

GYMNÁZIUM, PRAHA 6, ARABSKÁ 14
Programování

NÁVOD



Vyučující: Jan Lána

Prohlášení

Prohlašuji, že jsem jediným autorem tohoto projektu, všechny citace jsou řádně označené a všechna použitá literatura a další zdroje jsou v práci uvedené. Tímto dle zákona 121/2000 Sb. (tzv. Autorský zákon) ve znění pozdějších předpisů uděluji bezúplatně škole Gymnázium, Praha 6, Arabská 14 oprávnění k výkonu práva na rozmnožování díla (§ 13) a práva na sdělování díla veřejnosti (§ 18) na dobu časově neomezenou a bez omezení územního rozsahu

Dne:_____

Podpis_____

Anotace

Cílem projektu bylo vytvořit aplikaci, na telefony s operačním systémem Android, umožňující vytvoření a uschování vlastních návodů. Návod se skládá z obrázků a popisku na samostatných stránkách. Obrázek bude možné nahrát z galerie, či vyfotit. Stránky jsou zpětně upravitelné a listuje se v nich jako v normálním návodu.

The goal of this project was to create an application for mobile phones with Android operating system, which would enable the user to create and save instruction manuals. Instruction manual pages are composed of pictures and description. Picture can be chosen from gallery or taken as a new one. Pages are backwards editable and are supposed to look like in normal instruction booklet.

Obsah

1. Úvod.....	5
2. Zadání.....	5
3. Provedení.....	5
4. Uživatelské prostředí.....	6
4.1 Menu	6
4.2 Vytvoření návodu	6
4.3 Prohlížení a úprava návodu	7
5. Kód.....	8
5.1 MainActivity	8
5.2 ReadActivity	9
6. Fragmenty	11
6.1 Úvodní strana návodu	11
6.2 Strany návodu s instrukcemi	12
6.3 Uložení návodu.....	13
7. Závěr	14
8. Citovaná literatura.....	14

1. Úvod

Cílem ročníkové práce bylo naprogramovat aplikaci pro telefony s operačním systémem Android na vytvoření návodů. Návody jsou v dnešní době přibaleny u velkého množství produktů, přesto jsou zde však situace, ve kterých není návod dostupný, či dostatečně rozsáhlý. V těchto případech je dobré si návod vytvořit, aby se při možném opakování situace předešlo opětovnému problému. Například při opravě kola nemusí návod existovat, v tomto případě se musí člověk uchýlit k vyhledávání na internetu, které je často zdouhavé a někdy nemusí ani přinést tížený výsledek. Pokud návod nelze najít, je nutné si postup vytvořit, a protože je postup velmi často zapomenut, také si ho v případě příštího použití zaznamenat. Pokud je však postup napsán na kus papíru, je zde velké riziko ztráty.

Každý člověk však vlastní mobilní telefon, do kterého lze postup napsat a následně ho uchovat. Toto řešení však také není ideální, jelikož pouhý popis bez obrázků je těžký na vybavení. V aplikaci je možné vytvořit návod, který si uživatel bude moci napsat, při následovném použití se dle návodu orientovat, v případě chyb či vylepšení návod následně upravit.

2. Zadání

Ročníková práce bude vytvořit program na výrobu a ukládání návodů. Návody se budou skládat z obrázků a popisků. Obrázky bude možné do programu nahrát a následně upravit. Při vytváření i upravování návodu bude možné popisek s obrázkem barevně zkoordinovat (Barevně vyznačit část na obrázku, kde stejná barva bude vedle bodu v návodu).

3. Provedení

Aplikace je naprogramována v jazyku `Java`. Na vývoj bylo použito programovací prostředí `Android Studio`, které umožňuje snadnou integraci aplikace přímo do telefonu. Také je díky předinstalovanému emulátoru možné aplikaci vyzkoušet bez nutnosti zdouhavé instalace do telefonu. Nativní prostředí doplňuje `API Glide`, která umožňuje snadné a rychlé načítání obrázku s minimem kódu, hlavně při procházení návodu, bez nutnosti čekání. (Bumptechn, 2019)

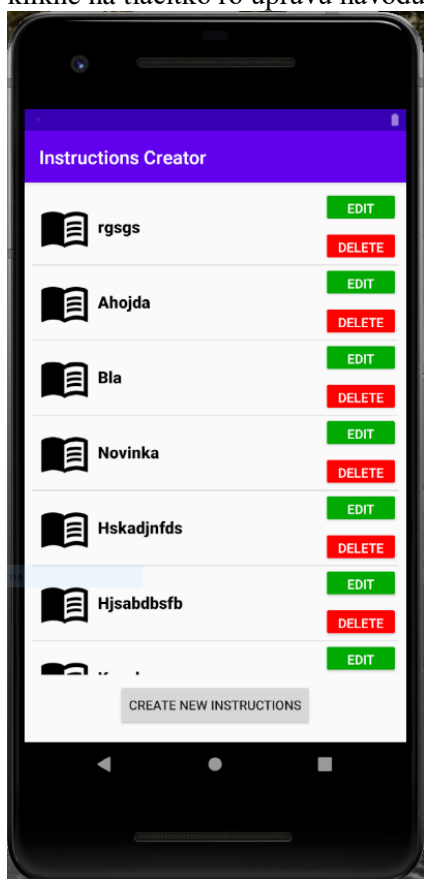
Aplikace pro operační systém android jsou uceleny do aktivit. Každá aktivita reprezentuje jedno zobrazení v uživatelském prostředí. Každá aktivita se skládá z `xml` souboru a souboru `java`, které mezi sebou komunikují. (Google Developers, 2019)

4. Uživatelské prostředí

4.1 Menu

Při spuštění aplikace se uživatel dostane na výchozí obrazovku. Tato obrazovka bude při prvním spuštění prázdná, pokud jsou však v aplikaci již uloženy návody, budou zobrazeny na této obrazovce ve formě seznamu. Položky v seznamu jsou seřazeny podle data vytvoření. Pod seznamem je umístěno tlačítko, které po kliknutí vytvoří nový návod.

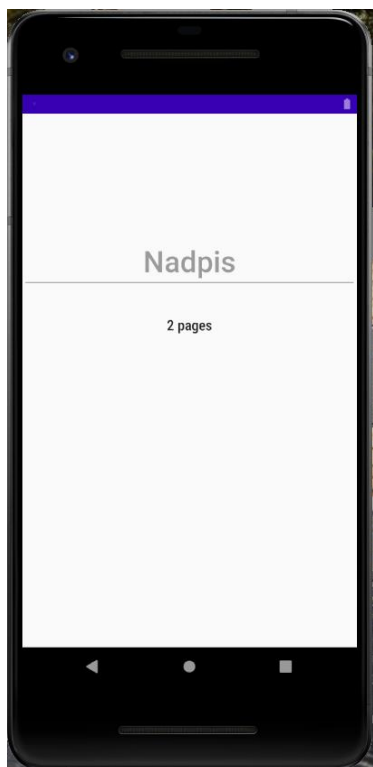
Každý návod má v seznamu svou vlastní kolonku. Kolonka se skládá z názvu, loga a dvou tlačítek. Zeleného na úpravu návodu a červeného na odstranění návodu. Po kliknutí na kolonku návodu, mimo tlačítko na úpravu návodu, se návod otevře v režimu pro listování. Pokud uživatel klikne na tlačítko pro úpravu návodu, návod se otevře v režimu pro úpravu.



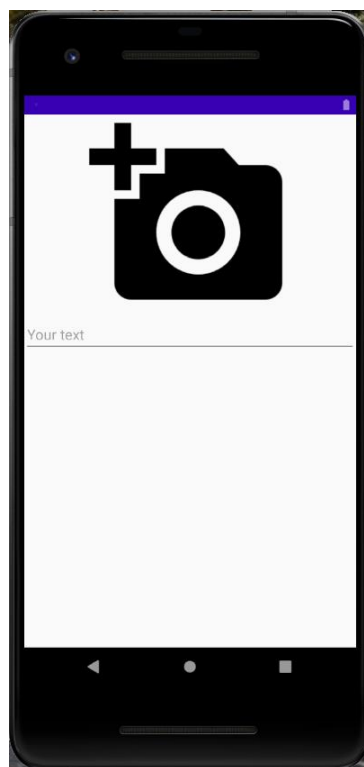
Obrázek 1 - Menu

4.2 Vytvoření návodu

Tlačítko pro vytvoření nového návodu otevře novou úvodní stránku. Úvodní stránka obsahuje název projektu a počet stran v návodu. Název může obsahovat libovolné znaky, jediné omezení je délka, která činí 24 znaků. Výchozí nastavení počtu stran jsou dvě, přičemž i úvodní strana se do celkového počtu nezapočítává. Po přetáhnutí prstem, stejně jako při otočení strany, se objeví první strana samotného návodu. Každá stránka obsahuje obrázek a text. Obrázek se dá změnit při kliknutí na výchozí ikonu přidání fotografie. Po kliknutí na ikonu se otevře menu, ve kterém je možné si vybrat ze všech fotografií a obrázků uložených v zařízení, popřípadě na SD kartě. Při dlouhém podržení na ikonu se otevře fotoaparát, a po vyfocení nové fotografie, se fotografie zobrazí v okně. Při vybrání obrázku je obrázek nastaven do vrchní části stránky. Po kliknutí na okno pro zadání textu je možné popsat kroky či postupy k obrázku. Pro vytvoření následujících stran, se znovu přetáhne prstem, jako při listování. Návod se automaticky uloží při návratu do hlavního menu, pro návrat do hlavního menu je nutné si projít návod zpětně. Tento postup byl zvolen záměrně, kvůli kontrole chyb v textu.



Obrázek 2 - Úvodní strana

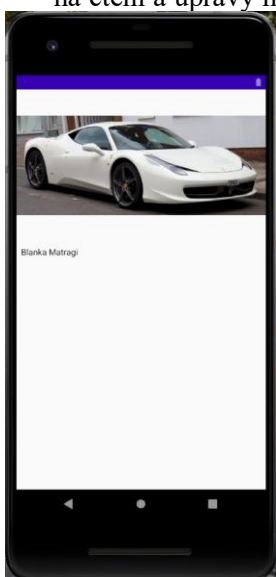


Obrázek 3 - Strana návodu

4.3 Prohlížení a úprava návodu

Prohlédnutí návodu se inicializuje kliknutím na pole s návodem. Po kliknutí se otevře úvodní strana s názvem a počtem stran. Návodem lze listovat dle libosti, pro návrat je však nutné projít návod až na začátek, nebo zavřít aplikaci.

Pro úpravu návodu je nutné návod otevřít pomocí tlačítka na úpravu. V tomto režimu je možné upravit na První straně název a na dalších jednotlivý obrázek či text. Po ložení návodu je nutné se vrátit z hlavní strany návodu do menu. Toto řešení bylo zvoleno pro maximální využití místa na čtení a úpravy návodu.



Obrázek 4 - Prohlížení

5. Kód

5.1 MainActivity

Při prvním otevření aplikace je uživatel dotázán na udělení povolení aplikace k získání obsahu z paměti telefonu, bez udělení tohoto povolení nelze aplikaci používat (Google Developers, 2019). Pokud uživatel povolení bude znovu dotázán na udělení. Po udělení se v aplikaci zobrazí seznam návodů. Každý návod v seznamu je uložen jako samostatný soubor, tyto soubory jsou uloženy v paměti aplikace a uživatel k těmto datům nemá mimo aplikaci přístup.

```
File path = getFilesDir();

File[] files = path.listFiles();
Log.d( tag: "Files", msg: "Size: " + files.length);
for (int i = 0; i < files.length; i++) {
    Log.e( tag: "Files", msg: "FileName:" + files[i].getName());
    navody.add(files[i].getName().replace( target: ".txt", replacement: ""));
}
```

Obrázek 5 - Přidání návodů

Vytvoření seznamu začne v aplikaci načtením vnitřní paměti (Yury, 2011) (Google Developers, 2019) a posléze vytvořením seznamu, do kterého jsou uloženy jednotlivé názvy seznamů, díky shodě názvu souboru a názvu seznamu zde není nutnost pamatovat si názvy. Aplikace pak díky adaptéru, který zjistí z listu název a přidělí ho k návodu do kolonek. (Codepath, 2019)

```
TextView textname = convertView.findViewById(R.id.textView);
textname.setOnClickListener((v) -> {
    Intent intent = new Intent(getContext(), ReadActivity.class);

    intent.putExtra( name: "EDITABLE", value: false);
    intent.putExtra( name: "TITLE", name);
    startActivity(intent);
});

convertView.findViewById(R.id.button2).setOnClickListener((v) -> {
    Intent intent = new Intent(getContext(), ReadActivity.class);
    intent.putExtra( name: "EDITABLE", value: true);
    intent.putExtra( name: "TITLE", name);
    startActivityForResult(intent, requestCode: 5);
});
```

Obrázek 6 - Tlačítka na otevření a editaci

Adaptér se také stará o určení správné pozice při kliknutí na tlačítka u návodu, či na návod samotný. (Codepath, 2019)

Pro ulehčení náročnosti na paměť jsou vytvořeny pouze viditelné kolony, které se při posunutí seznamu dolů či nahoru znovu využijí na nově zobrazené návody. Tento řádek kód napomůže plynulosti uživatelského prostředí. (Slidenerd, 2013)

```
if (convertView == null) {
    convertView = LayoutInflater.from(getContext()).inflate(R.layout.instruct, parent, attachToRoot: false);
}
```

Obrázek 7 – ConvertView

Při kliknutí na návod či tlačítko na úpravu návodu aplikace zavolá třídu `ReadActivity` a k zavolání přidá dva údaje. Pokud uživatel zvolil možnost úpravy návodu je do třídy `ReadActivity` odesláno jméno návodu v podobě `String` a boolean `true`, který informuje, že se jedná o upravení návodu. V případě kliknutí na návod samotný, je do třídy

`ReadActivity` odesláno jméno návodu a `boolean false`, aby se návod otevřel pouze v podobě pro čtení.

Kliknutím na tlačítko pro smazání se zjistí název návodu a následně je návod odstraněn z paměti aplikace. Po odstranění se znovu načte seznam, tentokrát již bez vymazaného návodu (plugmind, 2010), (Yadav, 2013).

Při stisknutí tlačítka pro tvorbu nového návodu je uživatel přesměrován také do `ReadActivity`. V tuto chvíli je předána jen informace `boolean true`, s kódem požadavku (K, 2013). Pokud vše v třídě `ReadActivity` proběhne úspěšně a uživatel je po ukončení tvorby návodu navrácen do hlavního menu, `MainActivity` provede aktualizaci seznamu (Shardul, 2010).

5.2 ReadActivity

Třída `ReadActivity` je stěžejní třídou projektu. Tato třída se stará o získávání dat z paměti a jejich následné zobrazení či o jejich úpravu. Třída také spolupracuje s dalšími třídami, které se starají o samotné rozložení stran.

Tato třída je zavolána z třídy `MainActivity` s dalšími informacemi viz (5.1). Získané jméno poslouží k načtení návodu z paměti aplikace a `boolean` se postará o zjištění úpravy. Pokud chce uživatel vytvořit nový návod, třída rozpozná, že návod v paměti aplikace není a vytvoří pro něj novou hlavní stranu a stranu pro instrukce (MBH, 2013).

```
Intent intent = getIntent();
editable = intent.getBooleanExtra( name: "EDITABLE", defaultValue: false);
instructionsTitle = intent.getStringExtra( name: "TITLE");
File path = getFilesDir();
File file = new File(path, child: instructionsTitle + ".txt");

String inputtext;
//Načtení a zjištění existence složky
if (file.exists()) {
    int length = (int) file.length();
    byte[] bytes = new byte[length];

    try {
        FileInputStream in = new FileInputStream(file);
        in.read(bytes);
        in.close();
    } catch (FileNotFoundException e) {
        e.printStackTrace();
    } catch (IOException e) {
        e.printStackTrace();
    }
}
```

Obrázek 8 - Získání informací a otevření

Tato část práce byla jednou z nejproblematictějších částí, protože bylo velmi těžké najít způsob uložení návodů bez velké náročnosti na paměť telefonu. Obrázky a fotografie jsou již v telefonu uloženy a jejich nové uložení není efektivní využití paměti, především v případech u telefonů s menším úložištěm.

Po načtení souboru typu `txt`, je informace typu `JSON` získána a její obsah v bytech a následně převeden na text. Z textu jsou získány informace o názvu, počtu stránek a jednotlivých údajích na každé stránce.

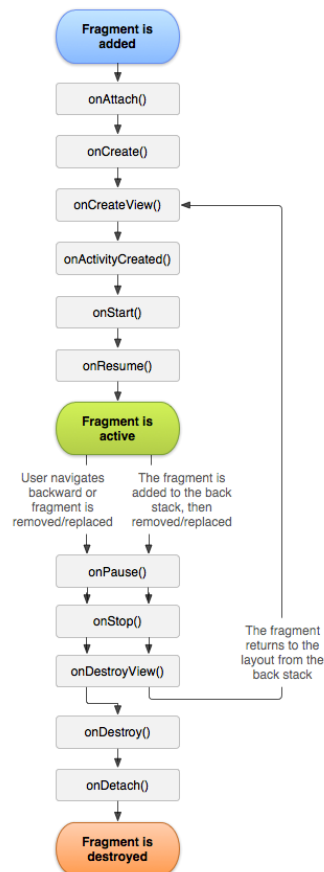
```

inputtext = new String(bytes);
Log.e( tag: "JSON Data", inputtext);
JSONObject instructs = new JSONObject(inputtext);
instructionsTitle = instructs.getString( name: "title");
NUM_PAGES = instructs.getInt( name: "num_pages");
JSONArray pagedata = instructs.getJSONArray( name: "pages");
if (editable) {
    mainPageEditable = new MainPageEditable();
    pages = new ArrayList<EditPageFragment>();
    for (int i = 0; i < pagedata.length(); i++) {
        JSONObject page = pagedata.getJSONObject(i);
        EditPageFragment ep = new EditPageFragment();
        ep.setText(page.getString( name: "text"));
        ep.setImage(page.getString( name: "img"));
        pages.add(ep);
    }
}

```

Obrázek 9 - Načtení návodu

Informace o hlavní straně a nesledujících stranách jsou převedeny do listu ve formátu fragmentu (Google Developers, 2019). Fragmenty jsou elementy, které jsou v jedné aktivitě. Fragmenty však fungují nezávisle v aktivitách. Díky jejich nezávislosti se mohou měnit a upravovat v průběhu fungování aktivity. Pokud je však aktivita pozastavena, jsou pozastaveny i fragmenty.



Obrázek 10 - Život fragmentů

Fragmenty jsou zobrazovány pomocí viewPager (CodingWithMitch, 2017), který z fragmentů vytvoří pomocí adaptéru ucelený návod. viewPager se stará nejen o zobrazení návodu, ale také hlídá pozici zobrazené stránky (Google Developers, 2019). Pokud jde o běžné listování návodem, musí adaptér na hlavní straně rozpoznat zmáčknutí tlačítka zpět pro zavření návodu. Když je však návod v režimu pro úpravu, je sledována pozice předposlední strany, tak aby se včas vytvořila následující strana, pokud by chtěl přidat instrukce. Také je sledováno stisknutí tlačítka zpět na hlavní straně, při kterém se celý návod znovu načte a následně uloží (R9J, 2013).

```

mPager = findViewById(R.id.readpager);
mPager.setOffscreenPageLimit(NUM_PAGES + 1);
pagerAdapter = new ScreenSlidePagerAdapter(getSupportFragmentManager());
mPager.addOnPageChangeListener(new ViewPager.OnPageChangeListener() {
    @Override
    public void onPageScrolled(int position, float positionOffset, int positionOffsetPixels) {
    }
    //vytvoreni dalsich stran pri uprave
    @Override
    public void onPageSelected(int position) {
        if (position == NUM_PAGES - 1 && NUM_PAGES != 100 && editable) {
            pages.add(new EditPageFragment());
            NUM_PAGES++;
            mPager.setOffscreenPageLimit(NUM_PAGES + 2);
            pagerAdapter.notifyDataSetChanged();
        }
    }
    @Override
    public void onPageScrollStateChanged(int state) {
    }
});

```

Obrázek 11 - Listování a přidání strany

6. Fragmenty

6.1 Úvodní strana návodu

Fragment na úvodní stranu je v aplikaci uložen ve dvou provedeních pro úpravu a čtení. Kódy obou dvou fragmentů jsou jednoduché. Jejich vizuální stránka je stejná. Vzhled stránek je definován v samostatném xml souboru. Tento vzhled je pak pomocí `LayoutInflater` přenesen do `ViewPager`. Fragment na čtení získá z třídy `ReadActivity` informaci v souboru jméno a datum. V případě upravitelné hlavní stránky se mění pouze možnost upravení názvu návodu, která následně změní název uloženého souboru.

```

public View onCreateView(LayoutInflater inflater, ViewGroup container,
    Bundle savedInstanceState) {
    ViewGroup rootView = (ViewGroup) inflater.inflate(
        R.layout.main_page_editable, container, attachToRoot: false);
    title = rootView.findViewById(R.id.editNadpis);
    numofpages = rootView.findViewById(R.id.numofpagesedit);
    if (tit != null) {
        title.setText(tit);
    }
    if (pages != -1) {
        numofpages.setText(pages + " pages");
    }

    return rootView;
}

//Upraveny nazev
public String getTitle() {
    return title.getText().toString();
}

//Ziskani puvodniho nazvu z ReadActivity
public void setTitle(String title1) {
    tit = title1;
}

//Ziskani poctu stran
public void setPageCount(int num) {
    if (numofpages == null) {
        pages = num;
    } else {
        numofpages.setText(num + " pages");
    }
}

```

Obrázek 12 - Úvodní strana návodu

6.2 Strany návodu s instrukcemi

Další strany v návodu se skládají z obrázku a textu. Při úpravě návodu se u prvního zobrazení objeví na místě obrázku ikona s přidáním fotografie. Při kliknutí na ikonu se otevře úložiště telefonu s fotkami a obrázky (Kahaduwa, 2018). Po výběru fotky kliknutím se otevře zpět návod s nahraným obrázkem fotky (K, 2013). Pro vyfocení nové fotky je nutné podržet chvíli obrázek, poté se otevře fotoaparát (Calvo, 2015) (Google Developers, 2019). Nahrání obrázku probíhá pře URI obrázku s pomocí knihovny Glide. Text se načte z ReadActivity a jeho následná úprava bude probíhat v textovém poli, kterým je možné listovat vertikálně a uživatel není omezen počtem znaků. Řádek může obsahovat pouze text na šířku obrazovky, aby jeho čtení nebylo nepříjemné (Maillet, 2010). Na podobném principu pracuje i fragment pro prohlížení. Fragment bude mít načtený text i URI obrázku z ReadActivity a pouze je zobrazí.

```

image.setOnClickListener((v) -> {
    if (ContextCompat.checkSelfPermission(getActivity(), Manifest.permission.READ_EXTERNAL_STORAGE)
        != PackageManager.PERMISSION_GRANTED) {
        ActivityCompat.requestPermissions(getActivity(),
            new String[]{Manifest.permission.READ_EXTERNAL_STORAGE},
            requestCode: 3);
        // Permission is not granted
    } else {
        Intent i = new Intent(Intent.ACTION_OPEN_DOCUMENT, android.provider.MediaStore.Images.Media.EXTERNAL_CONTENT_URI);
        i.setType("image/*");

        startActivityForResult(i, requestCode: 1);
    }
});

```

Obrázek 13 - Výběr obrázek

```

image.setOnLongClickListener((v) → {
    if (ContextCompat.checkSelfPermission(getActivity(), Manifest.permission.READ_EXTERNAL_STORAGE)
        != PackageManager.PERMISSION_GRANTED) {
        ActivityCompat.requestPermissions(getActivity(),
            new String[]{Manifest.permission.READ_EXTERNAL_STORAGE},
            requestCode: 3);
    } else if (ContextCompat.checkSelfPermission(getActivity(), Manifest.permission.CAMERA)
        != PackageManager.PERMISSION_GRANTED) {
        ActivityCompat.requestPermissions(getActivity(),
            new String[]{Manifest.permission.CAMERA},
            requestCode: 3);
    } else {
        //Starani se, aby se fotilo na jinem vlaknu nez bezi aplikace
        StrictMode.VmPolicy.Builder builder = new StrictMode.VmPolicy.Builder();
        StrictMode.setVmPolicy(builder.build());
        Intent i = new Intent(MediaStore.ACTION_IMAGE_CAPTURE);
        // Vytvoreni slozky
        File photoFile = null;
        try {
            photoFile = createImageFile();
        } catch (IOException ex) {
            Log.i( tag: "ERROR", msg: "IOException");
        }
        // Ulozeni fotky
        if (photoFile != null) {
            i.putExtra(MediaStore.EXTRA_OUTPUT, Uri.fromFile(photoFile));
            startActivityForResult(i, requestCode: 0);
        }
    }

    return true;
});

```

Obrázek 14 - Pořízení foto

6.3 Uložení návodu

Každý návod je uložen jako textový soubor s objektem `JSON`. Tento soubor obsahuje název návodu, počet stran a list, ve kterém jsou uloženy jednotlivé údaje z každé strany. Aplikace sama rozezná, jak dlouhý je návod a co je na samostatných stranách. Obrázky a fotky nejsou v souboru uloženy. Je k nim uložena cesta, pomocí které získá aplikace jejich `URI`.

Příklad:

```

{"id":1,"text":"Blanka Matragi",
"img":"content:\\\\com.android.providers.media.documents\\document\\image%3A28"
}

```

7. Závěr

S programem jsem celkem spokojen. Většina funkcí vymezená ze zadání pracuje, jak má. Jediné co se nepodařilo, je barevné sdružení popisku a obrázku. Tento problém jsem nedokázal vyřešit, kvůli náročnosti úpravy textu při psaní a rozpoznání barev na uloženém obrázku. Tento problém by pravděpodobně byl vyřešitelný s větším množstvím času a lepší znalostí programování pro Android, to však nebylo během roku dosažitelné.

Díky aplikaci jsem se naučil základy programování na operační systém Android, což je vhodná dovednost do budoucnosti, která bude využívat čím dál více mobilních aplikací na běžné úkony v běžném životě.

8. Citovaná literatura

- Bumptech. (2019). *Github Bumptech Glide Documentation*. Získáno 14. Březen 2020, z Github: <http://bumptech.github.io/glide/>
- Calvo, A. V. (10. Srpen 2015). *Stack Overflow*. Získáno 23. Prosinec 2019, z Stack Overflow: <https://stackoverflow.com/questions/5991319/capture-image-from-camera-and-display-in-activity?fbclid=IwAR38x7kqQ312UZTGYRkK6fcJXkUXKIKLGB19Uog3zVlcYmMeoMQIP-88SIQ>
- Codepath. (17. Srpen 2019). *Guides Codepath*. Získáno 29. Prosinec 2019, z Codepath: <https://guides.codepath.com/android/Using-an-ArrayAdapter-with-ListView>
- CodingWithMitch. (10. Duben 2017). *Youtube*. Získáno 30. Listopad 2019, z Youtube: <https://www.youtube.com/watch?v=UqtsyhASW74&list=PLgCYzUzKIBE-1BsxIQTIpF7B95ey34jUF&index=2>
- Google Developers. (27. Prosinec 2019). *Android Developers*. Získáno 21. března 2020, z Android Developers: <https://developer.android.com/guide>
- Google Developers. (27. Prosinec 2019). *Android Developers Reference*. Získáno 20. Březen 2020, z Android Developers: <https://developer.android.com/reference/>
- Google Developers. (27. Prosinec 2019). *Android Developers Training*. Získáno 14. Březen 2020, z Android Developers: <https://developer.android.com/training/>
- Google Developers. (2019). *Material*. Získáno 15. Listopad 2019, z Material Design: https://material.io/resources/icons/?icon=menu_book&style=baseline
- K, p. (12. Leden 2013). *Stack Overflow*. Získáno 12. Leden 2020, z Stack Overflow: <https://stackoverflow.com/questions/14292398/how-to-pass-data-from-2nd-activity-to-1st-activity-when-pressed-back-android>
- Kahaduwa, H. (22. Leden 2018). *Medium*. Získáno 30. Listopad 2019, z Medium: <https://medium.com/@hasangi/capture-image-or-choose-from-gallery-photos-implementation-for-android-a5ca59bc6883>
- Maillet, J. (18. Červenec 2010). *Stack Overflow*. Získáno 24. Březen 2020, z Stack Overflow: <https://stackoverflow.com/questions/3276380/android-word-wrap-edittext-text>
- MBH. (26. Duben 2013). *Stack Overflow*. Získáno 29. Prosinec 2020, z Stack Overflow: <https://stackoverflow.com/questions/16237950/android-check-if-file-exists-without-creating-a-new-one>
- plugmind. (24. Říjen 2010). *Stack Overflow*. Získáno 15. Duben 2020, z Stack Overflow: <https://stackoverflow.com/questions/3554722/how-to-delete-internal-storage-file-in-android>
- R9J. (17. Leden 2013). *Stack Overflow*. Získáno 2. Leden 2020, z Stack Overflow: <https://stackoverflow.com/questions/14376807/how-to-read-write-string-from-a-file-in-android>
- Shardul. (27. Prosinec 2010). *Stack Overflow*. Získáno 30. Listopad 2019, z Stack Overflow: <https://stackoverflow.com/questions/4540754/how-to-add-elements-to-a-listview-dynamically-in-android>

Slidenerd. (27. Červenec 2013). *Youtube*. Získáno 11. Duben 2020, z Youtube:
<https://www.youtube.com/watch?v=KHTi28zT7rI>
Yadav, C. (30. Leden 2013). *Tutorialspoint*. Získáno 25. Duben 2020, z
<https://www.tutorialspoint.com/how-to-reload-activity-in-android>
Yury. (27. Prosinec 2011). *Stack Overflow*. Získáno 29. Březen 2020, z Stack Overflow:
<https://stackoverflow.com/questions/8646984/how-to-list-files-in-an-android-directory>

Obrázek 1 - Autor

Obrázek 2 - Autor

Obrázek 3 - Autor

Obrázek 4 - Autor

Obrázek 5 - Autor

Obrázek 6 - Autor

Obrázek 7 - Autor

Obrázek 8 - Autor

Obrázek 9 - Autor

Obrázek10 - https://cms-assets.tutsplus.com/uploads/users/798/posts/26169/image/fragment_lifecycle_1.png

Obrázek 11 - Autor

Obrázek 12 - Autor

Obrázek 13 - Autor

Obrázek 14 - Autor