microsoft si vystačí pouze s jednotkami MB. [12]



Obrázek 22 - Ukázka vzhledu grafického rozhraní Windows (dlaždice)  
Zdroj: [34]

### 3.5.3 Porovnání OS Android s konkurencí

K porovnání zvolených operačních systémů bude sloužit vícekriteriální analýza variant za jistoty. Cílem této analýzy bude nalezení nejlepší varianty podle zvolených kritérií. Hodnocení každé varianty závisí na preferencích, reprezentované číselnou hodnotou, při využití metody stanovení vah kritérií, čím větší číselná hodnota u kritéria, tím je dána větší preference. V následující kapitole bude využita metoda pořadí a metoda stejné váhy.

#### 3.5.3.1 Kritéria

Stanovená kritéria, kterým se budou udělovat váhy:

- Podpora vývojářů – oficiální i neoficiální zdroje
- Vývojové prostředí – dostupnost, srozumitelnost, složitost instalace a hw nároky, stabilita
- Bezpečnost – systémové chyby v bezpečnosti
- Optimalizace – vyváženost mezi softwarem a hardwarem
- Obtížnost vývoje – hodnotí se z hlediska začínající programátora, kam až se programátor může dostat bez speciálních znalostí kódu a pouze s pomocí podpory od dané společnosti

### 3.5.3.2 Metoda pořadí

Jak název sám napovídá, každé z kritérií bude ohodnoceno číselnou hodnotou, tedy váhou, podle pořadí, které mu bylo určeno dle stanovených preferencí.

Zvolené kritérium	Preference	Hodnocení OS		
		Android	Windows Phone	iOS
Podpora vývojářů	0,25	1	3	2
Vývojové prostředí	0,2	1	2	1,5
Bezpečnost	0,35	2,5	3	1,5
Optimalizace	0,05	2,5	2	1,5
Obtížnost vývoje	0,15	1	2	2
Součet	1	1,600	2,6	1,700
Pořadí		<b>1</b>	<b>3</b>	<b>2</b>

**Tabulka 2 - Metoda pořadí**  
Zdroj: vlastní zpracování

Operační systém Android byl zvolen pomocí metody pořadí jako nejlepší OS pro mobilní telefony těsně před iOS. Kde jeden systém získává, druhý ztrácí a naopak. Z pohledu začínajícího vývojáře je podpora z třetí strany velmi důležitá a tady se musí Apple a celá komunita zlepšit. [10, 11, 12]

### 3.5.3.3 Metoda stejné váhy

Nejvíce se tato metoda využívá při nemožnosti se rozhodnout, které kritérium je nejdůležitější. Odborník zabývající se bezpečností, bude přiřazovat největší preferenci kritériu bezpečnost, naopak vývojář bude oponovat, že bez vyváženého systému a stabilního vývojového prostředí, nebude aplikace vytvořena, a tak je zabývání se bezpečností „bezpředmětné“. Pro tyto případy je zde metoda stejné váhy, která ohodnotí všechna kritéria stejným číselným vyjádřením. Celek, tedy 1, se podělí počtem kritérií a výsledek je váha, která se přiřadí všem kritériím. [10, 11, 12]

Zvolené kritérium	Preference	Hodnocení OS		
		Android	Windows Phone	iOS
Podpora vývojářů	0,2	1	3	2
Vývojové prostředí	0,2	1	2	1,5
Bezpečnost	0,2	2,5	3	1,5
Optimalizace	0,2	2,5	2	1,5
Obtížnost vývoje	0,2	1	2	2
Součet	1	1,600	2,4	1,700
Pořadí		<b>1</b>	<b>3</b>	<b>2</b>

**Tabulka 3 - Metoda stejné váhy**  
Zdroj: vlastní zpracování

Výsledek u metody stejné váhy vyšel totožně, jak s výsledným pořadím, tak mezi Androidem a iOS i v součtu číselných hodnot. Nicméně ani jedna z metod nemusí znamenat budoucí úspěch dané platformy, v tomto případě se ale průzkumy trhu a výsledky metody shodují. Ale stále se jedná o osobní preference autora práce. [10, 11, 12]

### 3.6 Využití OS Android mezi různými druhy zařízeními

Operační systém Android má velkou výhodu v přenositelnosti. Systém se pomalu rozrůstá i do ostatních odvětví elektroniky, mobilní telefony jsou nejznámějšími zařízeními vybavené tímto operačním systémem, ne však jedinými. [3, 8]

#### 3.6.1 Aplikace

Zůstaneme-li u aplikací, existuje mnoho z nich, které dokážou uživateli zjednodušit práci. Společnost Microsoft přesunula svůj nejpoblárnější kancelářský balíček přímo do mobilních zařízení, mezi verzi pro stolní počítače a té por telefony se nenajde příliš mnoho rozdílů s výjimkou ochuzení o pár funkcí. V obecném měřítku se může hovořit i o navigacích. Dříve výsada pouze speciálních zařízení k tomu určených, dnes je možné si navigaci formou aplikace stáhnout do mobilu. [7]

Výhodou je, že tyto aplikace zároveň využívají připojení na internet a tak má uživatel informace o provozu a případně má tak šanci vyhnout se zácpě. Do třetice je dobré se zmínit o aplikaci zachraňující lidské životy. Aplikace Záchranka byla vyvíjena za účelem pomoci lidem v nouzi. Disponuje totiž mimo jiné jedním naprogramovaným tlačítkem, které po stisknutí vytočí číslo rychlé záchranné služby, v případě, že se na druhé straně nikdo neohlašuje, záchranáři vyjedou směrem k pacientovi a jsou navigováni přesným GPS signálem, který aplikace zpřístupňuje. Záchranka přispívá k pomoci pacientům i nepřímo, neboť jedna z funkcí je i podrobný návod k resuscitaci člověka. [7]



Obrázek 23 - Vzhled aplikace záchranka  
Zdroj: vlastní zpracování

### 3.6.2 Android Wear

Jak již název napovídá, s nadsázkou lze říci, že se jedná o Android, který se nosí. Jedná se o velký trend dnešní doby. Jako první byly na trhu náramky určené pro sportovce. Měřily počty kroků s ušlou vzdáleností, tepovou frekvenci nebo zaznamenávaly trasu pomocí GPS a tu si následně mohl uživatel promítnout do některého z webových rozhraní resp. přímo do mapy. V době, kdy byly náramky na trhu vstoupily na něj i chytré hodinky. Ty plnily klasickou funkci a ukazovaly čas, ale jelikož jde o chytré hodinky obsahovaly i funkce navíc. Byly vybaveny totožnými možnostmi jako náramky, ale jelikož disponují i malým dotykovým displejem, má uživatel možnost odepisovat na smsky nebo zvednou hovor, aniž by musel vyndat telefon z kapsy. Osobně jedny hodinky s tímto systémem vlastním, a tak mohu podotknout, že se jedná o vydařený technologický výrobek, ale stále také slouží jako módní doplněk. Nevýhodou oproti tomu je, že k častému nabíjení mobilního telefonu, přibyla i povinnost častého nabíjení i hodinek, které v průměru vydrží pouhé 3 dny. [7]

### 3.6.3 Televize

V dnešní době není nic zvláštního narazit na televizi, která zároveň plní i funkci multimediálního zařízení, toto zajišťují primárně aplikace nainstalované do dané televize. A v případě, že uživatel není příliš hardwarově náročný k prohlížení internetových stránek a k občasnému poslechu hudby může sloužit právě televize. [7]

Není to tak dlouho co na trh přišla novinka v podobě tzv. chytrých krabiček. V jádru věci jde vlastně o mini počítač, který dokáže udělat i z „hloupé“ televize zase „chytrou. Velkou výhodou je, že ať danou krabičku připojíte k čemukoliv, dokáže z daného zařízení vytvořit počítač, což někteří uživatelé bývající často na cestách jistě ocení. [7]



### 3.6.4 Ostatní elektronika

Nemusí to být však na první pohled tak nápadná zařízení jako jsou televize nebo hodinky, které se mohou řadit do tzv. chytré domácnosti. Technologická vyspělost, dnešní doby, umožňuje rozšířit funkce, pomocí operačního systému Android, téměř každého zařízení v domácnosti. Lednička sledující čerstvost potravin není nic výjimečného, sám vlastním lednici, která při poruše nebo jakémkoliv jiném problému dokáže pomocí bluetooth zaslat na mobilní telefon zprávu informující o potížích. Stavějí se i chytré domy, kde kompletní rozvod vody, tepla nebo elektřiny je pod správou domu. Může šetřit i energie samotné, kdy se pomocí nastavení regulovat teplotu v domě, když obyvatelé nejsou zrovna přítomni. [7]

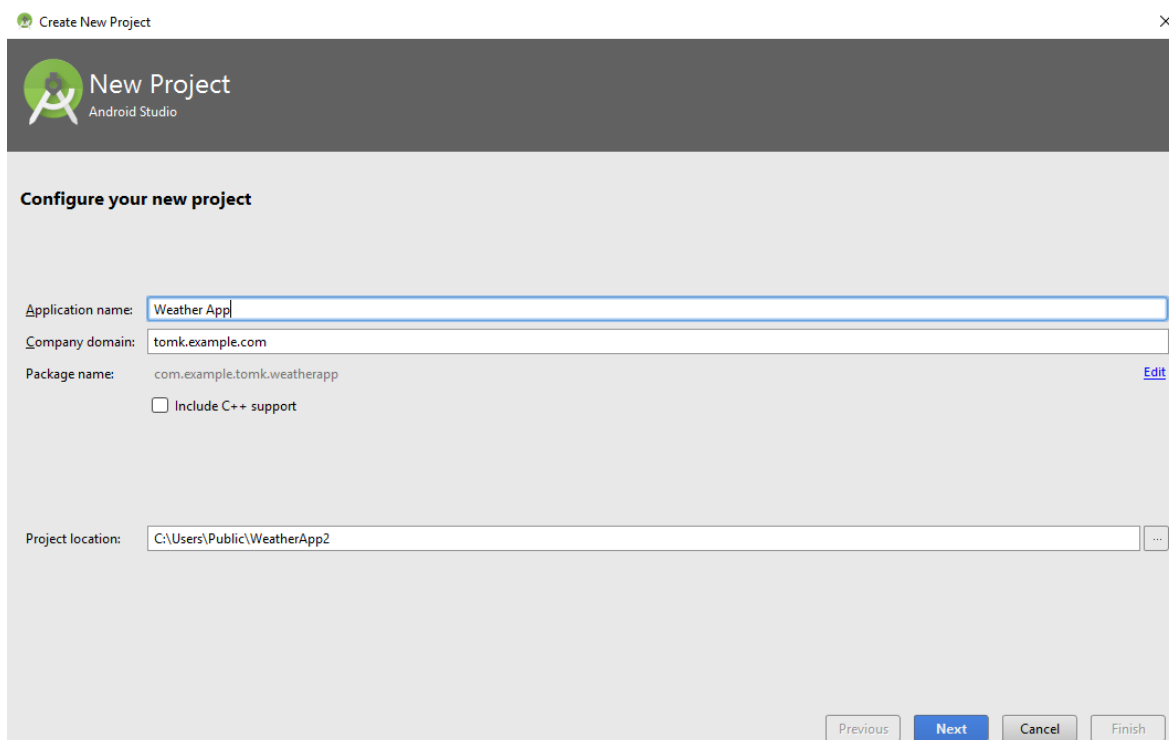
Operační systémy nejsou dnes výsadou jenom stolních počítačů nebo mobilních telefonů a je jenom otázkou času, kdy tato technologie převezme i lidskou činnost např. v podobě uklízení nebo sekání trávníku, i když, to už se děje. [7]

## 4 Vlastní práce

### 4.1 Aplikace počasí

Následující část diplomové práce se zabývá naprogramováním jednoduché aplikace počasí v Android Studiu s přispěním krátkého komentáře. Vzhledem k nulovým zkušenostem s programováním pro platformu Android, bylo zvolení typu vývojového prostředí Android Studia, otázkou preferencí samotné komunity a také z důvodu, že se jedná o oficiální nástroj společnosti Google. To se ukázalo nakonec jako chyba, ale to bude předmětem kapitoly diskuze.

Po spuštění Android Studia a zvolením možnosti založení nového projektu, uvítá uživatele standartní nabídka s pojmenováním vytvářené aplikace a zvolením místa na disku, kam bude průběžně ukládán. Další okno, které přivítá uživatele je zvolení minimálního API rozhraní, to znamená, na kterou nejstarší verzi operačního systému půjde spustit vytvořená aplikace. Nabídka je od nejnovější vydané verze systému až po verzi 4.3 s názvem Jelly Bean. Poslední okno, před samotným programováním se týká základního vzhledu layoutu, na výběr je z několika možností, dle toho o jakou aplikaci se bude jednat.



Obrázek 24- První okno po spuštění programu  
Zdroj: vlastní zpracování

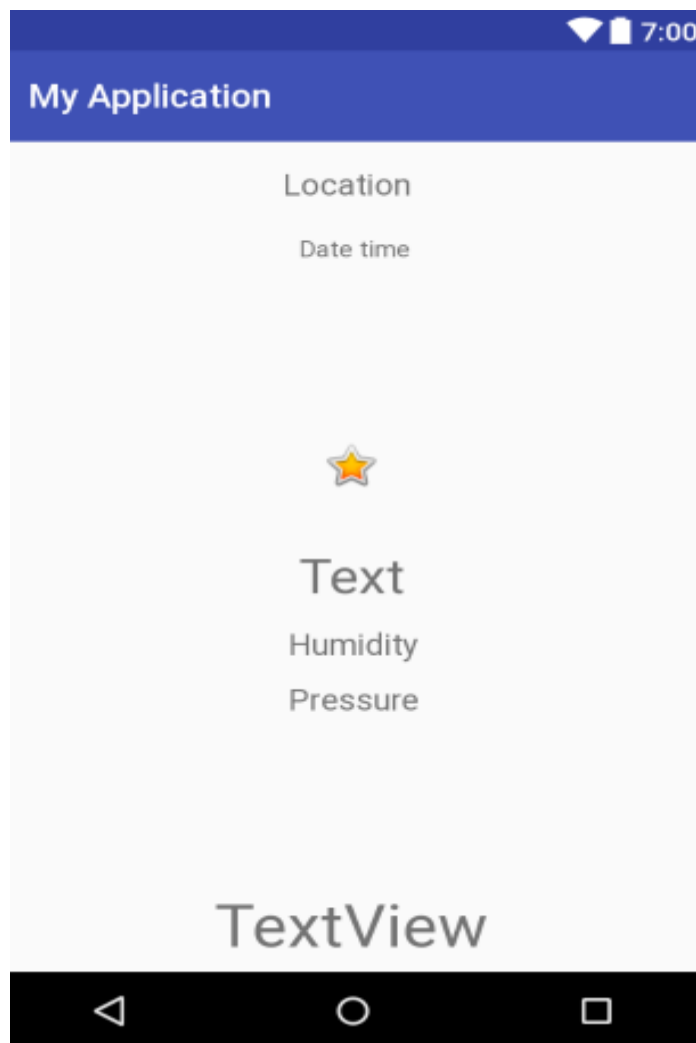
## 4.2 Grafický vzhled

První logickou volbu je vytvoření základního grafického vzhledu a poskládání jednotlivých prvků na které se při programování operací může uživatel odkazovat dle zvolených id prvků.

Menu ze kterého je možné objekty vybírat obsahuje dvanáct různých kategorií a ty další podkategorie, např. text obsahuje datum, čas, číslo atd. Emulátor ve kterém probíhají operace s grafickými prvky si může uživatel zvolit ze základní nabídky nebo dodatečně stáhnout. Vývojové prostředí nabízí samotné zobrazení uživatelského prostředí dvěma způsoby. Jedním je již představené, čistě grafické prostředí a tím druhým způsobem je pomocí XML kódu. Z pohledu programátora je lepší využít možnost XML, protože se definuje přesné místo, kde se prvky budou nacházet pomocí vzájemných vztahů. Obsahem každého prvku v XML jsou jeho základní vlastnosti jako je velikost a šířka, důležité je v tomto případě id jméno, které se využívá pro popis chování objektu.

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context="com.example.tomk.weatherapp.MainActivity">
    <TextView
        android:id="@+id/city_field"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true"
        android:textColor="#FFFFFF"
        android:textAppearance="?android:attr/textAppearanceLarge" />
    <TextView
        android:id="@+id/updated_field"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@+id/city_field"
        android:layout_centerHorizontal="true"
        android:textColor="#FFFFFF"
        android:textAppearance="?android:attr/textAppearanceMedium"
        android:textSize="13sp" />
```

Obrázek 25 - XML zobrazení  
Zdroj: vlastní zpracování



Obrázek 26 - Grafické znázornění rozložení prvků  
Zdroj: vlastní zpracování

### 4.3 Manifest

Jde o nejdůležitější soubor celého programu, nachází se v kořenovém adresáři a definuje jaká práva bude aplikace mít a její celkové nastavení. V případě ukázky se zde jedná pouze o práva na přístup k internetové síti.

```

<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.androstock.myweatherapp" >

    <uses-permission android:name="android.permission.INTERNET"/>

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:theme="@style/AppTheme" >
        <activity
            android:name="MainActivity"
            android:label="@string/app_name" >
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>

```

Obrázek 27 – Manifest  
Zdroj: vlastní zpracování

## 4.4 Použité API rozhraní

V jedné z předešlých kapitol bylo zmíněno využití různých API rozhraní, pro usnadnění práce, aby vývojář dané API nemusel naprogramovat sám. V případě ukázkové aplikace bylo využito pro rozhraní pro předpověď počasí od OpenWeatherMap. Služba nabízí několik možností na výběr, některé z nich jsou zobrazeny na obrázku dole .

### Current weather data

[API doc](#) [Subscribe](#)

- Access current weather data for any location including over 200,000 cities
- Current weather is frequently updated based on global models and data from more than 40,000 weather stations
- Data is available in JSON, XML, or HTML format
- Available for Free and all other paid accounts

### 5 day / 3 hour forecast

[API doc](#) [Subscribe](#)

- 5 day forecast is available at any location or city
- 5 day forecast includes weather data every 3 hours
- Forecast is available in JSON and XML
- Available for Free and all other paid accounts

### 16 day / daily forecast

[API doc](#) [Subscribe](#)

- 16 day forecast is available at any location or city
- 16 day forecast includes daily weather
- Forecast is available in JSON and XML
- Available for Developer, Professional and Enterprise accounts

Obrázek 28 - Nabídka API rozhraní  
Zdroj: [35]

## 4.5 Main Activity

Main Activity lze přirovnat k jádru celého programu. Nacházejí se zde všechny logické operace pro plnění účelu, pro který byla aplikace vyvinuta. Byl zde navíc nainportován externí soubor pocházející z webových stránek <http://erikflowers.github.io>. V zapsaném kódu se soubor „tváří“ jako vzhled písma, nicméně jde o ikony počasí, které se budou nacházet v grafickém návrhu místo hvězdičky-ImageView. Dohromady soubor obsahuje dvanáct různých ikon měnící se dle toho, jaká bude naměřená teplota.

```
import android.graphics.Typeface;
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.text.Html;
import android.widget.TextView;

public class MainActivity extends AppCompatActivity {
    |

    TextView cityField, detailsField, currentTemperatureField, humidity_field, pressure_field, weatherIcon, update

    Typeface weatherFont;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        getSupportActionBar().hide();
        setContentView(R.layout.activity_main);

        weatherFont = Typeface.createFromAsset(getAssets(), "fonts/weathericons-regular-webfont.ttf");

        cityField = (TextView) findViewById(R.id.city_field);
        updatedField = (TextView) findViewById(R.id.updated_field);
        detailsField = (TextView) findViewById(R.id.details_field);
        currentTemperatureField = (TextView) findViewById(R.id.current_temperature_field);
        humidity_field = (TextView) findViewById(R.id.humidity_field);
        pressure_field = (TextView) findViewById(R.id.pressure_field);
        weatherIcon = (TextView) findViewById(R.id.weather_icon);
        weatherIcon.setTypeface(weatherFont);

        Function.placeIdTask asyncTask =new Function.placeIdTask(new Function.AsyncResponse() {
            public void processFinish(String weather_city, String weather_description, String weather_temperature,

                cityField.setText(weather_city);
                updatedField.setText(weather_updatedOn);
                detailsField.setText(weather_description);
                currentTemperatureField.setText(weather_temperature);
                humidity_field.setText("Humidity: "+weather_humidity);
                pressure_field.setText("Pressure: "+weather_pressure);
                weatherIcon.setText(Html.fromHtml(weather_iconText));

            }
        });
        asyncTask.execute("25.180000", "89.530000");
    }
}
```

Obrázek 29 - Main Activity  
Zdroj: vlastní zpracování

## 4.6 Výsledná aplikace

Výsledná aplikace je funkční a slouží pouze jako jednoduchá ukázka práce ve vývojovém prostředí Android Studio. Aplikace promítá na hlavní obrazovku aktuální teplotu společně vlhkostí a tlakem. Malým doplňkem je aktuální datum a čas.

Náročnost aplikace, byla úměrná zadání diplomové práce, kdy naprogramování sloužilo pouze k určité prezentaci a nastínění programování v jazyce Java a vývojovém prostředí Android Studio. Ze stejného důvodu pak, nebyla aplikace vložena na Google Play, protože slouží pouze pro vlastní potřebu.



Obrázek 30 - Výsledný vzhled aplikace  
Zdroj: vlastní zpracování

## 5 Výsledky a diskuze

Android vstupoval na trh, kdy konkurence byla v silném postavení a především znalosti a zkušenosti z vývoje aplikací pro mobilní telefony, udávaly náskok před společností Android. I přes nesnadné začátky dokázala získat postavení na trhu, a především jméno v historii mobilních operačních systémů. Po letech se společnost stala nejpoužívanější platformou na světě. Zásluhou této skutečnosti stojí za společností široký zástup fanoušků a především vývojářů, kteří si jsou vědomi oblíbenosti Androidu a tím i důležitosti své práce.

Začínajícím vývojáři by měli vyzkoušet nejdříve App Inventor, aby poznaly základní principy programování a toho, jak která komponenta souvisí s jinou. V případě zájmu o programování je tu možnost přestoupit na těžší obtížnost v podobě vývojového prostředí Eclipse.

Práce v Android Studiu je více obtížná. Prostředí, které nahradilo právě Eclipse a je doporučováno přímo společností Google, nefunguje úplně bez chyby. Na vině je nejspíš stále špatná optimalizace jako před dvěma lety, kdy Android Studio opustilo beta testování a stalo se plnohodnotným vývojovým prostředím. Oficiální nároky na systém Windows hovoří o minimální operační paměti 3 GB a doporučené 8 GB. V případě uposlechnutí doporučeného hardwaru se však nic nemění. Systém je velice pomalý a znovuootevření starých projektů trvá velmi dlouho. Občas bývá nestabilní a první věc na seznamu oprav by mělo být povolení, ukládat soubory SDK do složek obsahujících diakritiku. Častá chybová hlášení o neplnění pravidel o pojmenování složky dle ASCII tabulky je nepochopitelné. Emulátory ihned přístupné po nainstalování jsou stejně jako elý program špatně optimalizovány a bez důvodu hlásí o nepřítomnosti souborů daného simulátoru i přes to, že v poli vedle názvu emulátoru je napsaná, že emulátor je nainstalovaný. Poslední a o to vážnější chybou se stává nepochopitelné hlášení chyb během samotného programování. Po restartu Android Studia tyto chybová hlášení zmizí. Momentálně je v prostředí Android Studia nepohodlné programovat a společnost Google by se měla zamyslet nad tím, zda by nebylo vhodné doporučit svým zákazníkům program, který opravdu funguje. Android Studio bylo dlouhá léta ve vývoji beta testování a vzhledem k problémům, které program přináší by se mělo věnovat více času na jeho vylepšení, hlavně v oblasti stability a náročnosti na výkon.



Dílčí cíl charakterizovat požadavky na aplikace pro operační systém Android byl splněn v teoretické části, s tím se podařilo splnit i druhý dílčí cíl v podobě využití systému v různých pohledech, kterými se zabývala poslední kapitola teoretické části.

Hlavní cíl charakterizovat vývoj aplikace, následnou distribuci a uplatnění na trhu byl splněn pomocí odborné literatury a praktického vyzkoušení programování ve vývojovém prostředí pro OS Android.

## 6 Závěr

Diplomová práce představuje operační systém Android jak z pohledu běžného uživatele, tak z pohledu vývojáře. V teoretické části je úvodní kapitolou představena historie mobilních OS jako celku. Práce pokračuje zaměřením na platformu Android, začínající historií a postupným vývoji systému reprezentující jednotlivé verze a dále zaměřením na vnitřní architekturu systému a zároveň i architekturu aplikace běžící na tomto systému., nebylo opomenuto ani důležitého tématu bezpečnosti. V závěru teoretické části byly uvedeny možnosti vývoje aplikací pro OS Android, představením nejznámějších vývojových prostředí a také byl umožněn pohled na největší konkurenty společnosti Android, kdy došlo i k jejich přímému srovnání využití vícekriteriální analýzy. Poslední kapitola teoretické rešerše splňuje i jeden z dílčích cílů, kterým bylo charakterizování různých pohledů na využití operačního systému Android.

V praktické části práce byly využity nově nabyté vědomosti a znalosti o problematice vývoje aplikací z předešlých kapitol. Obsahem je především ukázkový vývoj jednoduché aplikace v prostředí Android Studio za použití jazyku Java pro operativní část aplikace a značkovacího jazyku XML pro vytvoření grafického, uživatelského rozhraní. Všechny dílčí cíle byly splněny, hlavní cíl charakterizovat vývoj aplikací, jejich distribuci a uplatnění na trhu, byl také splněn.

## 7 Seznam použitých zdrojů

[1] MEDNIEKS, Zigurd R. *Programming Android*. Sebastopol, Calif.: O'Reilly, 2011. ISBN 978-1-449-38969-7.

[2] LACKO, Luboslav. *Vývoj aplikací pro Android*. Brno: Computer Press, 2015. ISBN 978-80-251-4347-6.

[3] DUBEY, Abhishek a Anmol MISRA. *Android security: attacks and defenses*. Boca Raton: CRC Press, c2013. ISBN 978-1-4398-9646-4.

[4] MURPHY, Mark L. *Android 2: průvodce programováním mobilních aplikací*. Brno: Computer Press, 2011. ISBN 978-80-251-3194-7.

[5] ALLEN, Grant. *Android 4: průvodce programováním mobilních aplikací*. Brno: Computer Press, 2013. ISBN 978-80-251-3782-6.

[6] HUDDLESTON, Rob. *Android fully loaded*. ISBN 978-0-470-93002-1.

[7] VÁVRŮ, Jiří a Miroslav UJBÁNYAI. *Programujeme pro Android*. 2., rozš. vyd. Praha: Grada, 2013. Průvodce (Grada). ISBN 978-80-247-4863-4.

[8] MEIER, Reto. *Professional Android 4 application development*. Updated for Android 4. Indianapolis: John Wiley, 2012. ISBN 978-1-118-10227-5.

[9] HERODEK, Martin. *333 tipů a triků pro Android: [sbírka nejužitečnějších postupů a řešení]*. Brno: Computer Press, 2014. ISBN 978-80-251-4310-0.

[10] VAN DE PUT, Peter. *Professional ios programming*. ISBN 978-1-118-66113-0.

[11] BRANNAN, James A. a Blake. WARD. *IOS SDK programming: a beginner's guide*. New York: McGraw-Hill, c2011. ISBN 978-0-07-175908-3.

[12] LACKO, Ľuboslav. *Vývoj aplikací pro Windows 8.1 a Windows Phone*. Brno: Computer Press, 2014. ISBN 978-80-251-3822-9.

[13] Friesen J., Smith D. *Android Recipes: A Problem-Solution Approach*. Apress Media, 2011. 442 s. ISBN 978-1-4302-3413-5.

[14] Meier R. *Professional Android Application Development*. Indianapolis: Wiley Publishing, 2009. 409 s. ISBN 978-0-470-34471-2.

[15] HTC Dream specs and review [Internet]. [cit. 2017-02-22]. Dostupné z: <http://phonesdata.com/en/smartphones/htc/dream-397/>

[16] The evolution of mobile phones [Internet]. [cit. 2017-02-22]. Dostupné z: <http://easytechnow.com/learn-technology/the-evolution-of-mobile-phones/>

[17] Smartphone at 20: IBM Simon to iPhone6 [Internet]. [cit. 2017-02-22]. Dostupné z: <http://www.telegraph.co.uk/technology/mobile-phones/11037661/Smartphone-at-20-IBM-Simon-to-iPhone-6.html>

[18] T-mobile MDA Pocket PC phone review [Internet]. [cit. 2017-02-22]. Dostupné z: <http://www.brighthand.com/phonereview/t-mobile-mda-pocket-pc-phone-review/>

[19] Nokia Xpressmusic 5800 [Internet]. [cit. 2017-02-22]. Dostupné z: [http://www.gsmarena.com/nokia\\_5800\\_xpressmusic-pictures-2537.php](http://www.gsmarena.com/nokia_5800_xpressmusic-pictures-2537.php)

[20] Avoiding BlackBerrys Fate: How Apple Could End Up In a Similar Position [Internet]. [cit.2017-02-22]. Dostupné z: <https://tech.slashdot.org/story/16/05/23/1046206/avoiding-blackberrys-fate-how-apple-could-end-up-in-a-similar-position>

[21] Gallery [Internet]. [cit. 2017-02-22]. Dostupné z: <http://keywordsuggest.org/gallery/201812.html>

[22] Did you know reason behind the Android version names! [Internet]. [cit. 2017-02-22]. Dostupné z: <http://www.graptel.com/uncategorized/did-you-know-reason-behind-the-android-version-names/>

[23] Android 2.0 review: almost human [Internet]. [cit. 2017-02-22]. Dostupné z: <http://gizmodo.com/5395801/android-20-review-almost-human>

[24] Android architektura [Internet]. [cit. 2017-02-22]. Dostupné z: <http://www.w3big.com/pl/android/android-architecture.html>

[25] Architecture of Linux Kernel [Internet]. [cit. 2017-02-22]. Dostupné z: <https://www.slideshare.net/dominiquec/architecture-of-the-linux-kernel-1851975>

[26] Android Platform: Native Layer [Internet]. [cit. 2017-02-22]. Dostupné z: <http://www.edc4it.com/blog/mobile/what-do-you-know-about-android-runtime.html>

[27] Google Play [Internet]. [cit. 2017-02-22]. Dostupné z: <http://www.svetandroida.cz>

[28] Perform operation right after showing an activity in Android [Internet]. [cit. 2017-02-22]. Dostupné z: <http://stackoverflow.com/questions/17354396/perform-operation-right-after-showing-an-activity-in-android>

[29] Android fundamentals and components [Internet]. [cit. 2017-02-22]. Dostupné z: <http://techblogon.com/android-fundamentals-components-activity-service-broadcast-receiver-content-provider/>

[30] Android launch activity title displayed in recently opened applications instead of App name [Internet]. [cit. 2017-02-22]. Dostupné z: <http://stackoverflow.com/questions/22791918/android-launch-activity-title-displayed-in-recently-opened-applications-instead>

[31] FragmentManager multiple on-screen fragments ? [Internet]. [cit. 2017-02-22]. Dostupné z: <http://stackoverflow.com/questions/14473202/fragmentmanager-multiple-on-screen-fragments>

[32] What causes a fragment to get detached from an Activity ? [Internet]. [cit. 2017-02-22]. Dostupné z: <http://stackoverflow.com/questions/33742646/what-causes-a-fragment-to-get-detached-from-an-activity>

[33] [App Inventor Classic Code Snippets Overview Internet]. [cit. 2017-02-22]. Dostupné z: <http://puravidaapps.com/snippets1.php>

[34] Budeme ve Windows 10 pracovat s dlaždicemi novým způsobem ? [Internet]. [cit. 2017-02-22]. Dostupné z: <https://mobilizujeme.cz/clanky/budeme-ve-windows-10-pracovat-s-dlazdicemi-novym-zpusobem>

[35] Rozhraní API [Internet]. [cit. 2017-02-22]. Dostupné z: <http://openweathermap.org/api>

[36] Mobilní platformy: historie a současnost [Internet]. [cit. 2017-02-22]. Dostupné z: <http://www.svetmobilne.cz/mobilni-platformy-historie-a-soucasnost/1926-24>

[37] Historie operačních systémů: Od UNIXu až po Chrome OS [Internet]. [cit. 2017-02-22]. Dostupné z: <https://www.cnews.cz/historie-operacnich-systemu-od-unixu-az-po-chrome-os/>

[38] Operační systém v telefonu aneb nahlédněte do světa smartphonů [Internet]. [cit. 2017-02-22]. Dostupné z: <https://mobilizujeme.cz/clanky/operacni-system-v-telefonu-aneb-nahlednete-do-sveta-smartphonu-vedecke-okenko>

## **8 Přílohy**

CD se zdrojovým kódem aplikace

## **9 Seznam zkratek**

IDE - Integrated Development Environment)

AVD- Android Virtual Device

SDK – Systém Development Kit

MHz – jednotka výkonu procesoru

PDA - Personal Digital Assistant

MDA – Mobile Digital Assistant

API – grafické rozhraní

HTML -programovací jazyk webových stránek

NFC - Near Field Communication

RAM – jednotka operační paměti

DVM – DalVik Virtual Machine

ADT – Android Development Tools