Elektrotechnika a informatika

# Mobilní aplikace pro ověřování čárových kódů

Bakalářská práce

Autor práce: Miroslav Zápařka

Vedoucí práce: Ing. Jan Hrabák

Jihlava 2021



# ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Autor práce:	Miroslav Zápařka
Studijní program:	Elektrotechnika a informatika
Obor:	Aplikovaná informatika
Název práce:	Mobilní aplikace pro ověřování čárových kódů
Cíl práce:	Vytvoření mobilní aplikace pro mobilní operační systém Android, která bude umět na základě naskenování čárového kódu z etikety společnosti PPL CZ s.r.o. vyhodnotit, zda je použitý čárový kód správného typu a se správným počtem znaků. Jedná se o rozpoznání dvou typů čárového kódu, a to čárového kódu Code 128 s počtem znaků 17 a Interleaved 2 of 5 with checksum s počtem znaků 12.

Ing. Jan Hrabák

vedoucí bakalářské práce

doc. Ing. Zdeněk Horák, Ph.D. vedoucí katedry

Katedra technických studií

## Abstrakt

Cílem této bakalářské práce je vytvořit mobilní aplikaci pro Android, která bude umět ověřit správnost čárových kódu používaných ve společnosti PPL CZ s.r.o., která je lídrem v přepravě kusových zásilek. Mobilní aplikace je realizována v programovacím jazyku Java a umí vyhodnocovat čárové kódy typu Code 128 a Interleaved 2 of 5 a ignoruje ostatní čárové kódy.

## Klíčová slova

Čárový kód; mobilní aplikace; Java; Android; Code 128; Interleaved 2 of 5

### Abstract

The aim of this bachelor's thesis is to create a mobile application for Android, which will be able to verify the correctness of barcodes used in the company PPL CZ s.r.o., which is a leader in the transport of piece shipments. The mobile application is implemented in the Java programming language and can evaluate bar codes of the Code 128 and Interleaved 2 of 5, ignoring other barcodes.

# Keywords

Barcode; Mobile application; Java; Android; Code 128; Interleaved 2 of 5

Prohlašuji, že předložená bakalářská práce je původní a zpracoval/a jsem ji samostatně. Prohlašuji, že citace použitých pramenů je úplná, že jsem v práci neporušil/a autorská práva (ve smyslu zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů, v platném znění, dále též "**AZ**").

Byl/a jsem seznámen/a s tím, že na mou bakalářskou práci se plně vztahuje AZ, zejména § 60 (školní dílo).

Podle § 47b zákona o vysokých školách souhlasím se zveřejněním své práce podle směrnice prorektora pro studium č. 2/2020, a to bez ohledu na výsledek obhajoby.

Beru na vědomí, že VŠPJ má právo na uzavření licenční smlouvy o užití mé bakalářské práce a prohlašuji, že **s o u h l a s í m** s případným užitím mé bakalářské práce (prodej, zapůjčení apod.).

Jsem si vědom/a toho, že užít své bakalářské práce či poskytnout licenci k jejímu využití mohu jen se souhlasem VŠPJ, která má právo ode mě požadovat přiměřený příspěvek na úhradu nákladů, vynaložených vysokou školou na vytvoření díla (až do jejich skutečné výše), z výdělku dosaženého v souvislosti s užitím díla či poskytnutím licence.

V Jihlavě dne 26. července 2021

.....

Podpis studenta/ky

# Poděkování

Rád bych poděkoval panu Ing. Janu Hrabákovi za vedení a podporu při řešení a možnosti volnější pracovní doby na úkor této práce. Dále mé poděkování patří hlavně rodině za její podporu a shovívavost při trávení volného času nad touto prací místo s nimi.

# Obsah

Se	eznam o	obrázk	.ů	. 8
Se	eznam t	tabule	k	10
Se	eznam z	zkratel	k	11
Ú	vod			12
1	Mob	oilní ap	plikace	13
	1.1	Andro	oid Studio	13
	1.2	Java		13
2	Cáro	ové kó	dy	14
	2.1	Histor	rie čarových kodu	14
	2.2	Prenie 1 (	ed hejpouzivanejsich carových kodu	14 15
	2.2.1	т ( 2 і		15
	2.2.3	- ` 3 l	UPC-E0	15
	2.2.4	4 E	EAN-8 a EAN-13	16
	2.2.5	5 (	Code 93	16
	2.2.6	6 I	Interleaved 2 of 5 a Interleaved 2 of 5 Mod 10	16
	2.2.7	7 (	Code 128	17
	2.2.8	8 (	Codabar	17
	2.2.9	9 1	MSI	17
	2.3	Code	128	18
	2.3.1	1 (	Code 128 A	21
	2.3.2	2 (	Code 128 B	21
	2.3.3	3 (	Code 128 C	22
	2.4	Interl	eaved 2 of 5 with modulo 10 checksum	22
3	O PF	PL CZ s	5.r.o	25
	3.1	Etiket	ta PPL	25
	3.1.1	1 9	Standardní etiketa	25
_	3.1.2	2 9	Smart Label	26
4	Vývo	oj aplil		27
	4.1	Pozac	davky na funkcnost	27
	4.2	Realiz	zače aplikace	27 27
	4.2.1	, i	Skenovací obrazovka	27 28
	4.2.2	3 1	Hlavní obrazovka	28
	4.2	- '		
	4.3	Ukazk 1 S	spouštění aplikace	30 30
	4.3.2	2 F	Hlavní obrazovka aplikace	32

b Závi	iviozne v	ylepseni aplikace	. 44 . 45
c	Možné su	vlančaní anlikaca	
	5.3.3	Výsledek testu ostatní čárových kódů	. 42
	5.3.2	Codabar	. 42
	5.3.1	EAN-8	. 42
5	.3 Osta	itní čárové kódy	. 42
	5.2.3	Testování Interleaved 2 of 5 s méně znaky v čárovém kódu	. 41
	5.2.2	Testování Interleaved 2 of 5 s více znaky v čárovém kódu	. 40
	5.2.1	Testování Interleaved 2 of 5 se právním počtem znaků	. 39
5	.2 Inte	rleaved 2 of 5	. 39
	5.1.3	Testování Code 128 s méně znaky v čárovém kódu	. 38
	5.1.2	Testování Code 128 s více znaky v čárovém kódu	. 37
	5.1.1	Testování Code 128 se správným počtem znaků	. 36
5	.1 Cod	e 128	. 36
5	Testován	í aplikace	. 36
	4.3.4	Info	. 34
	4.3.3	Nastavení	. 33

# Seznam obrázků

Obrázek 1 - Code 39	. 15
Obrázek 2 - UPC-A	. 15
Obrázek 3 - UPC EO	. 15
Obrázek 4 – EAN-8	. 16
Obrázek 5 – EAN-13	. 16
Obrázek 6 - Code 93	. 16
Obrázek 7 - Interleaved 2/5	. 16
Obrázek 8 - Interleaved 2/5 Mod 10	. 17
Obrázek 9 - Code 128	. 17
Obrázek 10 – Codabar	. 17
Obrázek 11 - MSI	. 17
Obrázek 12 - Code 128 B	. 21
Obrázek 13 - Code 128 B - struktura kódu	. 21
Obrázek 14 - přepis kontrolní číslice do čárového kódu	. 22
Obrázek 15 - Interleaved 2 of 5 - start znak	. 23
Obrázek 16 - Interleaved 2 of 5 - stop znak	. 23
Obrázek 17 - Interleaved 2 of 5 with modulo 10 checksum	. 24
Obrázek 18 - standardní etiketa	. 25
Obrázek 19 - Smart Label etiketa	. 26
Obrázek 20 - Welcome – nastavení a spuštění animace	. 27
Obrázek 21 - Welcome – odložení startu aplikace	. 28
Obrázek 22 - BarcodeActivity – uložení hodnot	. 28
Obrázek 23 - MainActivity – uložení dat z BarcodeActivity	. 29
Obrázek 24 - MainActivity – kontrola typu a délky čárového kódu	. 29
Obrázek 25 - MainActivity – výpočet odchylky počtu znaků	. 30
Obrázek 26 - Ukázka aplikace – start	. 31
Obrázek 27 - Ukázka aplikace – start	. 31
Obrázek 28 - Ukázka aplikace – start	. 32
Obrázek 29 - Ukázka aplikace – Hlavní obrazovka	. 33
Obrázek 30 - Ukázka aplikace – Nastavení	. 34
Obrázek 31 - Ukázka aplikace – Info	. 35
Obrázek 32 - Code 128 pro testování – správný počet znaků	. 36
Obrázek 33 - Výsledek Code 128 pro testování – správný počet znaků	. 36
Obrázek 34 - Code 128 pro testování – více znaků	. 37
Obrázek 35 - Výsledek Code 128 pro testování – více znaků	. 37
Obrázek 36 - Code 128 pro testování – méně znaků	. 38
Obrázek 37 - Výsledek Code 128 pro testování – méně znaků	. 38
Obrázek 38 - Interleaved 2 of 5 pro testování – správný počet znaků	. 39
Obrázek 39 - Výsledek Interleaved 2 of 5	. 39
Obrázek 40 - Interleaved 2 of 5 pro testování – více znaků	. 40
Obrázek 41 - Výsledek Interleaved 2 of 5 pro testování – více znaků	. 40
Obrázek 42 - Interleaved 2 of 5 pro testování – méně znaků	. 41

Obrázek 43 - Výsledek Interleaved 2 of 5 pro testování – méně znaků	41
Obrázek 44 - EAN-8 pro testování	42
Obrázek 45 - Codabar pro testování	42
Obrázek 46 - Výsledek EAN-8 + Codabar	43

# Seznam tabulek

Tabulka 1 - Code 128 - kódovací tabulka 1	8
Tabulka 2 - pomocná tabulka pro výpočet kontrolní číslice 2	1
Tabulka 3 - Interleaved 2 of 5 - kódovací tabulka 2	2
Tabulka 4 - Interleaved 2 of 5 - kódovací tabulka 2	3
Tabulka 5 - Výpočet kontrolní číslice 2	4

# Seznam zkratek

API	Aplication Programming Interface
ATL	Automatická třídící linka
PPL	Professional Parcel Logistic
ICT	Information and Communications Technology
DHL	Dalsey, Hillblom a Lynn – jména zakladatelů
IDE	Integrated Development Enviroment
JDK	Java Development Kit
SDK	Software Development Kit
JVM	Java Virtual Machine
UPC	Universal Product Code
ITF	Interleaved 2 of 5
iOS	iPhone Operation System
PC	Personal Computer
JIT	Just-in-Time

# Úvod

Tato bakalářská práce se zabývá vytvořením mobilní aplikace pro operační systém Android, která bude vytvořena pro společnost PPL CZ s.r.o. (dále jen "PPL"). Vytvořená aplikace bude využívána pro ověřování správnosti použitých čárových kódů na etiketách, které si zákazníci této společnosti vytvářejí sami v rámci implementace softwarového napojení na API.

Téma jsem si vybral z jednoho prostého důvodu, a to, protože v mém týmu IT regionální podpory společnosti PPL, máme na starost softwarové napojení zákazníka na PPL a vyvíjená aplikace nám zjednoduší proces schvalování etiketních štítků. V současné době zákazník společnosti PPL, který chce využívat přímé API napojení s vlastním tiskem etiket, musí vyřešit kromě datového napojení, také správný vzhled etikety a s tím spojené správné používání čárových kódů, které jsou specifikovány v dokumentaci. Překládané etikety ke schválení jsou ověřovány pomocí různých online generátorů čárových kódů dle uvážení ICT technika, který je zodpovědný za danou implementaci. Porovnávání dodané etikety ke schválení s vygenerovaným kódem probíhá vizuálně oproti online generátoru čárových kódů a pro nás jsou důležité pouze dva, a to ještě s omezeným počtem znaků. Etiketa je také následně po vytisknutí ještě ověřena na firemních skenerech a na ATL. Vizuální srovnávání čárových kódů není úplně uživatelsky ideální, a může dojít k lidské chybě, a proto by varianta validace etikety pomocí mobilní aplikace byla daleko uživatelsky přívětivější, a hlavně i přesnější než pouhé vizuální porovnávání.

Stanovený cíl práce je tedy v podstatě jasný, vytvořit jednoduchou mobilní aplikaci, která pozná a vyhodnotí dva typy čárových kódů včetně správné délky snímaného čárového kódu. Jedná se o čárový kód zásilky, který se používá Interleaved 2 of 5 with checksum s celkem 12 znaky a potom směrový neboli routovací kód typu Code 128 s celkovou délkou 17 znaků.

Pro vytvoření mobilní aplikace bylo zvoleno Android studio aktuálně ve verzi 4.1.2 s využitím programovacího jazyka Java. Aplikace bude vytvořena jako více okenní a její funkčnost bude popsána a ukázána na následujících stránkách. Stěžejní součástí aplikace bude snímání čárového kódů fotoaparátem mobilního telefonu s následným zpracováním a vyhodnocením, zda je z pohledu společnosti PPL čárový kód v pořádku.

# 1 Mobilní aplikace

Mobilní aplikace je typ softwaru, který je naprogramovaný pro použití na chytrých telefonech a tabletech. V současné chvíli jsou mobilní aplikace vyvíjeny převážně pouze na dvě mobilní platformy, a to Android a iOS. Mobilní aplikace často poskytují podobné služby jako programy, které jsou dostupné na PC. Většinou jsou to, ale malé aplikace s omezenými funkcemi, kdy často každá aplikace má svoji specifickou funkčnost jako například předpověď počasí, kalkulačka a mnoho jiných. Mobilní aplikace jsou ještě děleny do dvou kategorií, a to nativní aplikace a webová aplikace. Nativní aplikace jsou programovány přímo pro konkrétní platformu, nabízejí rychlejší výkon, lepší uživatelské rozhraní, ale jsou náročnější na vývoj a zajištění kvality. Webové aplikace jsou spouštěny přímo prostřednictvím webového prohlížeče, všechny informace jsou v databázi na serveru a k tomu je potřeba stabilní internetové připojení. (Techopedia, 2020)

### 1.1 Android Studio

Android studio je oficiální IDE založené na IntelliJ IDEA pro vývoj mobilních aplikací pro platformu Android. Bylo představeno v květnu 2013 na Google I/O konferenci a vyvinuto společně s firmou JetBrains, která má nejpokročilejší IDE pro Javu, do kterého se přidali funkce speciálně upravené pro vývoj aplikací na platformě Android. Android Studio obsahuje výkonný editor kódu, který je založen na IntelliJ IDEA a podporuje funkce jako inteligentní úpravy kódu, pokročilé refaktorování nebo hloubkou analýzu kódu. (Xavier Ducrohet, 2013)

Android studio je dostupné k používání zdarma a ke stažení pro Windows, macOS a pro operační systémy založené na Linuxu. (Anon., nedatováno)

### 1.2 Java

Java je objektově orientovaný programovací jazyk, který byl vyvinut v roce 1995 firmou Sun Microsystems. V roce 1996 byl představen první JDK, ve kterém bylo vše, co bylo potřeba ke tvorbě appletů. V současnosti SDK obsahuje kompletní zdrojový kód platformy Java. Jazyk vychází z jazyka C++, má však jednodušší syntaxi. Veliká výhoda Javy je přenositelnost, protože Java je interpretovaný jazyk a zdrojový kód se při kompilaci přeloží do mezikódu, který pak lze spustit na kdekoliv kde je nainstalovaný virtuální stroj JVM. Nevýhodou tohoto řešení je, že programy v Javě jsou pomalejší než programy, které jsou rovnou přeloženy do strojového kódu. Tuto nevýhodu částečně řeší JIT, kdy se mezikód překládá do strojového kódu za běhu programu. Tím se zrychluje běh aplikace. (Faltýnek, 2007)

# 2 Čárové kódy

Čárové kódy se skládají z určitého počtu čar a mezer. Před a za čárovým kódem musí být klidová zóna bez jakéhokoli potisku. Čarový kód začíná znakem start, potom následuje vlastní kód s případným kontrolním součtem a končí znakem stop. Počet čar i mezer a jejich počet je dán specifikací daného typu čárového kódu. (Gaben, nedatováno) Čárové kódy jsou v dnešní době nejrozšířenější prostředek automatické identifikace. Jejich výhoda je přesnost, jelikož při ručním zadávání dochází k daleko větší chybovosti než při použití čárových kódů, zvláště pokud je v čárovém kódu také kontrolní číslice, která ověřuje správnost přečtení ostatních číslic. Rychlost pořízení dat pomocí čárových kódů je nesrovnatelně rychlejší než ruční zadávání. Nezanedbatelné je také zvýšení produktivity práce, například v maloobchodě po zavedení čárových kódů se rychlost odbavování u pokladen zvýší a několik desítek procent. Laserové skenery čárového kódu vyzařují červené světlo, které je pohlcováno tmavými čárami, a naopak odráženo mezerami. Skener na základě rozdílů v reflexi přeměňuje šířky čar a mezer na elektrické signály, které jsou následně převedeny na číslice, písmena a případně speciální znaky, které může daný typ čárového kódů obsahovat. Aktuálně se pro snímání čárového kódu kromě laserových skenerů využívají i digitální skenery. Tyto skenery stejně jako u digitálního fotoaparátu vyfotí čárový kód a následně dojde k dekódování jeho obsahu pomocí dekodéru, který je součástí skeneru. (Kodys, nedatováno)

### 2.1 Historie čárových kódů

Vynálezci čárového kódu dle amerického patentu číslo US2612994A, patent popisující lineární a kružnicový čárový kód, podaného u patentového úřadu v říjnu 1949 a schváleném z října roku 1952 jsou pánové Bernad Silver a Norman Joseph Woodland. (Anon., nedatováno) Myšlenka z roku 1932, která předcházela čárovým kódům a to zavedení systému děrovacím karet pro sledování zboží v supermarketech a obchodech s potravinami, kterou navrhl Walace Flint, se pro svoji těžkopádnost a finanční náročnost nikdy neuskutečnila. V roce udělení patentu sestrojil Silver a Woodland první čtečku čárových kódů. V roce 1962 Philco kupuje patent, který později prodává společnosti RCA. 18měsíční test kruhových čárových kódů probíhal od roku 1972 v obchodě Kroger v Cincinnati. O rok později byl zaveden standard UPC pro označování produktů. V roce 1974 byl prodaný balíček žvýkaček Wrigley´s pomocí čárového kódu a stal se tak prvním prodaným zbožím pomocí skeneru čárových kódů. (Barcoding, nedatováno)

### 2.2 Přehled nejpoužívanějších čárových kódů

V této části je uveden seznam nejpoužívanějších čárových kódů včetně stručného popisu a vizuálního příkladu.

### 2.2.1 Code 39

Code 39 je čárový kód používaný především v automobilovém průmyslu, zdravotnictví a dalších odvětví průmyslu a obchodu, který umožňuje zakódovat číslice, písmena i některé alfanumerické znaky.



**Obrázek 1 - Code 39** (vlastní zpracování)

### 2.2.2 UPC-A

UPC-A čárový kód se používá především v USA k zakódování 12místného čísla, poslední číslice je kontrolní znak.



**Obrázek 2 - UPC-A** (vlastní zpracování)

### 2.2.3 UPC-E0

UPC-EO – je variantou čárového kódu UPC-A s potlačením nul, první znak je vždy nula a ostatní znaky mají stejný význam jako u UPC-A.



**Obrázek 3 - UPC E0** (vlastní zpracování)

#### 2.2.4 EAN-8 a EAN-13

EAN-8 a EAN-13 – je nadstavbou amerického UPC, obě verze jsou numerické a liší se počet kódovaných číslic, kde EAN-8 kóduje 8 znaků a EAN-13 kóduje 13 znaků.



**Obrázek 4 – EAN-8** (vlastní zpracování)



**Obrázek 5 – EAN-13** (vlastní zpracování)

#### 2.2.5 Code 93

Code 93 kóduje všech 128 ASCII znaků, jedná se o alfanumerickou symboliku proměnné délky.



**Obrázek 6 - Code 93** (vlastní zpracování)

2.2.6 Interleaved 2 of 5 a Interleaved 2 of 5 Mod 10

Podrobněji bude popsáno níže.



**Obrázek 7 - Interleaved 2/5** (vlastní zpracování)



Obrázek 8 - Interleaved 2/5 Mod 10 (vlastní zpracování)

2.2.7 Code 128

Podrobněji bude popsáno níže.



**Obrázek 9 - Code 128** (vlastní zpracování)

### 2.2.8 Codabar

Codabar je diskrétní samo opravný kód proměnné délky, zahrnuje celkem 16 znaků, číslice 0–9 a sady speciálních znaků.



20210204

**Obrázek 10 – Codabar** (vlastní zpracování)

2.2.9 MSI

MSI – je numerická symbolika proměnné délky, nejčastěji se používá na cenových etiketách v maloobchodě.



**Obrázek 11 - MSI** (vlastní zpracování)

(Gaben, nedatováno)

### 2.3 Code 128

Tento čárový kód byl vyvinut v roce 1981, je to alfanumerická symbolika s proměnnou délkou a využívá se především v logistice. Každý znak se skládá ze tří čar a tří mezer a to tak, že každý znak má šířku 11modulů, kromě znaku stop, který má 13 modulů. Čáry a mezery mohou mít jeden až čtyřnásobek základní šířky. (Gaben, nedatováno)

Je schopný zakódovat spodní ASCII tabulku, tedy 128 znaků s možností rozlišit malá a velká písmena. Má tři znakové sady, které se na začátku kódu nastaví speciálním znakem. (Codeware, nedatováno)

Specifikace čárového kódu Code 128, obsahuje níže uvedené částí:

- 1. klidová zóna
- 2. start znak
- 3. kódované znaky
- 4. kontrolní součet
- 5. stop znak
- 6. klidová zóna

(Anon., nedatováno)

Kontrolní číslice se vypočítává jako vážený součet hodnoty kódu (start A/B/C) a hodnoty každého znaku, který je vynásobený jeho pozicí v kódu. První symbol a start znak mají pozici 1. Celkový součet se poté vydělí modulo 103 a tento zbytek je poté hodnota kontrolní číslice, které se zapíše do čárového kód před stop znak. (EasierSoft, nedatováno)

Value	Table A	Table B	Table C	ASCII Code	Character	Pattern		
0	Space	Space	0	0032 or 0212	Space or O	11011001100		
1	!	!	1	33	!	11001101100		
2	"		2	34 "		11001100110		
3	#	#	3	35 #		10010011000		
4	\$	\$	4	36 \$		10010001100		
5	%	%	5	37 %		10001001100		
6	&	&	6	38 &		10011001000		
7	I	1	7	39 '		10011000100		
8	(	(	8	40	40 (			
9	)	)	9	41	)	11001001000		
10	*	*	10	42 *		11001000100		
11	+	+	11	43	43 +			
12	,	,	12	44 ,		10110011100		
13	-	-	13	45	-	10011011100		
14	•		14	46		10011001110		
15	/	/	15	47	/	10111001100		

Tabulka 1 - Code 128 - kódovací tabulka

16	0	0	16	48	48 0 10011			
17	1	1	17	48         0         1001110           49         1         100110           50         2         110010           51         3         110010           52         4         110010           53         5         110110           54         6         110010           55         7         110100           56         8         110010           57         9         110010           58         :         110010           60         <         110010           61         =         110010           62         >         110110           63         ?         110110           64         @         110011           65         A         101001           66         B         100110           67         C         100110           67         C         100110           70         F         100110           71         G         110100           73         I         100010           74         J         101100           75         K         101100				
18	2	2	18	50	2	11001110010		
19	3	3	19	51	3	11001011100		
20	4	4	20	52	4	11001001110		
21	5	5	21	53	5	11011100100		
22	6	6	22	54	6	11001110100		
23	7	7	23	55	7	11101101110		
24	8	8	24	56	8	11101001100		
25	9	9	25	57	9	11100101100		
26	:	:	26	58	:	11100100110		
27	;	;	27	59	;	11101100100		
28	<	<	28	60	<	11100110100		
29	=	=	29	61	=	11100110010		
30	>	>	30	62	>	11011011000		
31	?	?	31	63	?	11011000110		
32	@	@	32	64	@	11000110110		
33	A	А	33	65	А	10100011000		
34	В	В	34	66	В	10001011000		
35	С	С	35	67	С	10001000110		
36	D	D	36	68	D	10110001000		
37	E	E	37	69	E	10001101000		
38	F	F	38	70	F	10001100010		
39	G	G	39	71	G	11010001000		
40	Н	Н	40	72	Н	11000101000		
41	I	I	41	73	I	11000100010		
42	J	J	42	74	J	10110111000		
43	К	К	43	75	K	10110001110		
44	L	L	44	76	L	10001101110		
45	М	М	45	77	М	10111011000		
46	N	N	46	78	N	10111000110		
47	0	0	47	79	0	10001110110		
48	Р	Р	48	80	Р	11101110110		
49	Q	Q	49	81	Q	11010001110		
50	R	R	50	82	R	11000101110		
51	S	S	51	83	S	11011101000		
52	Т	Т	52	84	Т	11011100010		
53	U	U	53	85	U	11011101110		
54	V	V	54	86	V	11101011000		
55	W	W	55	87	W	11101000110		
56	Х	Х	56	88	Х	11100010110		
57	Y	Y	57	89	Y	11101101000		
58	Z	Z	58	90	Z	11101100010		
59	[	[	59	91	[	11100011010		
60	١	\	60	92	\	11101111010		

61	]	]	61	93 ] 110010		11001000010				
62	۸	۸	62	94	۸	11110001010				
63	_	_	63	95	_	10100110000				
64	nul	``	64	96	×	10100001100				
65	soh	а	65	97	а	10010110000				
66	stx	b	66	98	b	10010000110				
67	etx	с	67	99	С	10000101100				
68	eot	d	68	100	d	10000100110				
69	eno	е	69	101	е	10110010000				
70	ack	f	70	102	f	10110000100				
71	bel	g	71	103	g	10011010000				
72	bs	h	72	104	h	10011000010				
73	ht	i	73	105	i	10000110100				
74	lf	j	74	106	j	10000110010				
75	vt	k	75	107	k	11000010010				
76	ff	I	76	108	I	11001010000				
77	cr	m	77	109	m	11110111010				
78	sO	n	78	110	n	11000010100				
79	s1	0	79	111	0	10001111010				
80	dle	р	80	112	р	10100111100				
81	dc1	q	81	113	q	10010111100				
82	dc2	r	82	114	r	10010011110				
83	dc3	S	83	115	S	10111100100				
84	dc4	t	84	116	t	10011110100				
85	nak	u	85	117	u	10011110010				
86	syn	v	86	118	v	11110100100				
87	etb	w	87	119	w	11110010100				
88	can	х	88	120	х	10100001100           10010000110           10000101100           10110000010           101100000           10110000100           10011000010           10011000010           10000110100           10000110100           10000110000           1000011000           1000011000           1000011000           1100001000           1100001000           11000010100           1000111100           1001011100           10010111100           10010111100           100111100100           1101011100           1101011100           1101011100           1101011100           1111001000           1111001000           1111010100           1010111100           1010111100           1010111100           1011110000           1011110000           1011110000           1011110000           1011110000           1011110000           1011110000           1011110000           10111100000           10111100000           101111000000      <				
89	em	У	89	121	У	a         10010110000           b         1001001110           c         10000101100           d         1011001000           e         10110000100           f         1001101000           g         1001101000           g         1000110100           g         1000011000           k         1000010010           k         1100001000           k         1100001000           k         1100001000           k         1100001000           k         1100001000           n         11000010100           n         11000010100           p         1001011100           q         1001011100           q         1001011100           q         10011110010           v         1110101010           k         110101110           v         1111010100           x         110101110           x         100111100           x         1001011110           x         1010101110           x         1010101110           x         101011100           x         101011110				
90	sub	Z	90	122	b         1001000110           c         10000101100           d         10000100100           e         101100000           f         101100000           g         1001101000           j         1000011010           k         10000110010           k         1000010010           k         11000010010           k         110001000           m         111101100           n         1000111100           n         10000111100           p         10010111100           n         10010111100           n         10010111100           q         10010111100           q         10010111100           q         1001111000           v         111100100           v         111100100           v         111100100           v         111101010           v         111101010           v         1010011110           v         1010011110           k         101011100           k         1011101010           k         1011110000      k         1011110000					
91	esc	{	91	123	{	10110000100           100110100010           10000110100           10000110100           1100001000           1100001000           1100001000           1100001000           1100001000           1100001000           1100001000           1100001000           1000111100           1001011100           1001011100           1001011100           100111100100           100111100100           1111010100           1111010100           1111010100           1111010100           1111010100           101011110           101011110           101011110           1010011110           1011110000           1011110000           1011110000           1011110000           1011110000           1011110000           1011110000           1011110000           10111100000           10111100000           11101001110           11101001110           11101000100           1101000100           11010000000           11010000000				
92	fs		92	124	g1001101000i1000011010i10000110100k11000010010k110001000m1111011100m1110011100o1000111100o1000111100p10100111100q10010111100s1011110100v1010111100v1010111100v1010111100v111001010v111001010v1111010100v1111010100v1111010100v1111010100v1111010110v1111010110izi1010011110jii1010011110jiiiiij					
93	gs	}	93	125	}	10100011110				
94	rs	~	94	126	~	10001011110				
95	us	del	95	200	E	10111101000				
96	Fnc 3	Fnc 3	96	201	E	10111100010				
97	Fnc 2	Fnc2	97	202	E	11110101000				
98	Shift	Shift	98	203	E	11110100010				
99	Code C	Code C	99	204	I	10111011110				
100	Code B	Fnc 4	Code B	205	I	10111101110				
101	Fnc 4	Code A	Code A	206	1	11101011110				
102	Fnc 1	Fnc 1	Fnc 1	207	I	11110101110				
103	Start A	Start A	Start A	208	D	11010000100				
104	Start B	Start B	Start B	209	N	11010010000				
105	Start C	Start C	Start C	210	0	11010011100				

106	Stop	Stop	Stop	211	0	1100011101011
			(vlas	tní zpracování)		

(Anon., nedatováno)

### 2.3.1 Code 128 A

Code 128 A umožňuje kódovat všechny standardní velké alfanumerické znaky A–Z, 0–9, speciální a funkční znaky. (Softmatic, nedatováno)

#### 2.3.2 Code 128 B

Code 128 B umožňuje kódovat všechny alfanumerické znaky A–Z, a–z, 0–9 speciální a funkční znaky. (Softmatic, nedatováno)



20210204

**Obrázek 12 - Code 128 B** (vlastní zpracování)



Obrázek 13 - Code 128 B – struktura kódu (vlastní zpracování)

Příklad výpočtu kontrolní číslice u výše uvedeného čárového kódu 20210204, který je níže přepsán do pomocné tabulky.

pozice	1	1	2	3	4	5	6	7	8
znak	start B	2	0	2	1	0	2	0	4
hodnota	104	18	16	18	17	16	18	16	20

Гabulka 2 -	pomocná	tabulka	pro v	ýpočet	kontrolní	číslice

(vlastní zpracování)

Kontrolní číslice se vypočítává jako vážený součet hodnot kódovaných znaků včetně start znaku, start znak a první pozice kódu jsou v pozici 1.

V případě výpočtu kontrolní číslice výše uvedeného čárového kódu bude výpočet vypadat takto:

znak	hodnota	pozice	hodnota x pozice		
Start B	104	1	104		
2	18	1	18		
0	16	2	32		
2	18	3	54		
1	17	4	68		
0	16	5	80		
2	18	6	108		
0	16	7	112		
4	20	8	160		
	součet				
	zbyte	k po modulo 103	15		

Tabulka 3 - Interleaved 2 of 5 - kódovací tabulka

(vlastní zpracování)

Kontrolní číslice u kódu 20210204 má hodnotu 15 tedy dle tabulky 1 bude 10111001100 tzn. nejužší čára, nejužší mezera, trojitá čára, dvojitá mezera, dvojitá čára a konec je dvojitá mezera.



**Obrázek 14 - přepis kontrolní číslice do čárového kódu** (vlastní zpracování)

### 2.3.3 Code 128 C

Code 128 C je pouze číselný čárový kód obsahující číslice 0–9, kóduje dvě číslice najednou, velmi úsporný kód. (Softmatic, nedatováno)

### 2.4 Interleaved 2 of 5 with modulo 10 checksum

Tento čárový kód je numerický samo opravný kód, používaný především pro značení přepravních obalů v průmyslových a maloobchodních aplikacích. Párují se vždy dva znaky dohromady, první

znak do pěti čar a druhý znak do pěti mezer mezi čarami prvního znaku tzn., že všechny znaky na lichých pozicích jsou kódovány jako čáry a všechny znaky na sudých pozicích jako mezery. Dvě z pěti čar a stejně tak dvě z pěti mezer jsou široké, odtud pochází jméno kódu (2z5)

Struktura čárového kódu.

- 1. Start znak
- 2. Kódované znaky
- 3. Kontrolní číslice
- 4. Stop znak

Start znak obsahuje vždy dvě tenké čáry a dvě tenké mezery

(Gaben, nedatováno)



#### Obrázek 15 - Interleaved 2 of 5 - start znak (vlastní zpracování)

Stop znak obsahuje širokou čáru, úzkou mezeru a úzkou čáru)



#### Obrázek 16 - Interleaved 2 of 5 - stop znak

(vlastní zpracování)

Čísla jsou kódována pomocí níže uvedené tabulky, kde "n" znamená úzkou čáru nebo mezeru a "W" znamená širokou čáru nebo mezeru.

Číslo	Šírka čáry nebo mezery				
0	n	n	W	W	n
1	W	n	n	n	W
2	n	W	n	n	W
3	W	W	n	n	n
4	n	n	W	n	W
5	W	n	W	n	n

#### Tabulka 4 - Interleaved 2 of 5 - kódovací tabulka

6	n	W	W	n	n
7	n	n	n	W	W
8	W	n	n	W	n
9	n	W	n	W	n
(vlastní zpracování)					

Kontrolní číslice se vypočítává jako součet všech číslic na lichých pozicích, výsledek se vynásobí třemi a k tomu se přičte součet všech sudých číslic. Kolik zbývá do další desítky tolik se rovná kontrolní číslice. (IDAutomation, nedatováno)

Příklad výpočtu kontrolní číslice u kódu 80050411557.

rabulka 5 - vypočet kontrolní číslice							
80050411557							Součet
Liché pozice	8	0	0	1	5	7	21
Sudé pozice	0	5	4	1	5		15

Tabulka 5 - Výpočet kontrolní číslice

(vlastní zpracování)

#### Součet lichých pozic \*3 = 21 \* 3 = 63

*Přičteme součet sudých pozic* = 63 + 15 = 78

Do nejbližší desítky je potřeba přičíst číslo 2 a to je výsledná kontrolní číslice ke kódovanému číslu 80050411557, na obrázku níže je kód i s kontrolní číslicí.



Obrázek 17 - Interleaved 2 of 5 with modulo 10 checksum (vlastní zpracování)

### 3 O PPL CZ s.r.o.

Společnost PPL patří mezi nejvýznamnější poskytovatele logistických služeb v České republice, vznikla v roce 2004 sloučením sedmi navzájem spolupracujícími společnostmi založenými roku 1995. Po převzetí zahraniční skupinou Deutsche Post DHL v roce 2006 se portfolio služeb rozšířilo o zasílání zásilek do zahraničí. Aktuálně je PPL součástí divize DHL eCommerce Solution. Ke konci roku 2020 měla PPL CZ s.r.o. celkem 21 balíkových dep a centrální překladiště. (PPL, nedatováno)

### 3.1 Etiketa PPL

Etiketa PPL obsahuje kromě adresy příjemce a odesílatele také jeden až dva čárové kódy podle toho, kterou verzi etikety zákazník využívá. Buď starší typ etikety pouze s jedním kódem anebo aktuálně platný standard nazývaný "smart label" se dvěma čárovými kódy.

### 3.1.1 Standardní etiketa

Do května roku 2020 byl v platnosti vzhled etikety, který obsahovala pouze jediný čárový kód, a to byl Interleaved 2 of 5 with modulo 10 checksum, kterým se na etiketě zapisovalo číslo zásilky. Čísla zásilek u PPL jsou 11místná, ale v tomto čárovém kódu se používá také kontrolní číslice, která se tiskne v čárovém kódu, ale ne v číselné podobě pod čárovým kódem.

PROFESSIONAL PARCEL LOGISTIC profesionální baliková služba pevna linka : 225 331 500 https://www.ppl.cz info@ppl.cz	Private parcel name ( Contact - (mandat.) Street + number Olomouc	(Evening deli) - opt
PPL Parcel CZ Private	77200 Tel.:+420123456789	Vecer
)295509	Consignor/Odesilatel: Ulice spatna Sternberk 78501 Pozn.:Note to be printed on the labe Ref.:Customer_reference(CUST)	· [1/1

Obrázek 18 - standardní etiketa

(vlastní zpracování)

### 3.1.2 Smart Label

Od května roku 2020 vstoupil v platnosti nový vzhled etikety označovaný jako Smart Label, který kromě výše uvedeného jednoho čárového kódu zásilky obsahuje další textová pole, a navíc čárový kód typu Code 128 ve kterém je na etiketě uvedeno číslo zásilky spolu s číslem výstupního (doručujícího) depa a trasou doručujícího řidiče.

Tento nový typ etikety zrychlil proces zpracování zásilky na depu PPL tím, že odstranil potřebu manuálního lepení tohoto kódu na zásilku po zvážení zásilky. Tento 17místný kód obsahuje 11místné číslo zásilky, pomlčku a 5místné číslo depa s trasou doručujícího řidiče např. 80050411557-12999, kde 80050411557 je číslo dané zásilky, 12 je číslo výstupního depa a 999 je trasa doručujícího řidiče.



Obrázek 19 - Smart Label etiketa (vlastní zpracování)

## 4 Vývoj aplikace

Tato část se věnuje tvorbě mobilní aplikace pro ověřování čárových kódů.

### 4.1 Požadavky na funkčnost

Od začátku byly na aplikaci kladeny v podstatě pouze dva požadavky, a to čtení dvou typů čárových kódu (code 128 a Interleaved 2 of 5) s následným ověřením, zda naskenované čárové kódy mají také správný počet znaků pomocí mobilní aplikace pro Android.

### 4.2 Realizace aplikace

Jak již bylo uvedeno výše, tak pro vytvoření mobilní aplikace bylo použito Android Studio ve verzi 4.1.2 a programovací jazyk Java. (ITnetwork, nedatováno)

Aplikace je tvořena jako více okenní a skládá se z několika obrazovek neboli "Aktivit". Jako nejdůležitější součást aplikace je určitě skenovací aktivita, díky které je možné načíst jakýkoli čárový kód, který následně aplikace vyhodnotí podle zadaných kritérií, zda je pro PPL v pořádku či nikoliv.

Po několika slepých uličkách v podobě hledání nějakého funkčního kódu, který by umožnil používání vestavěného fotoaparátu pro skenování čárových kódů. A protože není potřeba vynalézat již vynalezenou věc, tak pro potřeby této aplikace bylo nakonec použito pro skenování čárových kódu předpřipravené řešení na GitHubu, což je v podstatě bezplatné úložiště pro opensource projekty. V aplikaci byly použity části kódu od David Rodilla (David, 2019) a od Ravi Tamada. (Ravi, 2017), které byly dále upraveny a přizpůsobeny na míru požadavkům na funkčnost aplikace.

### 4.2.1 Úvodní obrazovka

Pro úvodní aktivitu před načtením hlavní obrazovky bylo zvoleno animování firemního loga PPL na úvodní obrazovce spolu s logem školy. Animace se aplikuje na ImageView což je logo společnosti PPL.

// Nastavime a spustime animaci pro firemni logo
ImageView imgvwSpsoa = findViewById(R.id.logo\_ppl);
Animation animSpsoaLogo = AnimationUtils.loadAnimation( context: this, R.anim.welcome\_sc);
// animuje se imageView, což je ID firemniho loga.

imgvwSpsoa.startAnimation(animSpsoaLogo); // spuštění samotné animace

Obrázek 20 - Welcome – nastavení a spuštění animace (vlastní zpracování) Po spuštění výše uvedené animace je také spuštěn časovač, díky kterému je animace vidět. Jinak by se ihned načetla hlavní aktivita bez animace. V ostré verzi aplikaci je nastaven časovač na 5000ms

```
// Vytvoření druhého vlákna s časovačem, po uplynutí limitu je spuštěna další aktivita
Thread welcomeScreenDelay = new Thread() {
    public void run() {
        try {
            sleep( millis: 5000); // po ukončení časovače se spustí samotná aplikace, ostrá verze 5000
            startActivity(new Intent(getApplicationContext(), MainActivity.class));
        } catch (InterruptedException e) {
            e.printStackTrace();
        } finally {
            finish();
        }
    }
};
```

#### Obrázek 21 - Welcome – odložení startu aplikace (vlastní zpracování)

(Anon., nedatováno)

#### 4.2.2 Skenovací obrazovka

Po úspěšném naskenování čárového kódu, oznámeno zvukovým signálem, se pomocí Intentu přenesou proměnné "typeBC" a "valueBC" do hlavní aktivity, kde je s nimi dále pracováno pro vyhodnocení čárového kódu.

```
// pipnuti po načteni kódu
barcodeReader.playBeep();
Intent result = new Intent();
//uloženi potřebných dat k přenosu do MainActivity
result.putExtra( name: "typeBC",(String.valueOf(barcode.format))); //vrací číslo typu barcodu
result.putExtra( name: "valueBC",(barcode.displayValue)); // vrací hodnotu barcodu
setResult(Activity.RESULT_OK, result);
finish();
```

Obrázek 22 - BarcodeActivity – uložení hodnot (vlastní zpracování)

(Anon., 2013)

#### 4.2.3 Hlavní obrazovka

Po přenesení hodnot v proměnných pomocí Intentu ze skenovací obrazovky do hlavní obrazovky jsou učiněny níže uvedené kroky k vyhodnocení správnosti čárového kódu.

Pro potřeby vyhodnocení čárového kódu jsou vytvořeny dvě proměnné z přenesených hodnot ze skenovací obrazovky. A to typeBC ve které je uložen typ čárového kódu a valueBC ve které je uložena naskenovaná hodnota čárového kódu.

```
// uložení dat z BarcodeActivity
String typeBC = data.getStringExtra( name: "typeBC");
String valueBC = data.getStringExtra( name: "valueBC");
```

Obrázek 23 - MainActivity – uložení dat z BarcodeActivity (vlastní zpracování)

Aplikace je určena pro ověření čárových kódu Code 128 a Interleaved 2 of 5 with checksum a další typy čárových kódu ignoruje, resp. Vyhodnocuje jako "Nesprávný typ čárového kódu".

Správné typy čárových kódů jsou ověřovány pomocí proměnné typeBC, kde jsou v textové podobně uložené konstanty odpovídající typu čárového kódu:

- Code 128 = "1"
- Interleaved 2 of 5 with checksum = "128"

(Google.com, 2021)

Zároveň s typem čárového kódu je ihned ověřena i délka samotného kódu, zda odpovídá typu čárového kódu. Pro čárový kód Code 128 je to 17 znaků a v těchto 17 znacích je obsaženo 11místné číslo zásilky, pomlčka a routovací kód složený s čísla depa a směru doručujících řidiče.

U čárového kódu typu Interleaved 2 of 5 with checksum je to 12 znaků mezi kterými je 11místné číslo zásilky a kontrolní číslice.

```
/*** kontrola na typ BarCode a počet znaků
 * Code128 == 1
 * ITF (Interleaved 2 of 5) = 128
*/
if ((typeBC.contentEquals( cs: "1")) && (valueBC.length() == 17)) {
    scanResult.setText("Čárový kód typu: Code 128\n\nHodnota: " + valueBC);
} else if ((typeBC.contentEquals( cs: "1")) && (valueBC.length() != 17)) {
    scanResult.setText("Čárový kód typu: Code 128 NEPLATNÉ délky\n\nHodnota: "
            + valueBC + "\n\nVýsledek: " + pocetZnakuBC(typeBC, valueBC));
} else if ((typeBC.contentEquals( cs: "128")) && (valueBC.length() == 12)) {
    scanResult.setText("Čárový kód typu: ITF\n\nHodnota: " + valueBC);
} else if ((typeBC.contentEquals( cs: "128")) && (valueBC.length() != 12)) {
    scanResult.setText("Čárový kód typu: ITF NEPLATNÉ délky\n\nHodnota: "
            + valueBC + "\n\nVýsledek: " + pocetZnakuBC(typeBC, valueBC));
} else {
   scanResult.setText("Nesprávný typ čárového kódu");}
```

#### Obrázek 244 - MainActivity – kontrola typu a délky čárového kódu (vlastní zpracování)

Pro vylepšení vyhodnocení čárového kódu byla přidána funkce "pocetZnaku", která při správném typu čárového kódu, ale nesprávném počtu znaků vyhodnotí, kolik znaků, ve kterém čárovém kódu chybí či naopak přebývá.

Toto funkcionalita nebyla uvedena ve funkčních požadavcích na aplikaci, ale je díky ní je na první pohled patrné kolik znaků chybí či je navíc ve správném čárovém kódu.

```
public String pocetZnakuBC(String typ, String kod){
   String result = "";
   int delkaBC = kod.length();
   int ITF = 12;
   int code128 = 17;
   switch (typ) {
        case "1": // když code128
           if (delkaBC < code128) {</pre>
               //result = "chybi " + (code128 - delkaBC) + " znaků";
               result = "chybi " + (code128 - delkaBC) + "" + (((code128 - delkaBC)<5) ? " znak(y)" : " znaků");</pre>
           } else if (delkaBC > code128) {
               //result = "přebývá " + (delkaBC - code128) + " znaků";
               result = "navic " + (delkaBC - code128) + "" + (((delkaBC - code128)<5) ? " znak(y)" : " znaků");</pre>
            3
           break;
        case "128": // když Interleaved 2 of 5
           if (delkaBC < ITF) {</pre>
               //result = "chybi " + (ITF - delkaBC) + " znaků";
               result = "chybi " + (ITF - delkaBC) + "" + (((ITF - delkaBC)<5) ? " znak(y)" : " znaků");</pre>
           } else if (delkaBC > ITF) {
               //result = "přebývá " + (delkaBC - ITF) + " znaků";
               result = "navic " + (delkaBC - ITF) + "" + (((delkaBC - ITF)<5) ? " znak(y)" : " znaků");</pre>
            }
           break;
   3
   return result;
3
```



### 4.3 Ukázky běžící aplikace

V následujících několika bodech je představena hotová aplikace na ověřování čárových kódů vytvořená na míru společnosti PPL.

### 4.3.1 Spouštění aplikace

Na následujících třech obrázcích je "vidět" úvodní animace loga PPL.

Začátek animace firemního loga PPL.



Obrázek 26 - Ukázka aplikace – start (vlastní zpracování)

Další fáze animace firemního loga PPL.



Obrázek 27 - Ukázka aplikace – start (vlastní zpracování)

Konečná pozice firemního loga PPL.



Obrázek 28 - Ukázka aplikace – start (vlastní zpracování)

### 4.3.2 Hlavní obrazovka aplikace

Na hlavní obrazovce nalezneme několik funkčních tlačítek na ovládání aplikace a uprostřed dominantní textové pole zobrazující výsledky po naskenování čárového kódu, ve kterém po spuštění aplikace je zobrazena hláška "Nic k zobrazení".



Obrázek 29 - Ukázka aplikace – Hlavní obrazovka (vlastní zpracování)

### 4.3.3 Nastavení

Nastavení je zatím pouze předpřipraveno na budoucí využití a aktuálně zobrazuje hlášku "under construction". V některém s dalších releasů aplikace je počítáno s možností volby light/dark módu aplikace jež je teď tak moderní např. z důvodu úspory energie. Případně je zde možno přidat další nastavení, které vyplynou z požadavků společnosti PPL při používání aplikace.

Na	stavení
under c	onstruction
	ZPĚT
Q	0 🗆

Obrázek 30 - Ukázka aplikace – Nastavení (vlastní zpracování)

### 4.3.4 Info

Zde je informace o aktuální verzi aplikace, ze které lze rozklíčovat datum uvolnění nové verze aplikace.

Příklad verze 0.8.20210717.05, kde číslo 20210717 značí datum ve formátu "RRRRMMDD" tedy 17.07.2021 a číslo následující za tečkou značí pořadí, pokud bylo ten den více aktualizací, v tomto případě tedy pátá verze.



Obrázek 31 - Ukázka aplikace – Info (vlastní zpracování)

# 5 Testování aplikace

V této kapitole se věnuji testování správného chodu aplikace na ověřování čárových kódů na mnoha níže uvedených příkladech.

Pomocí online generátoru čárových kódů jsem připravil několik verzí obou používaných čárových kódu, a to v provedení se správným kódem a verzemi s nesprávnými délkami čárového kódu. A samozřejmě také úplně špatné typy čárových kódu libovolné délky, které aplikace vyhodnotí jako "Nesprávný kód".

### 5.1 Code 128

Pro potřeby testování byly vygenerovány tři níže uvedené čárové kódy typu Code 128 s různými délkami.

### 5.1.1 Testování Code 128 se správným počtem znaků



81212312312-1200

Obrázek 32 - Code 128 pro testování – správný počet znaků

(vlastní zpracování)

Výsledek testování:

20:10 🔜 🕬 🚥	O 📅 🧐 🖊 🕯 94 %
NASTAVENÍ	KONEC
INFO	

Čárový kód typu: Code 128 Hodnota: 81212312312-12001

	NASKENOVAT KÓD	
<	-	

Obrázek 33 - Výsledek Code 128 pro testování – správný počet znaků (vlastní zpracování)

### 5.1.2 Testování Code 128 s více znaky v čárovém kódu

Tento čárový kód obsahuje o pět znaků více, než je požadováno.



81212312312-1200112345

Obrázek 34 - Code 128 pro testování – více znaků (vlastní zpracování)

Výsledek testování:



Čárový kód typu: Code 128 NEPLATNÉ délky Hodnota: 81212312312-1200112345 Výsledek: navíc 5 znaků



Obrázek 35 - Výsledek Code 128 pro testování – více znaků (vlastní zpracování)

#### 5.1.3 Testování Code 128 s méně znaky v čárovém kódu

Tento čárový kód obsahuje o dva znaky méně, než je požadováno.



81212312312-121

### Obrázek 36 - Code 128 pro testování – méně znaků

(vlastní zpracování)

Výsledek testování:



Čárový kód typu: Code 128 NEPLATNÉ délky Hodnota: 81212312312-121 Výsledek: chybí 2 znak(y)



Obrázek 37 - Výsledek Code 128 pro testování – méně znaků (vlastní pracování)

(Anon., nedatováno)

### 5.2 Interleaved 2 of 5

Pro potřeby testování byly vygenerovány tři níže uvedené čárové kódy typu Interleaved 2 of 5 s různými délkami.

### 5.2.1 Testování Interleaved 2 of 5 se právním počtem znaků

Tento čárový kód obsahuje správný počet znaků.



Obrázek 38 - Interleaved 2 of 5 pro testování – správný počet znaků (vlatní zpracování)

Výsledek testování:

20:10 🖬 кв 🚥	🕕 🖼 🖞 🗖 🖉 94 %
NASTAVENÍ	KONEC
INFO	

Čárový kód typu: ITF Hodnota: 812123123128



Obrázek 39 - Výsledek Interleaved 2 of 5 (vlastní zpracování)

### 5.2.2 Testování Interleaved 2 of 5 s více znaky v čárovém kódu

Tento čárový kód obsahuje o šest znaků více, než je požadováno.



081212312312221458

#### Obrázek 40 - Interleaved 2 of 5 pro testování – více znaků (vlastní zpracování)

Výsledek testování:

20:11 🗳 кв 🐽	🕩 🖼 🖧 🖌 🕯 94 %
NASTAVENÍ	KONEC
INFO	

Čárový kód typu: ITF NEPLATNÉ délky Hodnota: 081212312312221458 Výsledek: navíc 6 znaků



Obrázek 41 - Výsledek Interleaved 2 of 5 pro testování – více znaků (vlastní zpracování)

#### 5.2.3 Testování Interleaved 2 of 5 s méně znaky v čárovém kódu

Tento čárový kód obsahuje o dva znaky méně, než je požadováno.



#### Obrázek 42 - Interleaved 2 of 5 pro testování – méně znaků (vlastní zpracování)

Výsledek testování:



Čárový kód typu: ITF NEPLATNÉ délky Hodnota: 8121231235 Výsledek: chybí 2 znak(y)



Obrázek 43 - Výsledek Interleaved 2 of 5 pro testování – méně znaků (vlastní zpracování)

(Anon., nedatováno)

### 5.3 Ostatní čárové kódy

Testování ostatních čárových kódu proběhlo na dvou vzorcích, a to kódů typu EAN-8 a Codabar.

### 5.3.1 EAN-8

Pro potřeby testu byl vytvořen kód s hodnotou 2021717 s kontrolním číslem 4



Obrázek 44 - EAN-8 pro testování (vlastní zpracování)

(Anon., nedatováno)

### 5.3.2 Codabar

Pro potřeby testu byl vytvořen čárový kód s hodnotou 20210717



Obrázek 45 - Codabar pro testování

(vlastní zpracování)

(Anon., nedatováno)

5.3.3 Výsledek testu ostatní čárových kódů

Vyhodnocení obou čárových kódů dopadlo dle očekávání viz. Obrázek níže.

19:58 KB 🛋 🚥	🗣 📅 🍄 🖌 🛔 95 %
NASTAVENÍ	KONEC
INFO	

Nesprávný typ čárového kódu

	NASKENOVAT KÓD	
<		

Obrázek 46 - Výsledek EAN-8 + Codabar (vlastní zpracování)

# 6 Možné vylepšení aplikace

Mobilní aplikace v aktuální verzi splňuje požadované funkce a pro svoji jednoduchost a přehlednost pro obsluhu není určitě v plánu aplikaci příliš zahalit novými funkcemi, které by aplikaci znepřehlednili.

Mezi několik plánovaných vylepšení aplikace je počítáno se zavedením přepínače na volbu režimu pro light/dark mód, který byl zmíněn v bodě 4.4.3.

Další možnosti vylepšení:

- Úprava designu tlačítek na modernější vzhled
- Úprava/změna použité knihovny pro skenování
- Refaktorování zdrojového kódu
- Rozpoznání typu zásilky z ITF kódu (Business, Private, Export)
- Rozpoznání dobírkové/nedobírkové zásilky z ITF kódu
- Výpočet kontrolní číslice u ITF kódu
- Rozpoznání čísla depa z code 128 a vyhodnocení, zda je reálné či nikoli
- Hlubší ověřování čárového kódu, jestli skladba znaků odpovídá zvyklostem

## Závěr

V této bakalářské práci byly zpracovány informace ohledně historie a přehledu nejpoužívanějších čárových kódů s detailnějším zaměřením na dva typy čárových kódů, které jsou používány na etiketách ve společnosti PPL.

Cílem této bakalářské práce bylo vytvořit funkční mobilní aplikaci pro operační systém Android, která bude umět vyhodnotit použitý typ čárového kódu na etiketě společnosti PPL. Tato aplikace po svém dokončení primárně slouží pro IT regionální podporu ve společnosti PPL, kde usnadňuje proces schvalování etiket vytvořených zákazníky společnosti.

Zadané cíle byly splněny a vytvořená mobilní aplikace funguje dle zadaných požadavků, jak je vidět na ukázkách z testování aplikace. Úplně se nepovedlo odladit zobrazení náhledu fotoaparátu při samotném snímání, kdy je zobrazovaný náhled jakoby "divně" přiblížený a toto bude potřeba vyladit, ale na funkci čtení čárových kódu to nemá vliv.

Pro komfortnější práci při ověřování správnosti čárového kódu byla přidána funkce na kontrolu případného počtu chybějících či naopak přebývajících znaků v čárovém kódu.

Další možné příklady rozšíření či vylepšení aplikace jsou uvedeny v předchozí kapitole a bude záležet na společnosti PPL, zda bude mít zájem o vylepšení a případně které požadavky budou mít jakou prioritu.

Zatím aplikaci testujeme v mém týmu IT regionální podpory, ale do budoucna by našla uplatnění i v provozu na depech, kde by mohla pomáhat provozním pracovníkům v identifikování příčiny pro technologii nečitelných zásilek. Na většině dep je na ATL sekce "no read" kam technologie odklání nečitelné zásilky, ať už z důvodu špatně nalepené etikety či etikety se špatným čárovým kódem. A v druhém případě může mobilní aplikace pomoci s identifikováním problému a následně můžeme kontaktovat zákazníka, aby provedl úpravu na etiketě pro zajištění bezproblémového čtení zásilek na ATL.

## Seznam použitých zdrojů

Anon., 2013. Android: how to make an activity return results to the activity which calls it? - Stack Overflow. [Online] Available at: https://stackoverflow.com/questions/14785806/android-how-to-make-anactivity-return-results-to-the-activity-which-calls-it [Přístup získán 10 07 2021]. Anon., 2021. Create app icons with Image Asset Studio / Android Developers. [Online] Available at: https://developer.android.com/studio/write/image-asset-studio [Přístup získán 10 07 2021]. Anon., nedatováno 7. diel - Android programovanie - Welcome screen animácie *kalkulačky*. [Online] Available at: https://www.itnetwork.sk/java/android/tutorial-prvni-aplikacie-kalkulackawelcome-screen-animacie [Přístup získán 10 03 2021]. Anon., nedatováno Download Android Studio and SDK tools & https://& https://white.com/android/a Developers. [Online] Available at: https://developer.android.com/studio [Přístup získán 02 02 2021]. Anon., nedatováno *Espacenet – search results*. [Online] Available at: https://worldwide.espacenet.com/patent/search/family/022402610/publication/US26129 94A?q=pn%3DUS2612994 [Přístup získán 03 02 2021]. Anon., nedatováno Free Online Codabar Barcode Generator to create, draw linear Codabar barcode images to encode Numeric Data Digits. Save Codabar to insert in *html and other applications.* [Online] Available at: http://www.keepautomation.com/online\_barcode\_generator/codabar/ [Přístup získán 17 07 2021]. Anon., nedatováno Free Online Code 128 Barcode Generator to create, draw linear Code 128 barcode images to encode AlphaNumeric Data. Save Code 128 to insert in *html and other applications.* [Online] Available at: http://www.keepautomation.com/online\_barcode\_generator/code\_128/ [Přístup získán 17 07 2021]. Anon., nedatováno Free Online EAN-8 Barcode Generator to create, draw linear EAN-8 barcode images to encode numeric data/ text string. Save EAN-8 images to insert in *html and other applications.* [Online] Available at: http://www.keepautomation.com/online\_barcode generator/ean 8/ [Přístup získán 17 07 2021]. Anon., nedatováno Free Online Interleaved 2 of 5 Barcode Generator to create, draw linear Interleaved 2 of 5 barcode images to encode numeric data/ text string. Save *Interleaved 2 of 5 images to insert in html and other applications.* [Online] Available at: http://www.keepautomation.com/online\_barcode\_generator/interleaved25/ [Přístup získán 17 07 2021]. Anon., nedatováno The code 128. [Online] Available at: http://grandzebu.net/informatique/codbar-en/code128.htm [Přístup získán 04 02 2021]. Barcoding, nedatováno The History of Barcodes - Barcoding, Inc. - Barcoding, Inc.. [Online]

Available at: <u>https://www.barcoding.com/resources/barcoding-basics/the-history-of-</u>

barcodes/ [Přístup získán 03 02 2021]. Codeware, nedatováno Blog - Lineární čárové kódy. [Online] Available at: https://www.codeware.cz/blog/linearni-carove-kody [Přístup získán 04 02 2021]. David, R., 2019. GitHub - Rodiman/AndroidBarcodeReaderTemplate: This is a template for a barcode reader app in Android, prepared to code whatever you need after getting the barcode from a button in the main activity.. [Online] Available at: https://github.com/Rodiman/AndroidBarcodeReaderTemplate [Přístup získán 08 07 2021]. EasierSoft, nedatováno EasierSoft - Free Bulk Online Multiple Code 128 Barcode Generator & Designer. [Online] Available at: http://barcode.design/Code128Online.asp [Přístup získán 05 02 2021]. Faltýnek, L., 2007. Java – dnes při šálku dobré kávy - Linux E X P R E S. [Online] Available at: https://www.linuxexpres.cz/praxe/java-dnes-pri-salku-dobre-kavy [Přístup získán 31 01 2021]. Gaben, nedatováno *Čárové kódv (teorie)*. [Online] Available at: http://www.gaben.cz/cz/faq/carove-kody-teorie [Přístup získán 02 02 2021]. Google.com, 2021. Barcode | Google Play services | Google Developers. [Online] Available at: https://developers.google.com/android/reference/com/google/android/gms/vision/barcod e/Barcode [Přístup získán 15 07 2021]. IDAutomation, nedatováno Interleaved 2 of 5 Barcode FAQ & amp; Tutorial / *BarcodeFAQ.com.* [Online] Available at: https://www.barcodefaq.com/1d/interleaved-2of5/ [Přístup získán 6 2 2021]. ITnetwork, nedatováno Android - Mobilní aplikace v Javě. [Online] Available at: https://www.itnetwork.cz/java/android [Přístup získán 15 04 2021]. Kodys, nedatováno Čárový kód | Kodys. [Online] Available at: https://www.kodys.cz/technologie/carovy-kod [Přístup získán 02 02 2021]. PPL, nedatováno Jsme PPL. Doručujeme zásilky | PPL CZ. [Online] Available at: https://www.ppl.cz/o-nas [Přístup získán 29 01 2020]. Ravi, T., 2017. Android Barcode / QR Code Scanner using Google Mobile Vision -Building Movie Tickets App. [Online] Available at: https://www.androidhive.info/2017/08/android-barcode-scanner-usinggoogle-mobile-vision-building-movie-tickets-app/ [Přístup získán 10 07 2021]. Softmatic, nedatováno Code 128 Barcode Explained - Describes Code 128 A, B, C -*Code 128 Sample Barcodes, Generator*. [Online] Available at: https://www.softmatic.com/barcode-code-128.html [Přístup získán 04 02 2021]. Techopedia, 2020. What is a Mobile Application? - Definition from Techopedia. [Online] Available at: https://www.techopedia.com/definition/2953/mobile-application-mobile<u>app</u>

[Přístup získán 01 02 2021]. w3schools.com, nedatováno *Java Tutorial*. [Online] Available at: <u>https://www.w3schools.com/java/</u> [Přístup získán 10 05 2021]. Xavier Ducrohet, T. N. K. C., 2013. *Android Developers Blog: Android Studio: An IDE built for Android*. [Online] Available at: <u>https://android-developers.googleblog.com/2013/05/android-studio-ide-built-for-android.html</u>

[Přístup získán 01 02 2021].