

**Česká zemědělská univerzita v Praze**

**Provozně ekonomická fakulta**

**Katedra informačního inženýrství**



**Bakalářská práce**

**Mobilní aplikace na tvorbu QR kódů**

**Arakelov Vsevolod**

© 2018 ČZU v Praze



# ČESKÁ ZEMĚDĚLSKÁ UNIVERZITA V PRAZE

Provozně ekonomická fakulta

## ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Vsevolod Arakelov

Informatika

Název práce

**Mobilní aplikace na tvorbu QR kódů**

Název anglicky

**Mobile Application for QR code creation**

---

### Cíle práce

Cílem práce je vytvořit mobilní aplikaci v prostředí Android, která vytvoří reklamní QR kód podle zadaných textových dat a výsledný obrázek pošle ve formátu PNG na e-mailovou adresu.

### Metodika

Nejprve bude vypracována analýza technologie QR kódů a možností jejich kombinace s bodovou grafikou. Potom bude vypracován návrh mobilní aplikace podle zásad Agilního programování. Dokumentace bude dodržovat standardy UML. Výsledná aplikace bude otestována a bude dostupná ze serveru Google Play.

### **Doporučený rozsah práce**

30-40 stran

### **Klíčová slova**

QR kod, mobilní aplikace, Android, grafika v reklamě

---

### **Doporučené zdroje informací**

CIZEK, Jakub. Zapomente na skarede cernobile QR kody. Zive [online]. 2013 [cit. 2017-05-06]. Dostupne z: <http://www.zive.cz/clanky/zapomente-na-skarede-cernobile-qr-kody/sc-3-a-168530/default.aspx>

DABNER, David, Sandra STEWART a Eric ZEMPOL. Skola grafickeho designu: principy a praxe grafiky.

V Praze: Slovart, 2015. ISBN 978-80-7391-894-1

БУГАЕВ, Л. Мобильный маркетинг: Как зарядить свой бизнес в мобильном мире. первое издание.

Москва: ООО «Альпина Паблицер», 2012. ISBN 978-5-9614-2222-1

ДЕЙТЕЛ, П., Х. ДЕЙТЕЛ a М. МОРГАНО. Android для разработчиков. Второе издание.

Санкт-Петербург: Питер, 2015. ISBN 978-5-496-01517-2

КОЛИСНИЧЕНКО, Д.Н. Программирование для Android 5. Первое издание. Санкт-Петербург:

”БХВ-Петербург”, 2015. ISBN 978-5-9775-3548-9

---

### **Предбěžný termín obhajoby**

2017/18 LS – PEF

### **Vedoucí práce**

doc. Ing. Vojtěch Merunka, Ph.D.

### **Garantující pracoviště**

Katedra informačního inženýrství

---

Elektronicky schváleno dne 11. 1. 2018

**Ing. Martin Pelikán, Ph.D.**

Vedoucí katedry

Elektronicky schváleno dne 11. 1. 2018

**Ing. Martin Pelikán, Ph.D.**

Děkan

V Praze dne 12. 02. 2018

---

### **Čestné prohlášení**

Prohlašuji, že svou bakalářskou práci "Mobilní aplikace na tvorbu QR kódů" jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou citovány v práci a uvedeny v seznamu literatury na konci práce. Jako autor uvedené bakalářské práce dále prohlašuji, že jsem v souvislosti s jejím vytvořením neporušil autorská práva třetích osob.

V Praze dne 14.03.2018

---

## **Poděkování**

Rád bych touto cestou poděkoval doc. Ing. Vojtěchu Meruňkovi, Ph.D. za zajímavé téma a vedení při zpracování bakalářské práce. Dále bych chtěl poděkovat za podporu a zázemí celé mé rodině.

# Mobilní aplikace na tvorbu QR kódů

## Abstrakt

Tato bakalářská práce se zabývá vytvořením nativní aplikace běžící na platformě Android, která umožní uživateli vytvořit QR kód podle zadaných textových dat, tento grafický kód upravit a buď ho uložit ve formátu PNG, nebo ho odeslat na e-mailovou adresu. Teoretická část této práce vysvětluje samotnou technologii QR kódu, účelnost v marketingu, možnost kombinování s bodovou grafikou a porovná ho s dalšími dvojdimenzionálními kódy. Popsán bude také operační systém Android, jeho architektura a nezbytné komponenty a znalosti pro vývoj mobilní aplikace. Praktickou část tvoří návrh aplikace, postup při její realizaci a rozmístění na Google Play Marketu.

**Klíčová slova:** QR kód, mobilní aplikace, Android, grafika v reklamě

# **Mobile Application for QR code creation**

## **Abstract**

This bachelor's thesis is focused on creating a native mobile application for Android. The application will allow the user to create a QR code, to edit this code from graphics side, to save it in PNG format or share it by email. The theoretical part of this thesis explains QR Code technology itself, efficiency in marketing, compatibility with raster graphics and comparing to other two-dimensional codes. Also, it describes the Android operating system, architecture of this system and necessary components for the development of mobile application. Practical part consists designing the application, implementing it and placing on the Google Play Market.

**Keywords:** QR code, mobile application, Android, graphics in marketing.



# Obsah

<b>1 Úvod</b> .....	<b>12</b>
<b>2 Cíl práce a metodika</b> .....	<b>13</b>
2.1 Cíl práce .....	13
2.2 Metodika .....	13
<b>I. Literární rešerše</b> .....	<b>14</b>
<b>3 Dvourozměrné kódy</b> .....	<b>15</b>
3.1 Víceúrovňové kódy .....	15
3.1.1 PDF417 .....	15
3.2 Maticové kódy.....	16
3.2.1 Aztec Code.....	16
3.2.2 MaxiCode.....	16
3.2.3 Data Matrix .....	17
<b>4 QR Code</b> .....	<b>19</b>
4.1 Úrovně korekce chyb .....	19
4.2 Velikost QR kódu.....	22
4.3 Druhy QR kódu .....	23
4.3.1 Micro QR .....	23
4.3.2 iQR.....	24
4.3.3 SQRC .....	25
4.4 Generování QR kódů.....	25
4.5 Čtení QR kódů.....	26
4.6 Obsah QR kódů .....	26
4.6.1 vCard.....	26
4.6.2 Konfigurace Wi-Fi .....	27
4.6.3 Ostatní možnosti .....	27
4.7 Porovnání 2D kódů .....	28
4.8 QR Code a mobilní marketing .....	29
4.9 QR Code a grafika.....	30
<b>5 OS Android</b> .....	<b>32</b>
5.1 Přehled verzí Android .....	32
5.2 Architektura.....	34
5.2.1 Linux Kernel .....	34
5.2.2 Libraries .....	34
5.2.3 Android runtime.....	34
5.2.4 Application Framework .....	35

5.2.5	Applications .....	35
5.3	Android z pohledu aplikace.....	35
5.4	Vývojové prostředí .....	36
5.5	Vývoj aplikace.....	37
5.5.1	Základní komponenty aplikace .....	37
5.5.2	Bezpečnost a oprávnění .....	37
5.5.3	Android Manifest.xml .....	38
5.5.4	R.java .....	38
5.5.5	Layout .....	38
5.5.6	Životní cyklus aktivity .....	38
<b>II.</b>	<b>Projekt.....</b>	<b>40</b>
<b>6</b>	<b>Zadání.....</b>	<b>41</b>
6.1	Návrh struktury aplikace .....	41
6.1.1	Aktivity .....	41
6.1.2	Třídy.....	43
6.2	Implementace aplikace .....	43
6.2.1	QRcreate.....	43
6.2.2	MainActivity .....	44
6.2.3	VizitkaActivity.....	44
6.2.4	TelefonActivity .....	45
6.2.5	SMS Activity.....	46
6.2.6	TextActivity .....	46
6.2.7	UrlActivity .....	46
6.2.8	WiActivity.....	47
6.2.9	SuccessActivity .....	47
6.2.10	CustomizeActivity .....	48
6.3	Testování .....	51
<b>7</b>	<b>Zhodnocení výsledku.....</b>	<b>53</b>
<b>8</b>	<b>Závěr.....</b>	<b>54</b>
<b>9</b>	<b>Seznam použitých zdrojů.....</b>	<b>55</b>
<b>10</b>	<b>Přílohy .....</b>	<b>57</b>

## Seznam obrázků

Obrázek 1 - PDF417 .....	15
Obrázek 2 – Aztec Code .....	16
Obrázek 3 – MaxiCode .....	17
Obrázek 4 – Data Matrix .....	18
Obrázek 5 – Schéma QR kódů.....	19
Obrázek 6 – Poškozené QR kódy .....	20
Obrázek 7 – Modely QR kódů [qrcode.com, 2018] .....	22
Obrázek 8 – Rozdíl Micro QR Code a QR Code [qrcode.com, 2018] .....	24
Obrázek 9 – iQR Code [qrcode.com, 2018] .....	25
Obrázek 10 – QR Art[Geektimes.com, 2012] .....	30
Obrázek 11 – JESS3 [Geektimes.com, 2012].....	31
Obrázek 12 – Statistika využití verzí Android [bgr.com, 2018].....	33
Obrázek 13 – Architektura Android .....	34
Obrázek 14 – Životní cyklus aktivity .....	39
Obrázek 15 – Vygenerované QR kódy .....	51
Obrázek 16 – Odkaz na stažení aplikace .....	52

## Seznam tabulek

Tabulka 1 – Úrovně korekce chyb .....	20
Tabulka 2 – Výsledky výzkumu 1 .....	21
Tabulka 3 – Výsledky výzkumů 2 .....	22
Tabulka 4 – Verze QR kódu [qrcode.com, 2018].....	23
Tabulka 5 – Verze Micro QR kódů [qrcode.com, 2018].....	24
Tabulka 6 – Porovnání 2D kódů .....	28

# 1 Úvod

Žijeme ve věku konzumismu, ve světě, kde se vyrábí a spotřebovává nepředstavitelné množství výrobků různých druhů. Pokud chcete v takovém světě přilákat zákazníka obklopoého nevkusnou reklamou, kterou se snaží obejít, musí se marketologové snažit najít různé a nestandardní způsoby.

Jedním z takových způsobů je použití tzv. QR kódů. Podstata jeho použití v marketingu spočívá v tom, že zjednodušuje proces zadání web adresy, což snadno propojuje online a offline marketing a tím i zvětšuje skupinu zájemců o produkt.

Problém je v tom, že většina marketologů je špatně seznámena s technologií dvourozměrných kódů a jejich vlastnostmi. Proto většinou umisťuje ošklivé černé a bílé kódy pod hromadu sloganů, což zákazníka spíše odpudí, než přiláká. Tento fakt, že QR kód sám o sobě má podobu, která nenesé žádný význam pro lidský vjem – někdy jindy nechá člověka nevšimnout si toho kódu. Ale tuto okolnost lze změnit a převést na výhodu obarvením samotného kódu nebo přidáním kódu grafických prvků.

Tato bakalářská práce si klade za cíl vytvořit jednoduchou aplikaci pro smartfony běžící na operačním systému Android, která bude schopna vytvořit reklamní QR kód podle zadaných textových dat a výsledný obrázek bude možné buď uložit do paměti, nebo ho rovnou odeslat na e-mailovou adresu.

V teoretické části této práce chci poskytnout přehled o dvourozměrných kódech, představit jejich druhy a vzájemně je porovnat. Bude také uvedena informace o operačním systému Android, jeho architektuře a nezbytné věci potřebné k vytvoření Android aplikace.

V praktické části bude předložen návrh mobilní aplikace a dynamický popis realizovaných komponentů.

K dosažení stanovených cílů byly použity hlavně zahraniční zdroje. Pro pochopení technologie QR byla nezbytná informace nalezená na oficiálních stránkách Denso Wave, korporaci, která vyvinula QR kód. Při vývoji aplikace byly využity oficiální stránky pro Adroid vývojáře, které poskytují podrobnou dokumentaci.

## **2 Cíl práce a metodika**

### **2.1 Cíl práce**

Cílem bakalářské práce je popsání technologií QR kódů a vytvoření aplikace, která tento kód vytvoří podle zadaných textových údajů, umožní ho doplnit o pomocnou grafiku a buď uložit do paměti, nebo odeslat na e-mailovou adresu.

### **2.2 Metodika**

Celá bakalářská práce je rozdělena do 3 částí. První část je teoretická bude v ní na základě externích zdrojů a metodou experimentu vypracovaná analýza technologie QR kódů, možností jejich kombinace s bodovou grafikou a porovnání s jinými dvourozměrnými kódy. Dále také bude popsán operační systém Android a nezbytné znalosti pro napsání aplikace pod tuto platformu.

Druhá část je praktická a bude popisovat proces vytvoření aplikace. Zaprvé bude provedena analýza nabídky oficiálního centrálního úložiště aplikace platformy Android, která se nazývá Play Market, na na existující řešení dané problematiky. Poté bude předložen vlastní návrh aplikace a popis vlastního řešení. Na konci druhé části bude provedeno testování a budou předloženy výsledky.

Třetí část tvoří závěr práce a obsahuje vlastní zhodnocení aplikace a možnosti dalšího vývoje.

## **I. Literární rešerše**

### 3 Dvourozměrné kódy

Dvourozměrné kódy byly vyvinuty s hlavním cílem umístit v sobě větší objem informací než jejich předchůdce, a to klasické čárové kódy UPC standardu. Podstatou této technologie je to, že data jsou šifrována ve dvou dimenzích (horizontálně a vertikálně). 2D kódy se dají rozdělit do dvou skupin, a to víceúrovňové (stacked) a maticové (matrix) kódy. [4]

#### 3.1 Víceúrovňové kódy

Víceúrovňové kódy přišly historicky dřív a představovaly navrstvení několika obyčejných lineárních kódů. Výstižným příkladem víceúrovňového kódu je PDF417.(4)

##### 3.1.1 PDF417

PDF417 je dvourozměrný kódem vyvinutý firmou Symbol Technology v roce 1991. V současné době je hojně využíván při prokazování totožnosti, při evidenci výrobků, na letenkách nebo vstupenkách. Tento kód je schopen zachovat až do 2720 numerických nebo do 1850 ASCII znaků.

PDF417 může obsahovat do 90 řádků. Každý řádek se skládá z:

- „Start a stop“ značek
- Sady klíčových slov:
  - Levý a pravý indikátor obsahují informace o pořadovém čísle řádků, celkovém počtu řádků a sloupců a úrovně korekce chyb.
  - Do 30 klíčových slov dat obsahují samotná data a informace pro rekonstrukce dat v případě poškození samotného kódu.

Každé klíčové slovo se skládá ze 4 čar a mezer, šířka klíčového slova je 17krát větší než nejmenší čára nebo mezera, odtud i přípona v názvu PDF417. Pro rekonstrukce dat je používané RS (Reedová Solomonová) kódování (viz kapitola 4.1) (3) (5)



Obrázek 1 - PDF417

## 3.2 Maticové kódy

Maticové kódy na rozdíl od víceúrovňových kódů uchovávají data horizontálně a vertikálně.(4)

### 3.2.1 Aztec Code

V roce 1995 doktor Andrew Longacre, Jr., výzkumník firmy Welch Allyn Inc., vyvinul dvourozměrný kód Aztec Code. Kód byl zveřejněn v roce 1997, a přestože dostal patent, je otevřen pro veřejnost. Název „Aztec“ dostal z toho důvodu, že poziční znak umístěný uprostřed kódu připomíná pyramidu z ptačí perspektivy, nacházející se v Jižní Americe.(5)

Velikost kódu může být stanovená od 15x15 do 151x151 modulu, přičemž minimálně může zachovat buď do 13 numerických znaků nebo do 12 písmen anglické abecedy a maximálně do 3832 numerických znaků nebo do 3067 písmen anglické abecedy.(3)

Existence zvláštního pozičního znaku (tzv. Bull's eye) umožňuje dekodovat i při zrcadleném obrazu kódu. Výhodou také je to, že nepotřebuje „tiché zóny“ po stranách, což umožňuje zachovat víc informací na menší ploše. Je možné také nastavit úroveň korekce chyb (viz kapitola 4.1), obecně je to 23 %, ale může být nastaven od 5 % do 95 %.



Obrázek 2 – Aztec Code

### 3.2.2 MaxiCode

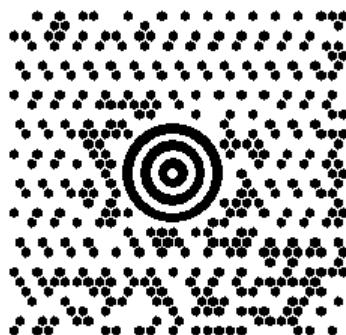
MaxiCode je dvourozměrný strojově čitelný kód zveřejněný v roce 1992 firmou United Parcel Service. Je vhodný pro sledování a řízení přepravy zásilek. Uprostřed kódu je stejně jako u Aztec kódů umístěn „Bull's eye“, ale má kulatý tvar, kolem něhož jsou uspořádány šestiúhelníkové body.

MaxiCode je schopen uložit až 93 znaků. Tento kód využívá několik režimů pro ukládání určitých typů dat obsahujících strukturované zprávy s informací o zásilce, která je také zabezpečená, RS korekce chyb (viz kapitola 4.1). Využívané módy (režimy):



- Mód 2 – data obsahující strukturovanou zprávu s poštovním směrovacím číslem;
- Mód 3 – data obsahující strukturovanou zprávu s alfanumerickým poštovním směrovacím číslem;
- Mód 4 – data s obecnou korekcí chyb;
- Mód 6 – používá se pro naprogramování zařízení.

Výhodou tohoto kódu je to, že se dá rychle a snadno načíst na rychle běžícím pásu.(6)



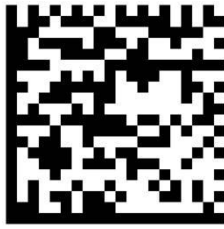
Obrázek 3 – MaxiCode

### 3.2.3 Data Matrix

Data Matrix je dvourozměrný kód vyvinutý firmou RVSI/Acuity CiMatrix. Je využíván v aerokosmickém průmyslu, pro označení dílu, v elektronickém a automobilovém průmyslu. (5)

Data Matrix může přijímat buď tvar čtverce, nebo obdélníka. Pokud se jedná o čtverec, velikost kódu může být stanovena od 10x10 do 144x144 modulu, ale kód ve tvaru obdélníka může být od 8x18 do 16x48 modulu (pouze sudý počet). Maximálně může zachovat do 3116 numerických nebo do 2335 alfanumerických znaků. Stejně jako i u ostatních dvourozměrných kódů lze nastavit úroveň korekce chyb a může být nastavená do 30 %. Místo pozičních znaků použitých v kódech standardu Aztec Code a MaxiCode pro správné určení polohy jsou v levé dolní straně souvislé čáry a v pravé horní straně střídavě umístěny černé a bílé znaky. (3)(4)(5)

Hlavní výhodou kódu standardu Data Matrix je uložení velkého objemu dat při využití menší plochy.



Obrázek 4 – Data Matrix

V současné době existují i další dvourozměrné kódy, ale v této práci se soustředím především na QR kódy.

## 4 QR Code

QR kód neboli Quick Response Code v překladu z angličtiny znamená rychlá odezva a je to dvourozměrný maticový kód vynalezený firmou „Denso-Wave“ v roce 1994 v Japonsku. Právě v Japonsku se QR kódy staly nejvyužívanějšími díky tomu, že na rozdíl od ostatních druhů dvourozměrných kódů umí zakódovat i symboly „Kanji“ a „Kana“. Název „QR Code“ je oficiálně registrovanou obchodní značkou společnosti „Denso Wave Incorporated“, ale využití kódů je zadarmo pro fyzické a právnické osoby, navíc jsou popsány a publikovány jako standardy ISO.(1)

Samotný QR kód představuje čtvercovou mřížku obsahující většinou černé body (tzv. moduly) na bílém pozadí. Na rozdíl od svého předchůdce, klasického EAN kódu, který je skenován tenkým laserovým paprskem, QR kód je detekován snímačem nebo kamerou jako 2D obrázek. Tři největší čtverce umístěné v úhlech kódu a synchronizační pásy umožňují normalizovat velikost obrázku, jeho orientaci a úhel mezi detekujícím zařízením a kódem. Na obrázku číslo 5 je dobře vidět jednotlivé části QR kódů a k čemu slouží.



Obrázek 5 – Schéma QR kódů

### 4.1 Úrovně korekce chyb

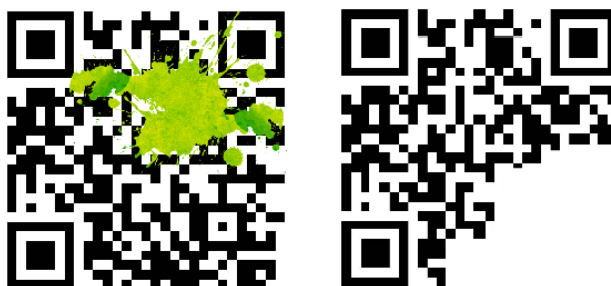
Další výhodou QR kódů je jejich schopnost uložit data tak, aby až při 30% poškození kódu byly stále přístupné pro čtení. Tato vlastnost je realizována pomocí metody RS kódů (Reedovy-Solomonovy kódy), což jsou nebinární cyklické kódy, umožňující opravovat chyby v bloku dat. Elementy kódového vektoru nejsou bity, ale skupiny bitů (bloky). Jsou využívány i pro rekonstrukce dat na CD a DVD discích. (2)(3)

Existují 4 úrovně korekce chyb, které definují, kolik procent kódu může být poškozeno. V tabulce číslo 1 jsou uvedené úrovně s jejich binární interpretací při šifrování a procentní vyjádření, jak velká část kódu může být poškozena, aby kód přesto zůstal čitelný pro zařízení.

Název	Binární zápis	Kolik procent kódů může být poškozeno
L	01	7 %
M	00	15 %
Q	11	25 %
H	10	30 %

Tabulka 1 – Úrovně korekce chyb

Volba určité úrovně záleží na stupni rizika, že v průběhu využití kódu bude nějakým způsobem poškozen. Například pokud je vytvořený kód určen pro umístění někde na ulici třeba jako reklamní poutače, chtělo by to použít vyšší úroveň korekce chyb. Využití nižších úrovní je lepší při potřebě kódování rozsáhlejšího objemu dat a umístění kódu někde, kde je minimální riziko poškození. Na obrázku číslo 6 je příklad využití vyššího stupně korekce chyb (Level H) pro poškozený QR kód, obsahující odkaz na webové stránky provozně ekonomické fakulty (<https://www.pef.czu.cz>).



Obrázek 6 – Poškozené QR kódy

Protože se různé typy dat ukládají do určitých částí QR kódu a jsou opakovatelné, například data o verzi kódů se nachází v levé horní straně QR kódu (obrázek číslo 5), samotný kód může být poškozen o větší procento po stranách než například uprostřed. Byl proveden výzkum, aby se zjistilo, jak velká plocha QR kódu může být poškozena, aby si kód ponechal svou čitelnost v závislosti na zvolené úrovni korekce při vytvoření kódu. Byly vytvořeny 4 kódy s různou úrovní korekce chyb, ale se stejným zakódovaným textovým řetězcem „HELLO WORLD“. Dále každý QR kód byl úmyslně poškozen

umístěním různě velkých čtverců červené barvy v jeho různých částech. Každý QR kód byl vytištěn na papír a skenován pomocí smartfonu vybaveného 12megapixelovou kamerou. Následující tabulka obsahuje výsledky získané v průběhu výzkumu. Ukazuje v kódech, s jakou úrovní korekce jak velká plocha by se dala zneužít (v modulech) a v jaké části kódu (v kulatých závorkách uvedena velikost samotného kódu v modulech).

	L (21x21)	M (21x21)	Q (21x21)	H (25x25)
uprostřed	5x5	6x5	7x7	10x10
po levé straně	9x7	9x7	11x7	13x11
na horní straně	7x9	7x9	7x11	11x13
dolní pravý uhel	5x5	6x6	6x6	9x9

Tabulka 2 – Výsledky výzkumu 1

Další výzkum měl za cíl ověřit vliv okolností při snaze načíst poškozený kód. Výzkum spočívá v tom, že se snaží načíst kódy použité v předchozím výzkumu, nacházející se v různých polohách vůči čtecímu zařízení a přírůzné úrovni osvětlení místnosti. Byly zkoumané vlivy následujících okolností:

- Načtení kódu z dálky (>0.5 metru);
- Načtení při denním osvětlení;
- Načtení při umělém osvětlení;
- Kombinace výše uvedených okolností s různými polohami kódu vůči čtecímu zařízení.

Pro každý kód s určitou úrovní samoopravy bylo provedeno 10 pokusů načtení. Následující tabulka číslo 3 zobrazuje výsledky (vztah počtu úspěšných pokusů k celkovému počtu)

Úroveň korekce	Počet úspěšných pokusů
L (7%)	7/10
M (15%)	9/10
Q (25%)	10/10
H (30%)	10/10

Tabulka 3 – Výsledky výzkumů 2

## 4.2 Velikost QR kódu

Velikost QR kódu záleží na zvolené verzi QR a objemu šifrované informace. Při tvorbě QR velikost výsledného kódu hraje velkou roli, protože menší QR kódy mohou být nečitelné pro některé typy zařízení majících kameru s malým rozlišením bodů.

Verze kódů jsou rozdělené podle počtu modulů. Originální QR kód byl schopen zakódovat až 1167 numerických znaků při maximální verzi – 14 (což představuje 73x73 modulu). Poté při vylepšení originální verze QR kódu byl vynalezen druhý model, který v současné době dosáhl až 40 verzí (177x177 modulu) a umísťuje až 7089 numerických nebo až 4464 alfanumerických znaků.(1)



Obrázek 7 – Modely QR kódů [qrcode.com, 2018]

Další tabulka číslo 4 obsahuje popis některých verzí QR kódů při určitých úrovních korekce chyb.

Verze	Počet modulů	Úroveň korekce chyb	Počet numerických znaků	Počet alfanumerických znaků
1	21x21	L	41	25
		M	34	20
		Q	27	16
		H	17	10
10	57x57	L	652	395
		M	513	311
		Q	364	221
		H	288	174
20	97x97	L	2061	1249
		M	1600	970
		Q	1159	702
		H	919	557
30	137x137	L	4158	2520
		M	3289	1994
		Q	2358	1429
		H	1782	1080
40	177x177	L	7089	4296
		M	5596	3391
		Q	3993	2420
		H	3057	1852

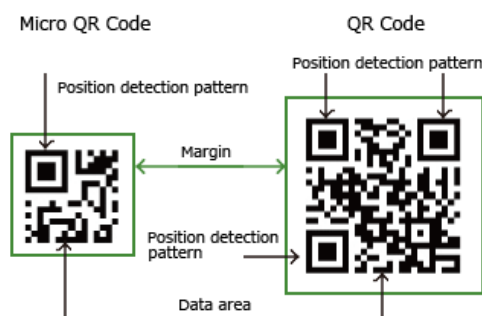
Tabulka 4 – Verze QR kódu [qrcode.com, 2018].

### 4.3 Druhy QR kódu

Při snaze zlepšit původní QR kód bylo navíc vyvinuto druhu dvourozměrných kódů, a to Micro QR, iQR a SQRC.

#### 4.3.1 Micro QR

Nejdůležitější vlastností Micro QR je to, že má pouze jeden poziční znak na rozdíl od původního QR kódu. Rozdíl představuje také to, že pro „tichou zónu“ kolem QR kódu je potřeba využít 4 moduly, zatímco pro Micro QR pouze 2, což teoreticky umožňuje zachovat větší objem informací při stejné velikosti.(1)(7)



Obrázek 8 – Rozdíl Micro QR Code a QR Code [qrcode.com, 2018]

Micro QR má pouze 4 verze a maximálně může uchovat až 35 numerických nebo až 21 alfanumerických znaků. V tabulce číslo 5 jsou popsány jednotlivé verze Micro QR.

Verze	Počet modulů	Úroveň korekce chyb	Počet numerických znaků	Počet alfanumerických znaků
M1	11x11	-	5	-
M2	13x13	L	10	6
		M	8	5
M3	15x15	L	23	14
		M	18	11
M4	17x17	L	35	21
		M	30	18
		Q	21	13

Tabulka 5 – Verze Micro QR kódů [qrcode.com, 2018]

#### 4.3.2 iQR

iQR Code dokáže zachovat o 80 % více informací než QR Code při stejné velikosti. Může přijímat jak tvar čtverců, tak i obdélníků, což umožňuje umisťovat kód i na cylindrický povrch a přitom zůstat čitelný. Důležitou vlastností je to, že může mít další úroveň korekce chyb – Level S, díky kterému kód zůstává pořád čitelný i při 50% poškození.

Velikost iQR ve čtvercovém tvaru může být stanovena od 9x9 modulů do 422x422 a může být uchováno až 40637 znaků, v obdélníkovém tvaru od 5x19 do 43x131 modulů a maximálně může být zašifrováno až 1202 znaků. (1)(7)





Obrázek 9 – iQR Code [qrcode.com, 2018]

### 4.3.3 SQRC

Jde o druh QR kódu, který je vybaven funkcí restrikce čtení, což umožňuje uchovat privátní data. Tento kód může být načten pouze určitým druhem zařízení, takže může být využit v rámci firmy.(1)

## 4.4 Generování QR kódů

Tvorba QR kódu se provádí pomocí generátoru neboli softwaru, který z údajů zadaných uživatelem vygeneruje QR kód. V dnešní době existuje velké množství online generátorů a mobilních aplikací umožňujících přečíst a vytvořit QR kód.

Generátor umožňuje zakódovat obecný text, kontakt (vCard [viz kapitola 4.6.1]), událost, udání zeměpisných souřadnic, odkaz na webovou stránku nebo na stažení aplikace pro smartphone například v Google Play Marketu, a také údaje pro připojení k bezdrátové síti.

Některé generátory můžou nabídnout za poplatek vytvořit dynamický QR kód. Jeho podstatou je to, že po vytvoření informace, na kterou kód odkazuje, může být upravena bez zásahu do samotného kódu. Po načtení dynamického kódu přesměruje uživatele na server, který v závislosti na uložených datech přesměruje uživatele na konečnou stránku. Dynamické kódy jsou vhodné proto, že jsou vytvořené pouze jednou a dál není potřeba je měnit.

Existuje možnost sledování QR kódu, kde a kdy byl načten, což je vhodné pro marketing. Umožňuje přidání takové funkce ke skutečnému odkazu UTM parametrů. UTM parametry jsou malé části kódu přidávané ke skutečnému URL oddělenému znakem „?“, každý parametr se skládá ze dvou částí, názvu parametrů a hodnoty. Při procházení takového odkazu se informace zachycuje analytickým nástrojem, v našem případě pro zjištění, že se někdo dostal na naše stránky, prostřednictvím QR kódu.

## 4.5 Čtení QR kódů

Pro nalezení dat zakódovaných v QR kódu se obvykle používá software, který pomocí kamery rozpozná kód, dostane vstup na zakódované řetězce, dekóduje a převede na výstup obsah v podobě srozumitelné uživateli. Existence kamery není nutná, některé aplikace umožňují nahrát obrázkový soubor a dekódovat ho z obrázku.

Většinou je pro načtení QR kódu využíván smartfone, existuje ale obrovské množství aplikací umožňujících přečíst kromě QR kódu také i jiné druhy dvourozměrných kódů. Ale pro načtení dvourozměrných kódů při inventarizaci zboží ve skladech nebo například u turniketu na letišti existují jiná zařízení, a to 2D čtečky, jejichž funkčnost je založena na fotosenzoru. Princip činnosti je založen na použití image senzoru, který vysokou rychlostí (obvykle do 60 záběrů za sekundu) a se skvělým rozlišením skenuje určitou oblast.

## 4.6 Obsah QR kódů

Do dvourozměrných kódů lze zakódovat pouze text, ale ten může reprezentovat mnoho různých věcí. Často lidé do 2D kódů kódují text, který reprezentuje URL adresu, například <http://google.com/m>. Jedná se o speciální řetězec textu, který je rozpoznáván čtečkou jako URL. Dvourozměrné kódy mohou zakódovat mnoho typů textů, které pak lze spustit. Text, který reprezentuje kontaktní informace, může po načtení čtečkou vyvolat výzvu k přidání kontaktu do adresáře. Ale toto funguje pouze tehdy, když čtečka pochopí, že nalezený text reprezentuje kontaktní informace. Proto existují speciální standardy. (14)

### 4.6.1 vCard

vCard je souborový formát pro výměnu elektronických vizitek. Soubor vCard se skládá z položek vCard druhu, z nichž každá obsahuje informace o jedné vizitce. Položka vCard může obsahovat jméno, adresu, telefonní čísla, URL, logo atd. vCard soubory mají formát vcf, ve kterém se ukládají i kontakty ve smartfonech. (14)

Formát vCard, který popsán v RFC 2426<sup>1</sup> je rozšířením MIME<sup>2</sup> formátu. Podle RFC, MIME představují text sestavený z řádků v podobě <atribut>:<hodnota>. (16)

Tak například náhodná osoba jménem Bob má následující kontaktní údaje

---

<sup>1</sup> RFC- Je v informatice označení řady dokumentů popisujících internetové protokoly, systémy apod.

<sup>2</sup> MIME – Standard popisující přenos různých druhů dat prostřednictvím elektronické pošty.

- telefonní číslo (pracovní): +420777777777
- firma, ve které pracuje: Abrakadabra s.r.o.
- web adresa: abrakadabrasro.cz,

bude mít následující vizitku:

```
BEGIN:VCARD
VERSION:3.0
N:Bob
TEL;WORK:+420777777777
ORG:Abrakadabra s.r.o.
URL:abrakadabrasro.cz
END:VCARD
```

Příklad 1

#### 4.6.2 Configurace Wi-Fi

Stejným způsobem lze specifikovat wi-fi konfigurace. Syntax je velice podobná jako v příkladu s vizitkou, jenže všechny údaje jsou zapisované do jednoho řádku. Skenování takového kódu by konfigurovalo wi-fi připojení odpovídajícím způsobem. Pro úspěšnou konfiguraci by měl kódovaný řetězec obsahovat správné definované atributy: název sítě (S), typ šifrování (T) a heslo (P). (14)

Příklad kódovaného řetězce pro připojení k síti „home“, která má heslo pro připojení „hello“ a typ šifrování WPA:

```
WI-FI:S:home;T:WPA;P:hello;;
```

Příklad 2

#### 4.6.3 Ostatní možnosti

Zakódovaná informace může také reprezentovat jenom telefonní číslo, po načtení takového kódu se systém zeptá uživatele, jestli chce zavolat na toto číslo. Toho lze dosáhnout, když před kódovaným telefonním číslem bude přidán řetězec „tel:“.

Navíc jde zakódovat SMS zprávu. Po načtení kódu by se systém zeptal uživatele, zda chce zaslat nalezenou zprávu na nalezené telefonní číslo. Kódovaný řetězec by přitom měl mít následující podobu „smsto: [telefonní číslo]:[text zprávy]“.

Pokud naskenovaný 2D kód bude obsahovat nějakou URL adresu, systém ji rozpozná a otevře v prohlížeči.

## 4.7 Porovnání 2D kódů

Následující tabulka porovná dvourozměrné kódy představené v této práci dle určitých vlastností a možností užití, které nabízejí.

Kritérium	Aztec Code	MaxiCode	Data Matrix	QR Code
Autonomní načtení	Ano	Ano	Ano	Ano
Optimalizace pro existující technologii tisku	Obrázek se sestavuje ze standardních čtvercových pixelů	Obrázek se sestavuje se z čar, centrální poziční znak je kulatý, tisk s menším rozlišením je obtížný.	Obrázek se sestavuje ze standardních čtvercových pixelů	Obrázek se sestavuje ze standardních čtvercových pixelů
Maximální objem dat	Do 3832 numerických nebo 3067 alfanumerických znaků (2 kB)	Do 138 numerických nebo 93 alfanumerických znaků (90 Bajt)	Do 7089 numerických nebo 4296 alfanumerických znaků (3kB)	Do 3116 numerických nebo 2335 alfanumerických znaků (1,5kB)
Minimální velikost (modulů)	15x15	33x30(29)	10x10	21x21
Maximální velikost (modulů)	151x151	-	144x144	177x177
Úrovně korekce chyb	Od 5 % do 95 %	Do 12,5 %	Do 30 %	Od 7 do 30 %
Stabilita prostorového rozpoznávání	Může být otočen o libovolný počet stupňů, čitelný i při zrcadleném odrazu.	Může být otočen o libovolný počet stupňů, čitelný i při zrcadleném odrazu.	Může být otočen o libovolný počet stupňů, čitelný i při zrcadleném odrazu.	Může být otočen o libovolný počet stupňů.
Oblasti využití	Používá se na online lístcích aero a železničních společností.	Vnitropodnikové použití firmou UPS.	Velmi široká oblast využití.	Velmi široká oblast využití, ale často využíván v reklamě.

Tabulka 6 – Porovnání 2D kódů

## 4.8 QR Code a mobilní marketing

Velice rychlý vývoj mobilních technologií způsobil růst mobilního trhu, v dnešní době skoro každý člověk má smartphone, který je svým funkcionálem schopen nahradit osobní počítač, jenž je neodmyslitelným atributem života pro většinu lidí. V souladu se statistickými údaji představuje dnes počet uživatelů smartphonů počet kolem 3,5 miliardy.

Takové okolnosti způsobily vývoj jednotlivých oblastí marketingu, např. mobilní marketing. Mobilní marketing je soubor marketingových nástrojů, které umožňují organizacím působit na cílovou skupinu interakčním způsobem pomocí mobilního zařízení nebo mobilní sítě.

QR Code je jedním z důležitých nástrojů mobilního marketingu spojujícího online a offline marketing. Místo toho, aby člověk po písmenech zadával odkaz na stránky prodejce, stačí jenom jednoduše naskenovat QR Code. V dnešní době jsou QR kódy neodmyslitelnou součástí skoro každé reklamní kampaně. (10)

Jelikož QR kódy zajišťují spojení reálného světa se světem internetu, jsou schopné vzbudit zájem uživatelů smartfonů v jakékoliv situaci. Začíná od jednoduché informace (kontaktní údaje, datum nějaké události a popis produktu) přes speciální nabídky (losování a slevové kupóny) až po možnost předložit téměř všechno, co může zobrazit standardní prohlížeč.

Tím, že potenciální klienti naskenují QR kód umístěný v inzerátu či jiné reklamě, projevují o ně zájem. Úmyslně se budou snažit dozvědět se o tom víc. Z toho vyplývá, že jsou vtaženi do procesu získávání informací a také pozorněji vnímají. Tuto okolnost nelze podceňovat z toho důvodu, že spotřebitel získává za den průměrně kolem stovky různých reklamních zpráv.

Pomocí QR kódů lze stanovit přímý kontakt s klientem, například pro získávání zpětné vazby na výrobek, přičemž s odvoláním dostanete elektronickou adresu nebo jinou kontaktní informaci, která může být využita v budoucnosti pro další interakci s klientem.(9)

Ten fakt, že QR kód sám o sobě má podobu, která nenesé žádný význam pro lidský vjem – někdy jindy nechá člověka nevšimnout si toho kódu. Ale tuto okolnost lze napravit a převést na výhodu obarvením samotného kódu nebo přidáním kódu grafických prvků.

## 4.9 QR Code a grafika

QR Code je z hlediska marketingu velice užitečný nástroj, i když je často kritizován marketology za to, že má nevábný vzhled kvůli použitým černobílým barvám, což je velkou nevýhodou.

Ale není to pravda, protože QR kód může být zpracován různými způsoby a zanechá si při tom svou čitelnost. QR Code má neomezený potenciál z hlediska designů. Například nemusí být vytvořen pouze s použitím černé a bílé barvy, je potřeba, aby byly využity barvy s dostatečným kontrastem, aby čtečka mohla kód rozpoznat. Taky je to může být například vytvořen s použitím gradienta. Podstatou je zanechat dostatečný kontrast mezi kódem a pozadím, ale zpravidla je lepší využít tmavý odstín na světlém pozadí.

Další možností je vytvoření modulů s kulatými rohy, což estetizuje QR kód. Hezký způsob ke zvýšení působení na cílovou skupinu je umístění obrázků uprostřed kódů, což je umožněné díky korekci chyb kódů. Může to být cokoliv, například logo firmy nebo tematický obrázek. Velikost obrázku je definovaná úrovní korekce chyb QR kódu, proto je lepší využívat nejvyšší úroveň, protože umožňuje umístit obrázek velikosti 30 % od celkové plochy QR kódu. Další QR kódy jsou výtvarným dílem odborných designerů.(11)



Obrázek 10 – QR Art[Geektimes.com, 2012]



Obrázek 11 – JESS3 [Geektimes.com, 2012]

## 5 OS Android

Android OS – operační systém vytvořený na základě otevřené platformy Linux. Každý si ho může zdarma stáhnout, změnit a zveřejnit vlastní verze. Dokonce i konkurenti systému Android mohou používat tento operační systém pro své vlastní účely. Android OS je používán v různých typech zařízení jako tablet, smartfon, elektronická kniha, smart watch, televize aj.

Původně operační systém byl vyvinut firmou Android Inc., kterou v roce 2005 koupila společnost Google. Následně zahájila tvoření aliance Open Handset Alliance (OHA) zabývající se nyní podporou a dalším rozvojem platformy.

Dneska je Android nejpopulárnější operační systém na světě. Dle statistických údajů Googlu byly v roce 2017 aktivovány více jak 2 miliardy zařízení běžících na OS Android a jejich počet stále roste každý den. (15)(12)

### 5.1 Přehled verzí Android

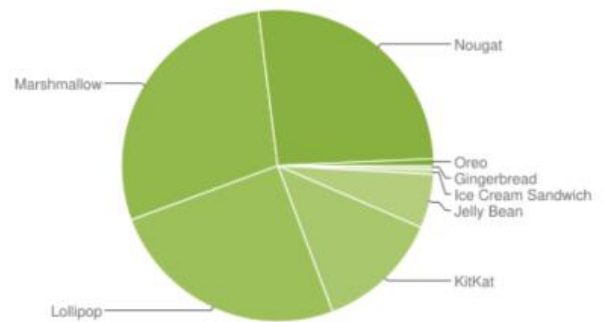
Po celou dobu existence operačního systému byl vypuštěn relativně velký počet jeho verzí, které mají názvy podle různých sladkostí. Tady je přehled některých z nich:

- **Android 2.3 Gingerbread (API level 9)** – Verze byla publikovaná v roce 2010, dle statistik byly smartfony s Gingerbread populární do 2013 roku.
- **Android 4.0 Ice Cream Sandwich (API level 14)** – V této verzi se objevil přístup k aplikacím přímo z lock screenů, objevila se i funkce odemčení obrazovky gestem a funkce rozpoznání obličeje.
- **Android 4.1 Jelly Bean (API level 16)** – bylo provedeno přejmenování Android Market na Google Play. Byl přidán Servis Google Now, který umožňoval poslouchat hudbu, psát zprávy a používat navigaci současně. Servis odhadoval chování a zvyky svého uživatele.
- **Android 4.4 Kit Kat (API level 19)** – úplné obnovení uživatelského rozhraní, podpora krokoměřících zařízení, podpora aplikací s infračerveným portem.
- **Android 5.0 Lollipop (API level 21)** – nahrazení Dalvik Virtual Machine na Android Runtime.
- **Android 6.0 Marshmallow (API level 23)** – možnost odemčení obrazovky otiskem prstu.



- **Android 7.0 Nougat (API level 24)** – nový design, možnost přepínání mezi aplikacemi jediným dotykem. Byl vytvořen režim rozdělené obrazovky na dvě samostatná okna.
- **Android 8.0 Oreo (API level 26)** Omezení procesu běžícího na pozadí pro zvětšení výkonu procesoru.(13)

Version	Codename	API	Distribution
2.3.3 - 2.3.7	Gingerbread	10	0.4%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	0.5%
4.1.x	Jelly Bean	16	1.9%
4.2.x		17	2.9%
4.3		18	0.8%
4.4	KitKat	19	12.8%
5.0	Lollipop	21	5.7%
5.1		22	19.4%
6.0	Marshmallow	23	28.6%
7.0	Nougat	24	21.1%
7.1		25	5.2%
8.0	Oreo	26	0.5%
8.1		27	0.2%

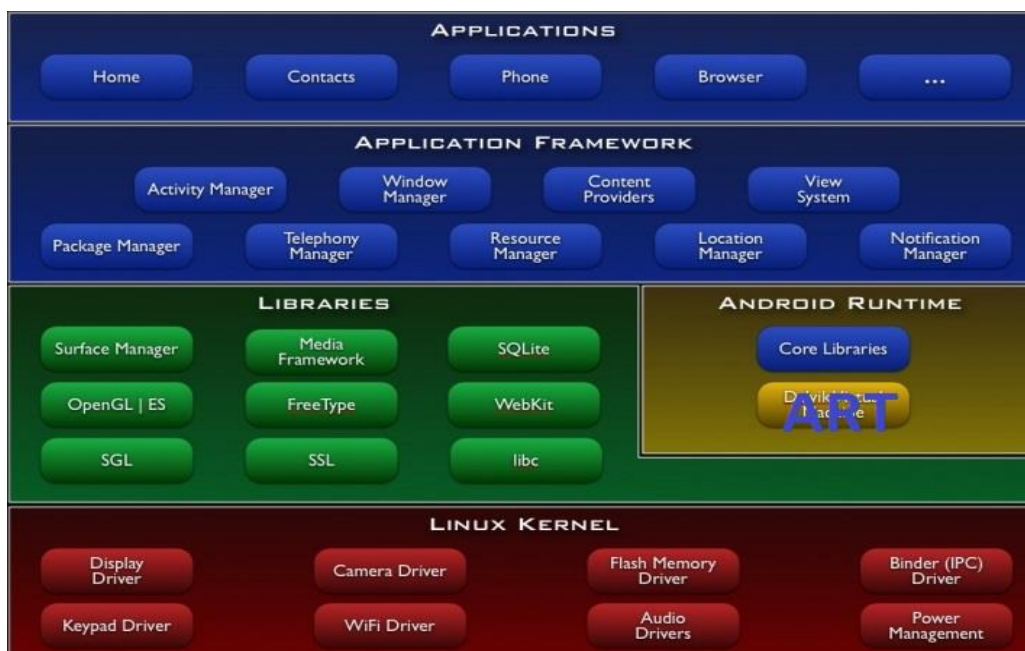


Data collected during a 7-day period ending on January 8, 2018.  
Any versions with less than 0.1% distribution are not shown.

**Obrázek 12 – Statistika využití verzí Android [bgr.com, 2018]**

## 5.2 Architektura

Architektura OS Android je tvořena 5 vrstvami, které je vidět na obrázku číslo 13.



Obrázek 13 – Architektura Android

### 5.2.1 Linux Kernel

Linux Kernel (Úroveň jádra) – je základní vrstva operačního systému. Platforma založená na OS Linux verzi 2.6, má přístup k systémovým službám jádra jako řízení paměti a procesy, poskytnutí bezpečnosti, práce se sítí a drivery. Jádro také slouží jako abstrakční vrstva mezi hardwarem a softwarem.

### 5.2.2 Libraries

Libraries (Úroveň nativních knihoven) je další vrstvou operačního systému, která obsahuje sadu C/C++ knihoven používaných různými komponentami OS. Pro vývojáře je přístup k těmto knihovnám umožněn pomocí vrstvy Application Framework.

### 5.2.3 Android runtime

Android runtime (Úroveň běhového prostředí) je součástí vrstvy nativních knihoven, je běhovým prostředím Android aplikace, vyvinutá firmou Google jako nahrazení předchozí verze virtuálního stroje Dalvik Virtual Machine. Slouží pro zpracování tzv. Java bytecode, který je kompilovaný ze zdrojových kódů. Díky tomu, že Android runtime kompiluje aplikaci současně s jejím instalováním, zrychluje se práce systému a šetří energii.

#### **5.2.4 Application Framework**

Application Framework (Úroveň aplikačního rámce) je infrastrukturou systému. Obsahuje různé API (Application Programming Interface – je sbírkou tříd, procedur a funkcí), které je možné využít při napsání vlastní aplikace.

#### **5.2.5 Applications**

Applications – je vrstvou operačního systému, do níž patří sada základních aplikací jako SMS nebo e-mail klient, kalendář, mapy, prohlížeč, kontakty a jiné.(12)

### **5.3 Android z pohledu aplikace**

Každá aplikace provádí svou činnost v izolovaném prostředí. Operační systém Android je představen jako víceuživatelský systém Linux, ve kterém je každá aplikace „samostatným uživatelem“. Ve výchozím nastavení systém přiřazuje každé aplikaci jedinečné ID uživatele Linuxu, systém nastavuje oprávnění pro všechny soubory v aplikaci tak, aby k nim mohli přistupovat pouze uživatelé s identifikátorem přiřazeným k dané aplikaci. Každý proces má svůj vlastní virtuální stroj, takže kód aplikace běží odděleně od jiných aplikací. Ve výchozím nastavení běží každá aplikace ve svém vlastním procesu Linuxu. Android spustí proces, kdy je potřeba spustit libovolnou součást aplikace, a poté proces ukončí, když již není potřeba, nebo když je potřeba uvolnit místo v paměti pro jiné aplikace.

Takovým způsobem Android implementuje zásadu udělení minimálních práv. To znamená, že každá aplikace ve výchozím nastavení má přístup pouze k těm komponentám, které potřebuje pro práci, nikoli jiným. To vytváří výjimečně zabezpečené prostředí, ve kterém aplikace nemá přístup k neoprávněným oblastem systému.

Nicméně aplikace může poskytnout své data jiným aplikacím a přístupu k systémovým službám. Ve dvou aplikacích může být přiřazeno jedno ID uživatele Linux. V tomto případě může každý z nich přistupovat k souborům jiné aplikace. Aplikace může požádat o přístup k datům zařízení, například uživatelským kontaktům, SMS zprávám, k SD kartě, fotoaparátu, Bluetooth atd.

## 5.4 Vývojové prostředí

Java – nejpoužívanější programovací jazyk pro vytvoření Android aplikací. Zařízení běží na operačním systému Android, nespouštějí soubory typu „.class” a “.jar”. Android – zařízení používá vlastní optimalizované formáty kompilačního kódu, což znamená, že není možné použít obecné prostředí pro vývoj aplikace v jazyku Java. Je potřeba nainstalovat speciální nástroje pro převedení kompilovaného kódu ve formát používaný v Androidu.

Důležitým krokem při tvorbě aplikace je zvolení určitého vývojového prostředí (IDE). Pro vývoj aplikací běžících na platformě Android existují převážně tři alternativní verze IDE: „Eclipse IDE”, „NetBeans IDE” a „AndroidStudio”.

„Eclipse IDE” s nainstalovaným pluginem „Android Developer Tools” byl dříve nejčastější variantou pro vývoj aplikace, ale v roce 2015 Google přestal podporovat tento plugin, a proto dneska volba „Eclipse IDE” není aktuální.

„NetBeans IDE” umožňuje vývoj aplikací pomocí pluginu „NBAndroid”, ale tento plugin není oficiální a nikdy nebyl podporován společností Google, což představuje také nevýhodu.

Založené na „IntelliJ IDEA” (IDE pro vývoj Java aplikací) - „Android Studio” je nejčastější výběr pro vývoj aplikace pod platformu Android. Toto IDE je oficiálním produktem Google, a proto bylo zvoleno při tvorbě aplikace v praktické části této práce.

Dále je potřeba nainstalovat Android Software Development Kit (SDK), který obsahuje knihovny a nástroje nezbytné pro vývoj Android-aplikace:

- SDK Tools – nástroje pro ladění a testování stejně jako další užitečné programy. Také obsahuje sadu platformě závislých nástrojů.
- Dokumentace – poskytuje autonomní přístup k nejnovější dokumentaci API.
- Pomocné nástroje pro Android – doplňkové API nepřítomné ve standardní formě.
- Příklady aplikací – příklady pro pochopení toho, jak používat API.

## 5.5 Vývoj aplikace

### 5.5.1 Základní komponenty aplikace

Android-aplikace se skládá z následujících částí:

- **Activity** – Je nejdůležitějším komponentem aplikace, který reprezentuje obrazovku a poskytuje interakce s uživatelem. Aplikace se může skládat z několika takových Activity.
- **Views** – Jsou uživatelským rozhraním activity. Patří sem různé prvky umožňující ovládání aplikací. To jsou prvky jako: EditText (pole pro zadání textu), TextView (zobrazení určitého textu), RadioButton (přepínač), Spinner (tzv. číselník), ImageView (vyhrazené místo pro umístění obrázku) atd.
- **Services** – Je komponent, který může provádět delší procesy na pozadí a neobsahuje uživatelské rozhraní.
- **Content Provider** – Je komponent, který řídí přístup ke strukturované datové sadě. Pomocí tohoto komponentu si může aplikace vyměňovat data s jinými aplikacemi.
- **Intents** – Jsou asynchronní zprávy, pomocí kterých může aplikace zažádat o funkci z jiných služeb nebo activity (např. zažádat o kontakt z aplikace Kontakty).
- **Broadcast Receiver** – Je „přijímač“, který přijímá systémové zprávy, může být použit za účelem reakce na změnu stavů systému.

### 5.5.2 Bezpečnost a oprávnění

Android definuje určitá povolení pro určité úkoly, které musí plnit aplikace pomocí Android Runtime Permission. Do verze Android 6.0 byla všechna požadovaná oprávnění při instalaci aplikace zobrazené uživateli. Pokud uživatel neudělí alespoň jedno oprávnění ze seznamu, aplikace nebude nainstalována. Potom by tyto okolnosti byly změněny. Všechna oprávnění je možné rozdělit do dvou skupin – obecné (normal) a nebezpečné (dangerous).[13]

Obecná oprávnění – jsou oprávnění, která neohrožují soukromí uživatelé (např. oprávnění k internetovému přístupu).

Nebezpečná oprávnění – jsou opakem obecných a do této skupiny patří oprávnění jako přístup ke kameře, mikrofonu, ke galerii a jiné. Tímto způsobem budou při stažení a instalaci přidělena aplikaci práva ze seznamu obecných oprávnění, ale pokud se pro plnění činností aplikace vyžadují práva ze seznamu nebezpečných oprávnění, zeptá se na

to uživatele, který buď tato práva udělí, nebo odmítne. Pokud uživatel neudělí aplikaci potřebná oprávnění, aplikace i přesto bude nainstalována, ale její činnost bude omezena. Pokud se potom uživatel rozhodne udělit dříve požadované oprávnění, může to udělat přes nastavení.

### **5.5.3 Android Manifest.xml**

Android Manifest – je XML soubor, který obsahuje nezbytnou informaci o aplikaci. Pouze když systém Android obdrží tuto informaci, bude schopen vykonávat program. Musí také obsahovat seznam oprávnění, které aplikace bude vyžadovat pro plnění své činnosti.

### **5.5.4 R.java**

R.java je generovaná třída, která obsahuje odkazy na zdroje ze složky „res”. Tyto zdroje se nachází v adresáři „res” a mohou být reprezentovaný jako hodnoty, obrázky, ikonky nebo XMLsoubory. Při přidání nových zdrojů budou příslušné odkazy generované v „R.java”. Odkazy jsou statickými hodnoty typu int. Android poskytuje určité metody přístupu k určitým zdrojům. Katalog „res” obsahuje strukturované hodnoty, které jsou interpretovatelné platformou Android, ale existuje i katalog „assets”, který může být použit i pro ostatní druhy dat. Přístup k tomuto katalogu je umožněn pomocí AssetsManager.

### **5.5.5 Layout**

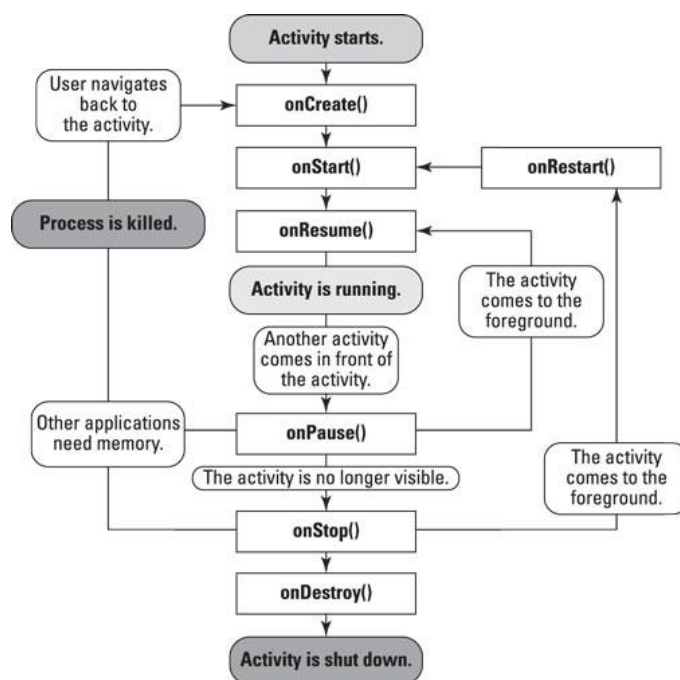
Uživatelské rozhraní (UI) je definováno pomocí makety. Maketa definuje jednotlivé prvky UI, jejich vlastnosti a umístění. Maketa může být definována pomocí Java-kódů nebo XML. Obvykle je Java-kód využíván v případech, kdy není předem stanoven obsah makety, například pokud je maketa závislá na obsahu, který je načítán z webu. Definování makety přes XML je preferovanější, protože odděluje programovou logiku od definování makety a umožňuje definovat různé zdroje pro různé typy zařízení. Dá se i kombinovat XML s Java-kódem.

### **5.5.6 Životní cyklus aktivity**

Operační systém řídí životní cyklus aplikace. Kvůli příchodnému volání může například zastavit nebo zrušit proces. Android definuje životní cyklus aktivity pomocí metod, z nichž nejdůležitější jsou:

- **onCreate()** – metoda, která je volaná, když jsou aktivity poprvé vytvořeny.
- **onPause()** – metoda, která je volaná vždy při ukončení aktivity nebo když uživatel například opustí aplikaci, ale neukončí ji.
- **onResume()** – metoda, která je volaná, když aplikace přechází do stavu onPause nebo když se uživatel znovu pustí do této aktivity.
- **onStop()** – tato metoda úplně zastaví aktivitu, volaná je například za tím účelem, když aktivita běží na pozadí a zbytečně zabírá místo v paměti.

Na obrázku číslo 14 je schéma popisující životní cyklus aktivity.



Obrázek 14 – Životní cyklus aktivity

## **II. Projekt**



## 6 Zadání

Cílem vlastní práce je vytvořit funkční prototyp aplikace, která bude schopna vytvořit QR kód podle zadaných textových dat a zvolené úrovně korekce chyb. Umožní uživateli zakódovat jak samotný text, tak i vizitku, SMS zprávu, telefon, wi-fi připojení nebo odkaz na webovou stránku. Dále aplikace umožní nastavit barvu pozadí nebo barvu samotného kódu. Také dovolí použít pro kód buď gradient, nebo uložení obrázku vybraného uživatelem z vlastní galerie. Konečný QR kód bude možné uložit do paměti zařízení a odeslat na e-mail. V závěru práce je publikace aplikace v Play Marketu.

Před zahájením práce byla provedena analýza Play Marketu na existenci podobných řešení výše uvedeného zadání. Bylo nalezeno neomezené množství generátorů černobílých kódů a několik řešení, která sice odpovídají stanovenému cíli, ale s velice omezeným funkcionaléem. Nalezené aplikace konkrétně umožňují zakódovat pouze text a doplnění o pomocnou grafiku je omezeno pouze nastavením barvy pozadí a barvy samotného kódu.

### 6.1 Návrh struktury aplikace

Aplikace bude mít 9 aktivit a jednu třídu, která bude reprezentovat samotný QR kód. Bodem vstupu bude MainActivity, dalších 6 aktivit umožní zadat textová data v závislosti na tom, co uživatel bude chtít zakódovat, další aktivita vygeneruje černobílý QR kód s úrovní korekce M a umožní uživateli změnit úroveň korekce, poslední aktivita umožní vygenerovaný QR kód doplnit o pomocnou grafiku: zadat jiné barvy pro kód nebo pro pozadí, zadat gradient pro kód nebo upraví QR kód podle zvoleného z galerie. V této aktivitě bude také realizovaná metoda pro uložení QR kódu jako obrázku ve formátu PNG a metoda, jež umožní obrázek rovnou odeslat pomocí e-mailu klienta.

#### 6.1.1 Aktivity

- **MainActivity** – vstupní bod aplikace, umožní uživateli vybrat náplň budoucího QR kódu.
- **VizitkaActivity** – aktivita bude určena pro zadání kontaktních údajů uživatelem, umožní uživateli buď zadat data ručně, nebo vybrat z kontaktu již existující kontakt pomocí ContactProvider. Odpovídá za tvorbu určitého textového řetězce, který pak bude zakódován. QR kód vytvořený na základě dat zadaných v této aktivitě bude reprezentovat vCard. Po načtení takového kódu smartfon automaticky dosadí

nalezená data do určitých polí při tvorbě nového kontaktu a nabídne uživateli uložit tento kontakt.

- **TelefonActivity** – aktivita bude určena pro zadání telefonního čísla. Odpovídá za tvorbu určitého textového řetězce, který pak bude zakódován. QR kód vytvořený na základě dat zadaných v této aktivitě bude reprezentovat telefonní číslo. Po načtení takového kódu smartfon nabídne uživateli zavolat na nalezené číslo.
- **UrlActivity** – aktivita bude určena pro zadání odkazu na webovou stránku. Odpovídá za tvorbu určitého textového řetězce, který pak bude zakódován. QR kód vytvořený na základě dat zadaných v této aktivitě bude reprezentovat odkaz na webovou stránku. Po načtení takového kódu smartfon automaticky otevře prohlížeč se zadanou adresou nalezenou v kódu.
- **TextActivity** – aktivita bude určená pro zadání libovolných textových dat. Odpovídá za tvorbu určitého textového řetězce, který pak bude zakódován. QR kód vytvořený na základě dat zadaných v této aktivitě bude reprezentovat samotný text. Po načtení takového kódu smartfon automaticky zobrazí nalezený text.
- **SmsActivity** – aktivita bude určena pro zadání textových dat reprezentujících telefonní číslo a SMS zprávu. Odpovídá za tvorbu určitého textového řetězce, který pak bude zakódován. QR kód vytvořený na základě dat zadaných v této aktivitě bude reprezentovat SMS zprávu. Po načtení takového kódu smartfon automaticky předloží uživateli možnost odeslat nalezený text na nalezené telefonní číslo SMS zprávou.
- **WiActivity** – aktivita bude určena pro zadání názvu bezdrátové sítě (SSID), hesla a výběru určitého typu šifrování. Odpovídá za tvorbu určitého textového řetězce, který pak bude zakódován. QR kód vytvořený na základě dat zadaných v této aktivitě bude reprezentovat Wi-Fi připojení. Po načtení takového kódu předloží smartfon automaticky uživateli možnost připojit se k určité bezdrátové síti podle nalezených údajů.
- **SuccessActivity** – tato aktivita bude určena pro zobrazení vytvořeného černobílého QR kódu a umožní uživateli zvolit jiný stupeň korekce chyb, čímž ho předělá.
- **CustomizeActivity** – je konečnou aktivitou, která bude určena pro doplnění o grafiku vytvořeného QR kódu a uložení ho do paměti smartfonu. Odpovídá za nastavení gradientu, barev popředí a pozadí, výběr a nastavení obrázku

z galerie, uložení QR kódu jako PNG obrázku a zaslání pomocí služby ContentProvider.

### 6.1.2 Třídy

Pro tvorbu samotného QR kódu bylo rozhodnuto použít externí knihovnu. Ze všech existujících řešení se zdálo jako nejvhodnější řešení použít open source project Zxing. Jeho výhodou je to, že podporuje různé platform, do nichž patří i Android. Má licenci Apache License, Version 2.0, což znamená možnost použití zdrojového kódu jak ve volných, tak i v komerčních projektech. Celá knihovna je dobře dokumentovaná a používá se ve většině aplikací stejného druhu. Projekt je dostupný na GitHub- [<https://github.com/zxing/zxing>].

Při tvorbě aplikace budou použity následující třídy knihovny:

- BitMatrix – tato třída reprezentuje 2D matici bitu, která pak bude interpretována jako obrázek.
- QRCodeWriter – tato třída odráží QR-kód jako 2D masiv hodnot.
- Encode – třída, která odpovídá vlastně za kódování na základě vstupních parametrů (textového řetězce, úrovně korekce chyb a formát kódování řetězců).

Dále bude vytvořena třída QRcreate, která pomocí výše uvedených tříd vytvoří QR kód a vrátí ho v podobě obrázku, se kterým bude možné dále pracovat.

## 6.2 Implementace aplikace

### 6.2.1 QRcreate

QRcreate je třída, která obsahuje metody pro tvorbu QR kódu a vrácení ho jako obrázku. Třída obsahuje 3 globální proměnné – **WIDTH**, **HEIGHT**, **count**. První dvě proměnné určují rozměry budoucího obrázku, je to čtverec o straně odpovídající šířce obrazovky zařízení, na kterém běží aplikace. Třetí proměnná uloží hodnotu definující odsazení mezi QR kódem a hranou obrázku. Třída obsahuje metodu **qrBitmap()**, jež přijímá na vstup 3 proměnné typu integer a nedefinovaný počet textových řádků, které pak se představují jako masiv **params**. První dvě číselné proměny představují barvy RGBA modelu<sup>3</sup> a třetí představují úroveň korekce chyb zvolenou uživatelem. Na základě masivu **params**, Hash

---

<sup>3</sup> Aditivní barevný model, popisující způsob kódování barvy pro její reprodukci.

tabulky, obsahující zvolenou úroveň korekce chyb a formát kódování (UTF-8), pomocí třídy `QRCodeWriter` vytvoří matici bitů, která pak pomocí metody `matrixToBitmap()` bude představena jako obrázek. Metoda `matrixToBitmap()` projde po řádcích a sloupcích maticí a udělí každému bitu buď jednu barvu, nebo druhou. Zároveň bude v této metodě realizováno počítadlo, které při procházení řádků matice zaznamená jejich počet jedné barvy až do prvního bitu, kterému bude přidělena jiná barva, hodnota bude uložena do proměnné `count`, jež definuje odsazení samotného QR kódu od nahraného obrázku. Nasledující část kódu reprezentuje část metody `qrBitmap()`, ve které se snaží zakódovat řetězec pomocí třídy `QRCodeWriter`, jako parametry zasílá textový řetězec, druh kódu (tato knihovna dokáže zakódovat text i do jiných 2D kódů), délku a šířku potřebného kódu a také úroveň korekce chyb.

```
for (int i = 0; i < params.length; i++) {
    try {
        BitMatrix matrix = new QRCodeWriter().encode(
            params[i],
            com.google.zxing.BarcodeFormat.QR_CODE,
            WIDTH, HEIGHT, hints);

        bitmap = matrixToBitmap(front, back, matrix);

    } catch (WriterException e) {
        e.printStackTrace();
    }
}
```

Příklad 3

### 6.2.2 MainActivity

`MainActivity` je vstupní aktivitou, která se spouští metodou `onCreate()` při spuštění aplikace uživatelem. Grafické rozhraní je realizováno v xml souboru `activity_main.xml`. Za aktivní prvky má sadu `RadioButtons` reprezentující výběr mezi tím, co chce uživatel zakódovat, a tlačítko „NEXT“, po jehož stisknutí bude přeměřován na aktivitu odpovídající jeho výběru pomocí objektu `Intent`.

### 6.2.3 VizitkaActivity

`VizitkaActivity` je aktivitou určenou primárně pro zadání textových dat a jejich formulování do určitého textového řetězce, který pak bude zakódován. Spouští se metodou `onCreate()`. Grafické rozhraní je realizováno v xml souboru `activity_vizitka.xml`. Za aktivní prvky má dvě tlačítka. Jedno je určeno pro načtení dat z kontaktů uživatele pomocí

metody **pickContact()** a druhé pro formulování určitého textového řetězce dle zadaných textových dat a odeslání ho do aktivity, v níž bude výsledný obrázek zakódován a umístěn na obrazovce. Toto je realizováno metodou **onClickQR()**. Dále je za sebou směrem shora dolů rozmístěno 11 prvků, které jsou určeny pro vkládání textových dat (EditText).

Aktivita také obsahuje 2 statické proměnné typu integer určených pro zaslání systémových zpráv při přidělení přístupových práv aplikace ke čtení kontaktu. Toto bude realizováno pomocí řady metod navázaných na tlačítko pro výběr kontaktů. Jeho stisknutím bude spuštěna metoda **hasPermission()**, jež má za úkol ověřit, jestli je aplikaci přiděleno právo na čtení dat z kontaktu, pokud ano, bude vrácena hodnota „true” a spuštěna metoda pro výběr kontaktu, pokud ne, bude vrácena hodnota „false” a spuštěna metoda **requestPermissionWithRationale()**, která vytvoří oznámení „Snackbar”, obsahující zprávu uživateli, proč potřebuje tato práva, a tlačítko „Grant”, po jehož stisknutí bude spuštěna metoda **requestPerms()**, ta zašle požadavek systému v podobě zprávy obsahující účel a kód uložený ve statické proměnné. Tím, že systém odpoví, spouští metodu **onRequestPermissionsResult()**, která se zeptá uživatele na přístup ke kontaktům. Pokud to uživatel povolí, bude otevřen seznam s kontakty uživatele, po výběru určitého kontaktu, jména a telefonního čísla budou importované do aktivity a doplněné do určitých políček. Výběr dat z „kontaktních” tabulek bude prováděn metodou **onActivityResult()** pomocí rozhraní Cursor. Toto rozhraní poskytuje přístup pro čtení a zápis v sadě výsledků vrácené databázovým dotazem.

Druhé tlačítko „Create your QR code” spustí metodu **onClickQR()**, která agreguje všechna textová data obsazená v polích do jedné proměnné typu string a zašle na spravování pomocí objektu Intent do další aktivity **SuccessActivity**.

#### 6.2.4 TelefonActivity

TelefonActivity je aktivita určená pro zadání textových dat reprezentujících telefonní číslo. Grafické rozhraní aktivity je realizováno v xml souboru `activity_telefon.xml`. Obsahuje pouze pole pro zadání telefonního čísla (EditText) a tlačítko „Create your QR code”, které stejným způsobem jako ve **VizitkaActivity** agreguje textová data a data zasláná pro další zpracování do aktivity **SuccessActivity**.

### 6.2.5 SMS Activity

SMS Activity je aktivita určená pro zadávání textových dat reprezentujících telefonní číslo a SMS zprávu. Grafické rozhraní aktivity je realizováno v xml souboru `activity_sms.xml`. Na rozdíl od předcházející aktivity má navíc ještě jeden aktivní prvek `EditText` pro vložení zprávy. Následující část kódu reprezentuje metodu `onClickQR()`, která odpovídá za formulování textového řetězce, který pak bude zpracován v **SuccessActivity**.

```
public void onClickQR(View view)
{
    EditText edPhone =(EditText) findViewById(R.id.tel);
    EditText edMess =(EditText) findViewById(R.id.message);

    String s="smsto:"+edPhone.getText().toString()+":"
+edMess.getText().toString();

    if (s.length() > 7) {
        Intent intent = new Intent(this, SuccessActivity.class);
        intent.putExtra("data", s);
        startActivity(intent);
    } else {
        Toast.makeText(this, "There is nothing to code.",
Toast.LENGTH_SHORT).show();
    }
}
```

#### Příklad 4

### 6.2.6 TextActivity

TextActivity je aktivita určená pro zadání textových dat reprezentujících samotný text, který pak bude zakódován. Grafické rozhraní aktivity je realizováno v xml souboru `activity_text.xml`. Aktivita obsahuje pouze dva aktivní prvky `EditText` pro zadání textu a tlačítko „Create your QR code”, které stejným způsobem jako v předchozích aktivitách vytvoří textový řetězec a odešle ho pro další zpracování v aktivitě **SuccessActivity**.

### 6.2.7 UrlActivity

UrlActivity je aktivita určená pro zadávání textových dat reprezentujících odkaz na webové stránky. Grafické rozhraní aktivity je realizováno v xml souboru `activity_url.xml`. Aktivita se téměř neliší od TextActivity, ale při realizaci grafického rozhraní této aktivity u `EditText` prvku byl nastaven atribut `text`, který předem vkládá do pole textový řetězec „http://” což zjednoduší proces vkládání webového odkazu uživatelem.

## 6.2.8 WiActivity

WiActivity je aktivita určená pro zadávání textových dat reprezentujících údaje pro připojení k bezdrátové síti. Grafické rozhraní aktivity je realizováno v xml souboru activity\_wi.xml. Obsahuje dvě textová pole (EditText) a jeden výběrový list (Spinner) a tlačítko, po jehož stisknutí metoda **onClickQR()** agreguje zadaná data do určitého textového řetězce a zašle je pro další zpracování do **SuccessActivity**. První textové pole je určeno pro zadání názvu sítě, tzv. SSID, druhé pole je určeno pro zadání hesla pro připojení k této síti. Spinner umožní uživateli výběr pro označení typu šifrování existující sítě buď WEP, nebo WPA.

## 6.2.9 SuccessActivity

SuccessActivity je aktivita určená pro vytvoření a zobrazení černobílého QR kódu podle textových dat nalezených v jedné ze šesti výše popsaných aktivit a umožnění uživateli nastavit jednu ze čtyř úrovní korekce chyb (implicitně nastavena úroveň M). Grafické rozhraní aktivity je realizováno v xml souboru activity\_success.xml. Obsahuje oblast pro rozmístění obrázku (ImageView), Spinner pro zvolení jiné úrovně korekce, tlačítko, po jehož stisknutí se spustí metoda **onClickChangeQR()**, ta vytvoří nový QR kód s ohledem na výběr uživatele. Následující část kódu (Příklad 5) zobrazí, jakým způsobem aktivita dostane textový řetězec pro kódování a odešle ho pro kódování jako proměnnou typu string spolu s hodnotami označujícími barvu popředí, pozadí a úroveň korekce a uplatní výsledný obrázek na obrazovce.

```
imageView= findViewById(R.id.imgv);
Bundle extras = getIntent().getExtras();
if (extras != null)
{
    String s =extras.getString("data");
    Bitmap bitmap=new QRcreate().qrBitmap(FRONT, BACK, LEVEL, s);
    imageView.setImageBitmap(bitmap);
}
```

### Příklad 5

Dále obsahuje druhé tlačítko „Customize your QR code“, které spustí další aktivitu CustomizeAktivity pomocí objektu Intent a zároveň přepoše proměnnou s textovým řetězcem a proměnnou definující stupeň korekce chyb.

### 6.2.10 CustomizeActivity

CustomizeActivity je konečnou aktivitou, ve které probíhá doplnění QR o pomocnou grafiku a jeho uložení jako png obrázku. Grafické rozhraní aktivity je realizováno v xml souboru activity\_customize.xml. Tato aktivita obsahuje 8 tlačítek a jedno pole pro zobrazení obrázku (ImageView).

Hlavní myšlenka doplnění QR kódu o pomocnou grafiku spočívá v tom, že se vytvořila pod bitmapou obsahující QR kód dodatečná vrstva, která bude obsahovat buď gradient, nebo obrázek, vybraný uživatelem z galerie. Přitom Bitmapa obsahující QR kód bude generovaná znovu, ale tentokrát bude za barvu samotného kódu zvolena barva RGBA modelu, ve které „alfa“ kanál bude nastaven na nulovou hodnotu, což udělá QR průhledným. Následující část kódu je metoda **createImage()**, která řídí sloučení dvou vrstev do jednoho objektu Drawable určitým způsobem.

```
public void createImage()
{
    Drawable d = new BitmapDrawable(getResources(), qrLayer);
    Drawable dd=firstLayer;
    layers= new Drawable[2];
    layers[0]=dd;
    layers[1]=d;
    LayerDrawable layerDrawable = new LayerDrawable(layers);
    int i=QRcreate.count;
    layerDrawable.setLayerInset(0, i, i, i, i);
    imageView.setImageDrawable(layerDrawable);
}
```

#### Příklad 6

Proměna „d“ v této metodě obsahuje objekt drawable, který reprezentuje vrstvu QR kódů a proměna „dd“ je pozadím. Tato metoda sjednotí vrstvy takovým způsobem, že vrstva reprezentující podlahu bude uplatněna uprostřed a s rovnoměrným odsazením po stranách o délku, která je uvedena v proměnné „count“ ve třídě **createQR**, čímž napodobí rozměr QR kódů.

První tři tlačítka určená pro nastavení gradientu. Pomocí prvního tlačítka uživatel zvolí počáteční barvu a druhým konečnou barvu. Stisknutím třetího tlačítka „Set gradient“ uživatel potvrdí výběr barev a spustí metodu **setGrad()**. Ta vytvoří objekt třídy drawable-čtverec plněný gradientem směrem shora dolů a uplatní ho do vrstvy reprezentující pozadí.

Pro výběr barev byl použit open source „Colorpicker by QuadFlask“, který má licence Apache License, Version 2.0, celý projekt je dostupný na GitHub –



[<https://github.com/QuadFlask/colorpicker>]. Tento „colorpicker“ umožňuje implementovat nástroj pro výběr barev jako dialogové okno.

Další dvě tlačítka jsou určena pro výměnu barev horní vrstvy, která reprezentuje QR kód. Tlačítko „Set front color“ umožní změnit barvu samotného QR kódu. Tlačítko „Set back color“ umožní změnit barvu „tiché zóny“.

Tlačítko „Set photo“ je určeno pro výběr obrázku z galerie uživatelem. Na toto tlačítko je navázána metoda **photoPickInit()**, která zajistí existenci ve smartfonu externího úložiště, které je přípustné ke čtení, pokud existuje, ověří, jestli je aplikaci přiděleno právo ke čtení dat, pokud ano, bude spuštěna metoda **photoPick()**, která zašle požadavek na čtení obrázku z galerie a obrázek vybraný uživatelem určitým způsobem zpracuje. Pokud aplikaci není povoleno čtení dat, podobným způsobem jako v aktivitě VizitkaActivity bude uživatel dotázán, aby povolil určité oprávnění aplikace. Následující část kódu reprezentuje dvě metody pro výběr a vkládání obrázku do vrstvy.

```
public void photoPick()
{
    Intent photoIntent = new Intent(Intent.ACTION_PICK);
    photoIntent.setType("image/*");
    startActivityForResult(photoIntent, REQUEST_PERMISSION_READ);
}

@Override
public void onActivityResult(int requestCode, int resultCode,
Intent data) {
    super.onActivityResult(requestCode, resultCode, data);
    Bitmap bitmap=null;
    switch (requestCode)
    {
        case REQUEST_PERMISSION_READ:
            if(resultCode==RESULT_OK)
            {
                Uri selectedImage = data.getData();
                try {
                    bitmap =
MediaStore.Images.Media.getBitmap(getContentResolver(), selectedImage);
                } catch (IOException e)
                {
                    e.printStackTrace();
                }
                firstLayer=new BitmapDrawable(getResources(),
createScaledBitmap (bitmap, QRcreate.WIDTH, QRcreate.HEIGHT, false));
                createImage();
            }
            break;}}
```

Příklad 7

Předposlední tlačítko „Save QR code” je určeno pro uložení obsahu ImageView jako obrázkový soubor ve formátu png. Metoda **photoSaveInit()** je navázaná na toto tlačítko stejně jako i v případě s výběrem obrázku z galerie. Metoda ověří existenci externího úložiště, a jestli aplikace má oprávnění do něj zapisovat. Pokud má aplikace všechna potřebná oprávnění, spouští se metoda **photoSave()**, která vezme objekt Drawable nacházející se v ImageView, převede ho do bitmapy a pošle na vstup metody saveBitmap, který vypadá následovně:

```
private File saveBitmap(Bitmap bitmap) {  
  
    File folder=new File(Environment.getExternalStoragePublicDirectory  
    (Environment.DIRECTORY_PICTURES)+"/myQR");  
    if (!folder.exists())  
    {folder.mkdir();}  
    SimpleDateFormat sdf=new SimpleDateFormat("yyymmsshhmmss");  
    String date= sdf.format(new Date());  
    String name= "Img"+date+".png";  
  
    FileOutputStream fOut=null;  
    File file = new File(folder, name);  
    try {  
        fOut=new FileOutputStream(file);  
        bitmap.compress(Bitmap.CompressFormat.PNG, 100, fOut);  
        fOut.flush();  
        fOut.close();  
        Toast.makeText(this, "File is saved", Toast.LENGTH_SHORT).show();  
        refreshGal(folder);  
    }catch (Exception e) {  
        e.printStackTrace();  
    }  
    return file;  
}
```

#### Příklad 8

Podstata této metody spočívá v tom, aby zaprvé zajistila místo k uložení souboru (v daném případě vytvoří složku „myQR” ve složce „Pictures” ve vnějším úložišti zařízení). Dále vytvoří unikátní název souboru skládajícího se z časového okamžiku, v němž probíhá vytváření, pak ho uloží jako png obrázek, oznámí uživateli úspěšnost provedené operace a zašle signál pro obnovení galerie, aby uložený obrázek byl lehce dosažitelný.

Poslední tlačítko „Share QR code” má za cíl umožnit podílet se vytvořeným obrázkem přes jiné aplikací, například „WhatsApp” nebo „Gmail”. Řešeno je to pomocí

poskytovatele obsahu (Content Provider). Metoda, která toto zabezpečuje, je **photoShare()**. Obecně tato metoda také uloží soubor jako png obrázek prostřednictvím výše popsané metody **saveBitmap()** a vytvořený soubor, který metoda vrátí, bude nabídnut pro otevření jiným aplikacím, které dokážou tento typ souboru otevřít.

### 6.3 Testování

Výsledná aplikace byla otestovaná na následujících mobilních zařízeních:

- Aligator S4080 (Android 6.0-Marshmallow) s rozlišením – 800x480px
- Samsung Galaxy S4 zoom (Android 4.4-KitKat) s rozlišením displeje – 960x540px
- Samsung Galaxy A3 (Android 6.0-Marshmallow) s rozlišením displeje – 1280x720px

Protože kvalita obrázku výsledného QR kódu záleží na rozlišení displeje, na němž byl vygenerován, důraz byl kladen na QR kódy vytvořené v aplikaci na smartfonu Aligator S4080. Na následujícím obrázku jsou zobrazeny QR kódy s nastaveným stupněm korekce H, reprezentující odkaz na webovou stránku Provozně ekonomické fakulty [<http://pef.czu.cz>] a vygenerování právě vytvořené aplikace. Podle metody, s jejíž pomocí byl prováděn výzkum v kapitole 4.1, byl otestován každý z uvedených kódů pro zajištění jejich čitelnosti. První QR byl tak načten v deseti případech z deseti, stejný výsledek prokázal i druhý QR kód, ale třetí byl úspěšně načten v osmi případech z deseti.



Obrázek 15 – Vygenerované QR kódy

Při testování aplikace nebyly nalezeny žádné chyby, které by vyvolaly dočasné ukončení aplikace nebo nějak omezily použití aplikací uživatelem. Na základě výše uvedených

výsledků bylo zadané řešení považované za úspěšné a výsledná verze aplikace byla zveřejněna v Play Marketu. Při publikování byl zadán kontrolní název aplikace „*QR Editor*” a byly stanoveny státy, ve kterých lze tuto aplikaci stahovat, a to Česká republika a Ruská federace.

Načtením následujícího QR kódu budete přesměrováni ke stažení vytvořené aplikace v Play Marketu.



Obrázek 16 – Odkaz na stažení aplikace

## 7 Zhodnocení výsledku

Výsledná aplikace dokáže vytvořit QR kód podle zadaných textových dat, jež mohou reprezentovat jak samotný text, tak i elektronickou vizitku, wi-fi konfiguraci, SMS zprávu, telefonní číslo nebo odkaz na webovou stránku. Také lze definovat, jaký stupeň korekce chyb bude mít výsledný QR kód. Dále lze výsledný kód doplnit o pomocnou grafiku. Konkrétně umožní nastavit různé barvy na popředí a na pozadí, obarvit výsledný QR kód přes gradient a nebo jakýkoliv obrázek nacházející se u uživatele v galerii. Konečný QR kód je čitelný, lze ho uložit do paměti a odeslat prostřednictvím jakéhokoliv e-mailu klienta nainstalovaného do zařízení nebo messengeru.

Cílová skupina zájemců o aplikaci jsou lidé ve věku od 15–40 let. Při zveřejnění aplikace na Play Marketu jí byl přidělen rating 3+. Aplikace je vhodným nástrojem pro mobilní marketing, ale může být použita i pro zábavu.

Při vývoji aplikace byl řešen pouze jeden problém, konkrétně to, jakým způsobem kombinovat vytvořenou bitmapu s QR kódem a jednotlivým obrázkem tak, aby kód zůstal čitelný. Jako první řešení byla realizovaná metoda, která měla na vstupu vytvořenou bitovou matici. Tato metoda procházela jednotlivými bity a nastavovala pro každý pixel určitou barvu. Byla ale příliš náročná a není racionální při použití na QR kódech reprezentujících vCard, které mají minimální sedmou verzi (45x45 modulů). Konečné řešení s průhledným kódem je příliš jednoduché a dalo se snadno implementovat. Jeho hlavní výhodou je to, že pokud při použití obrázku, který má stejné barvy jako pozadí QR kódu, lze pomocí nastavení barvy popředí regulovat průhlednost QR kódu a tím zabezpečit kontrast a zachovat čitelnost kódu.

Grafické rozhraní této aplikaci není silnou stránkou, protože při návrhu byly použity standardní styly jednotlivých prvků. Byla ale dodržena všechna pravidla pro zajištění responsivity aplikace. Jednotlivé prvky byly rozmístěny tak, aby aplikace byla intuitivní z hlediska použití. Všechny atributy prvku byly definované v jednotlivých XML souborech.

V samostatném XML souboru byly definované všechny textové hodnoty jednotlivých prvků, což lze využít v budoucnu pro překlad celé aplikace do různých jazyků. Nyní je aplikace vytvořena v angličtině.

Minimální verze operačního systému, na níž se dá nainstalovat aplikace, je Android 4.4 KitKat (API 19). Aplikace bude vyžadovaná k oprávnění ke čtení kontaktu, externí paměti a zápisu souboru do této paměti.

## 8 Závěr

Smyslem bakalářské práce bylo vytvořit mobilní nástroj, který umožní nejen QR kód vytvořit, ale udělat ho přívětivějším, aby přitahoval pozornost jiných lidí, čemuž podle názoru autora výsledná aplikace odpovídá.

Aplikaci lze ale vyvíjet i dál. Kromě uživatelské přívětivosti z hlediska grafického rozhraní ji lze výrazně rozšířit o funkcionál, a to zaprvé implementováním čtečky, což zvětší skupinu zájemců o tuto aplikaci, a zadruhé lze implementovat generátor odkazu s UTM parametry, což umožní vytvářet QR kódy, které pak bude možné sledovat z hlediska použitelnosti. K tomu lze přidat i tzv. „URL shortener“, který výrazně zkracuje odkazy.

Při tvorbě této práce byly provedeny dva výzkumy. Cílem prvního výzkumu bylo zjistit, jak velká plocha QR kódu může být poškozena, aby kód neztratil svou čitelnost v závislosti na zvolené úrovni korekce při vytvoření kódu. Další výzkum měl za cíl ověřit vliv okolností při snaze načíst poškozený kód. Výzkum spočíval v tom, že se snažil načíst kódy použité v předchozím výzkumu a nacházející se v různých polohách vůči čtecímu zařízení a při různých úrovních osvětlení místnosti. Dle nalezených výsledků výzkumu lze tvrdit, že při tvorbě QR kódu je nejlepší variantou pro volbu stupně korekce chyb úroveň H, protože u testovaného kódu s tímto stupněm korekce se dala poškodit plocha o maximální velikostí 13x11 modulu, přitom bude zachována čitelnost kódu. Při takovýchto parametrech nebylo čtení kódu žádným způsobem ovlivněno okolnostmi popsanými ve druhém výzkumu (viz kapitola 4.1).

Při návrhu aplikace byl také navržen způsob kombinace QR kódu a pomocné grafiky. Konkrétně se jedná o navrstvení generovaného QR kódu na obrázek zvolený uživatelem. Podstatou takového způsobu je to, že generovaný QR kód má místo černé barvy nastavenou „průhlednou“ barvu a obrázek zvolený uživatelem je umístěn pod QR kódem. Výhodou tohoto způsobu je to, že uživatel může regulovat průhlednost QR kódu, čímž zabezpečí dostatečný kontrast pro jeho lepší čitelnost. Formou elektronické přílohy byl k této práci přidán archivační soubor obsahující celý projekt se zdrojovým kódem a obrázky UML diagramu.

## 9 Seznam použitých zdrojů

- 1) *QRcode.com / DENSO WAVE*. [Online] Denso Wave Inc. [cit. 2018-02-12]  
Dostupné z: <http://www.qrcode.com/en/>.
- 2) *ISO - International Organization for Standardization*. [Online] [cit. 2018-02-12]  
Dostupné z:  
[http://www.iso.org/iso/home/store/catalogue\\_tc/catalogue\\_detail.htm?csnumber=62021](http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=62021).
- 3) **E.I.PRYAKHIN**, Dr. of Engineering Sciences, Professor. *Nanobar code as multi-purpose two-dimensional notation with new features* [online].(PDF).[cit. 2018-02-12]. Dostupné z WWW: <https://cyberleninka.ru/article/n/nanobar-kod-mnogofunktsionalnaya-dvuhmernaya-simvolika-s-novymi-vozmozhnostyami.pdf>
- 4) Wikipedia. *Barcode*[online].[cit. 2018-02-12]. Dostupné z WWW:  
<https://en.wikipedia.org/wiki/Barcode>
- 5) **Pavel Krizhepolsky**, Android mobile-review. *2D Barcodes*[online].[cit. 2018-02-12]. Dostupné z WWW: <http://android.mobile-review.com/market/10752/>
- 6) Cognex, manatee works. *What is MaxiCode?*[online].[cit. 2018-02-12].  
Dostupné z WWW: <https://manateeworks.com/maxicode>
- 7) Geektimes. *Základní vlastnosti QR* [online].[cit. 2018-02-12].  
Dostupné z WWW: <https://geektimes.ru/post/63913/>
- 8) Geektimes. *Čtení QR kódu* [online].[cit. 2018-02-12]. Dostupné z WWW:  
<https://geektimes.ru/post/256932/>
- 9) **BUGAEV,L.** *Mobilny marketing: Kak zarzadit svoj biznes v mobilnom mire*.  
Pervoe izdanie. Moskva: OOO «Alpina pabliher», 2012. 978-5961422221
- 10) Sostav. *Do we need QR codes?* [online].[cit. 2018-02-12].  
Dostupné z WWW: <http://www.sostav.ru/publication/nuzhny-li-qr-kody-4183.html>
- 11) Creambee. *Non-standard QR Codes* [online].[cit. 2018-02-12].  
Dostupné z WWW: <http://creambee.ru/blog/post/create-nonstandard-qr-code/>
- 12) **KOLISNICHENKO, D.N.** *Programirovanie dlya Android 5*. Pervoe izdanie.  
Sankt-Peterburg: "BHV-Peterburg", 2015. 978-5977535489
- 13) *Android.com*[online]Google.[cit. 2018-02-12].  
Dostupné z WWW: <https://developer.android.com/develop/index.html>
- 14) **Sean Owen**. GitHub. *Barcode contents*. [online].[cit. 2018-02-12]. Dostupné z WWW: <https://github.com/zxing/zxing/wiki>

15) Wikipedia. *Android*[online],[cit. 2018-02-12]. Dostupné z WWW:

<https://ru.wikipedia.org/wiki/Android>

16) Lotus Development Corporation. *vCard MIME Directory Profile*. [online],[2018-02-12]. Dostupné z WWW: <https://tools.ietf.org/html/rfc2426>



## **10 Přílohy**

Práce obsahuje pouze jednu přílohu a je to archivační soubor xarav001.zip obsahující celý projekt se zdrojovým kódem a obrázky UML diagramu.