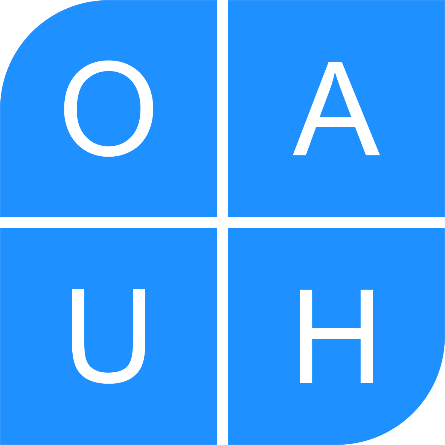
Obchodní akademie, Vyšší odborná škola a Jazyková škola s právem státní jazykové zkoušky Uherské Hradiště



Maturitní projekt

Docházková aplikace

2023 Tomáš Hanáček, 4.B

Obsah obrázku stůl

Popis byl vytvořen automaticky

Obsah obrázku stůl

Popis byl vytvořen automaticky

Prohlášení:

Souhlasím s tím, že s výsledky mé práce může být naloženo podle uvážení vedoucího maturitní práce a ředitele školy. V případě publikace budu  jako spoluautor.

Prohlašuji, že jsem na celé maturitní práci  samostatně a veškeré použité zdroje jsem .

V Uherském Hradišti, dne

..................................................  
 podpis absolventa

RESUMÉ

Aplikace slouží k evidenci docházky žáků. Aplikace umožňuje jednoduchý a přehledný způsob zaznamenávání docházky. Díky aplikaci mohou lektoři snadno zapisovat docházku do kroužku či jiné lekce. Docházková aplikace je také užitečná pro lektory, kteří potřebují zapisovat docházku pro více kroužků či lekcí najednou. Aplikace umožňuje zobrazit, zda jsou žáci přítomni, či nikoliv. Celkově lze říci, že docházková aplikace je užitečným nástrojem pro správu docházky žáků a poskytuje lektorům snadný a efektivní způsob zaznamenávání a sledování pracovní doby.

OBSAH

úvod 8

1 Uživatelská dokumentace 9

1.1 Instalace 9

1.2 Popis aplikace 10

1.3 Spuštění aplikace 11

1.4 Obrazovka pro registraci 12

1.5 Úvodní stránce 14

1.6 Menu na úvodní stránce 15

1.7 Stránka výběru a přidání lekce 16

1.8 Vytvoření nového data 20

1.9 Vytvoření žáci 23

1.9.1 Filtrování seznamu žáků 27

1.10 Přidávání žáka 28

1.10.1 Podmínky vytvoření žáka 29

1.11 Editace žáka 31

1.12 Zápis docházky 32

2 Programátorská dokumentace 35

2.1 Úvod 35

2.2 Firebase 36

2.2.1 Použité pluginy 37

2.2.2 Implementace pluginů 37

2.2.3 Použité Firebase služby 38

2.3 Databáze 39

2.4 Metody 40

2.4.1 Metoda *Prihlasovani* 40

2.4.2 Metoda *Dispose* u příhlášení 41

2.4.3 Metoda *Dispose* u registrace 41

2.4.4 Metoda *Registrace* 42

2.4.5 Metoda \_*ukazKalendar* 42

2.4.6 Metoda *hesloSpravne* 43

2.4.7 Metoda *\_vymazVse* 43

2.5 Třídy 44

2.5.1 Třída *Uvodni* 44

2.5.2 Třída *Cesty* 45

2.5.3 Třída *Autorizacni* 46

2.5.4 Třída *Prihlaseni* 47

2.5.5 Třída *Registrace* 50

2.5.6 Třída *HlavniStranka* 52

2.5.7 Třída *VytvoreniLekce* 56

2.5.8 Třída *ZapsaniDochazky* 57

2.5.9 Třída *SeznamLidi* 58

2.5.10 Třída *PridatLidi* 61

2.5.11 Třída *Editace* 63

2.5.12 Třída *Nezaplaceni* 63

2.5.13 Třída *NeniGDPR* 63

Závěr 65

seznam použité literatury 66

Seznam obrázků 67

Seznam Tabulek 68

seznam Příloh 69

úvod

V dnešní době existuje mnoho zájmových kroužků, sportovních klubů a pořádají se různé schůze, na kterých se většinou píše docházka všech členů. Většina těchto zápisů bývá dodnes psána na papír či jinými podobnými způsoby. Toto může být z hlediska přehlednosti i z dlouhodobého hlediska nepříznivé, jelikož papír lze lehce poškodit, ať už vlivem špatného zacházení člověka, či dokonce vlivem slunečního záření a jiných přírodních vlivů. Abychom předešli různým komplikacím, můžeme využít mobilní aplikaci pro docházku, ke které nám bude stačit pouze mobilní telefon. Tento způsob zápisu docházky může být pro značnou část uživatelů vyhovující a více přehledný.

V aplikaci bude efektivně zpracovaný seznam kroužků, které si přihlášený uživatel sám přidá a nastaví či později upraví jejich datum a čas konání. Ve zvoleném kroužku si zaznačí, kdo se jej účastní či neúčastní, popřípadě je nemocný nebo jinak omluvený. Každý účastník bude mít u uživatele aplikace svůj profil, ve kterém si bude zapisovat kontaktní údaje na účastníka samotného, kontakt na rodiče, potvrzení zdravotního stavu, souhlas s podmínkami GDPR a další poznámky. Další položky bude snadné přidat. V aplikaci půjde přidávat hodiny/lekce a efektivní formou zadávat docházku přímo v hodině/lekci. Hodiny/lekce budou v seznamu a bude možné řazení seznamu podle jména, podle příjmení nebo podle toho, zda je účastník přítomen či není. V docházce budou zaznačeni účastníci, se kterými je potřeba cokoliv řešit, například dodání přihlášky, potvrzení o GDPR, platba nebo jiné nevyřízené věci. Tato funkce by měla jít vypnout v nastavení. Aplikace bude podporovat export docházky do vybrané aplikace jako je například Excel nebo formát CSV. Aplikace bude mít možnost přístupu více uživatelských účtů, které budou mít přidělené oprávnění pro zápis či pouze pro čtení. Aplikace bude umožňovat funkci zápisu dat off-line z různých zařízení a to například způsobem, že se databáze bude duplikovat lokálně i v cloudu.

Aplikace bude navržena pro velice jednoduché ovládání, aby byla uživatelsky přívětivá pro všechny uživatele a aby se nemusely veškeré věci zdlouhavě hledat. Aplikace bude vytvořena pro platformy Android i iOS. V aplikaci bude jako navigace sloužit spodní lišta, ze které se uživatel dostane do přehledu všech kroužků, kde bude moci ze seznamu vybrat příslušný kroužek a zapsat do něj docházku. Dále se dostane na stránku se všemi uživateli, které si vytvořil nebo nahrál z externího zdroje – databáze. Poslední stránka z odkazu na spodní liště bude profil přihlášeného uživatele, kde si na svém profilu bude moci nastavit jméno, příjmení, heslo, fotku, datum narození apod. Na této stránce přihlášený uživatel najde ikonu ozubeného kolečka, kde si bude nastavovat různé možnosti aplikace.

# Uživatelská dokumentace

## Instalace

1. Instalační soubor dochazka.apk zkopírujte do telefonu.
2. Spusťte instalační soubor.
3. Pokud Vás telefon upozorní na nějaké bezpečnostní riziko nebo že aplikace není ověřena, tak pokračujte dále.
4. Pokud váš telefon zakazuje instalaci aplikací z neznámých zdrojů, přejděte do Nastavení>> Zabezpečení>> Povolit instalaci aplikací z neznámých zdrojů. Je možné, že Vás tam telefon přesune automaticky. Poté prosím tuto funkci povolte.
5. Pokud bude telefon potřebovat povolený přístup k uložišti, tak tuto možnost povolte.
6. Po instalaci se aplikace zobrazí na poslední stránce Plochy nebo v přehledu aplikací.

## Popis aplikace

Tato aplikace, fungující na platformách Android, slouží pro efektivní zápis docházky do kroužku či lekce. Je určena pro vedoucí kroužků, kteří si chtějí ulehčit zapisování docházky.

Uživateli umožňuje jednoduše a efektivně zapsat docházku v aplikaci, kterou má stále u sebe ve svém mobilním telefonu. Další funkcí této aplikace je zpětné přidávání docházky, přidávání lidí do kroužku, editace jejich údajů, filtrování lidí, kteří ještě něco nesplnili.

Celá aplikace funguje online, takže je dostupná odkudkoliv s připojením na internet.

Postup instalace je popsán v samostatné kapitole. Aplikace funguje na systému Android 4.4 a vyšší. Instalační soubory naleznete v příloze.

## Spuštění aplikace

Při spuštění aplikace se jako první zobrazí přihlašovací obrazovka s tlačítkem pro přesun na formulář pro registraci.

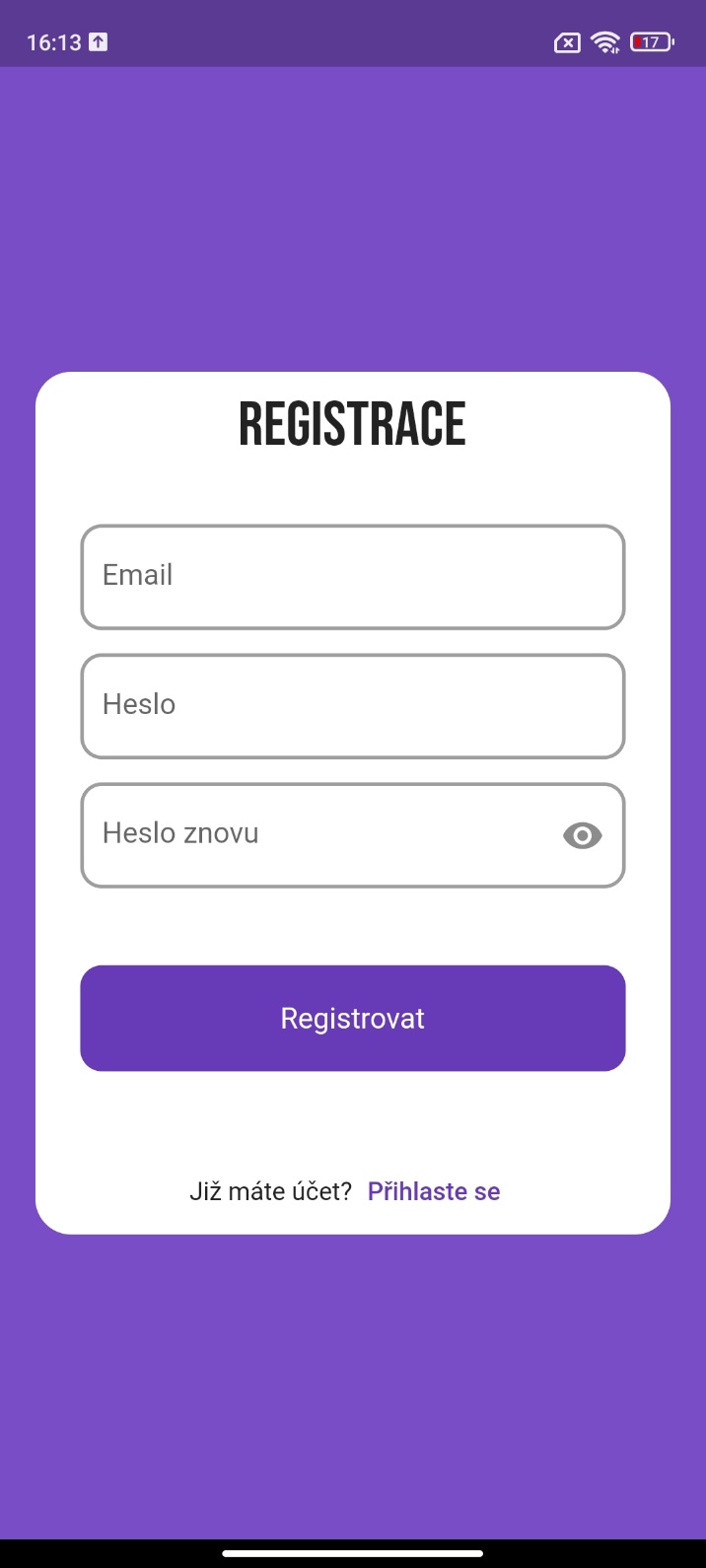


Obrázek 1 Obrazovka přihlášení

**Zdroj: vlastní**

## Obrazovka pro registraci

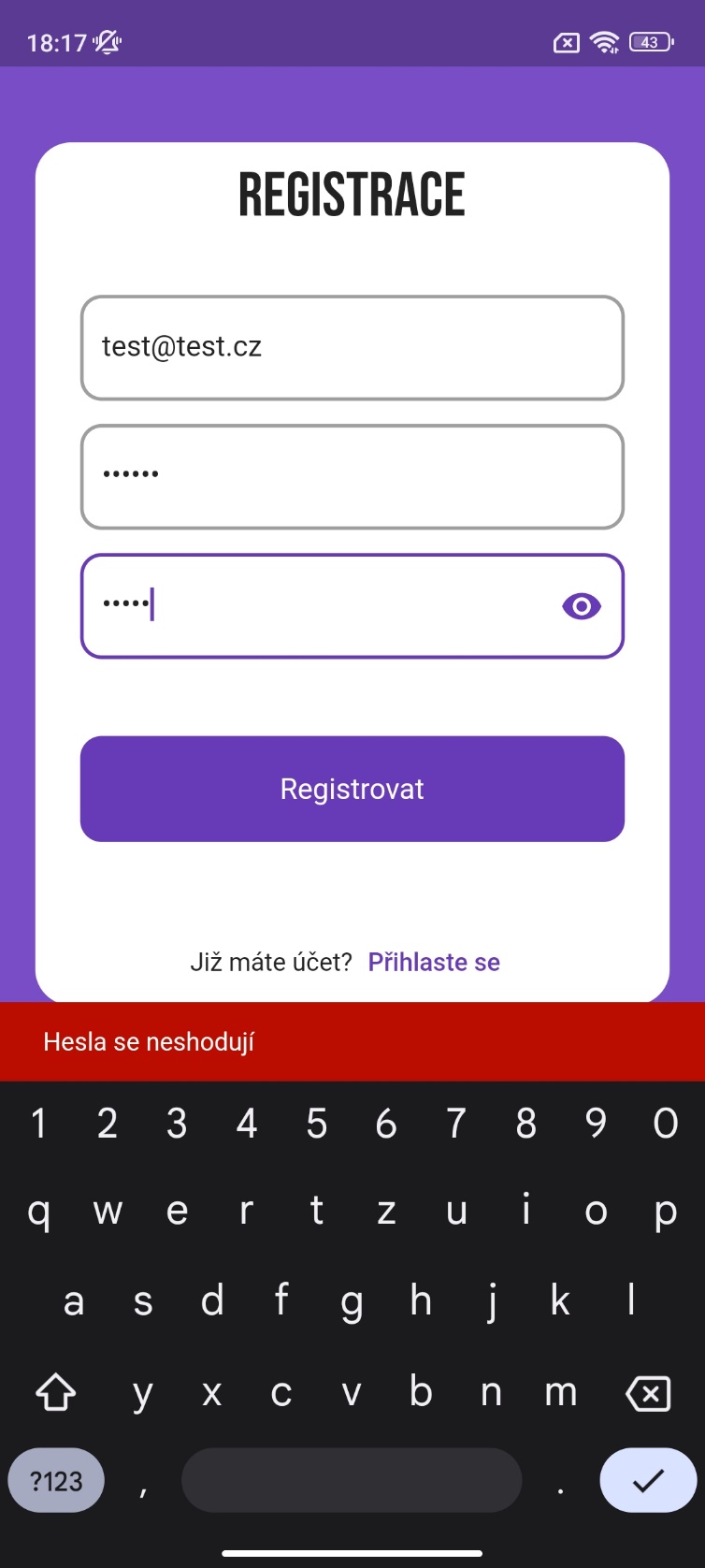
Po kliknutí na tlačítko *Registrujte se* se zobrazí formulář pro registraci. Uživatel si zde zadá svůj email pro přihlášení, heslo a stejné heslo znovu pro potvrzení hesla.



Obrázek 2 Obrazovka registrace

**Zdroj: vlastní**

Pokud uživatel zadá odlišné heslo, upozorní ho na to hláška ve spodní části obrazovky.



Obrázek 3 Chybné heslo

**Zdroj: vlastní**

## Úvodní stránce

Po úspěšném přihlášení se na úvodní stránce se v horní části nachází lišta, ve které se nachází text s názvem vybrané lekce, nebo pokud uživatel ještě lekci vybranou nemá, zobrazí se text *Lekce není vybraná*. Po pravé straně textu se nachází ikona pro přesun do seznamu žáků a ikona menu. V dolní částí se nachází tlačítko pro přidání nového dne. Pokud lekce není vybraná, objeví se v tlačítku ikona křížku, když následně lekci vybereme, objeví se v tlačítku ikona +.



Obrázek 4 Úvodní stránka

**Zdroj: vlastní**

## Menu na úvodní stránce

V menu nalezneme tři položky. První slouží pro přechod na stránku pro vybrání či vytvoření lekce, druhá pro zpětné přidání data do lekce a třetí slouží pro odhlášení účtu z aplikace.



Obrázek 5 Menu úvodní obrazovky

**Zdroj: vlastní**

## Stránka výběru a přidání lekce

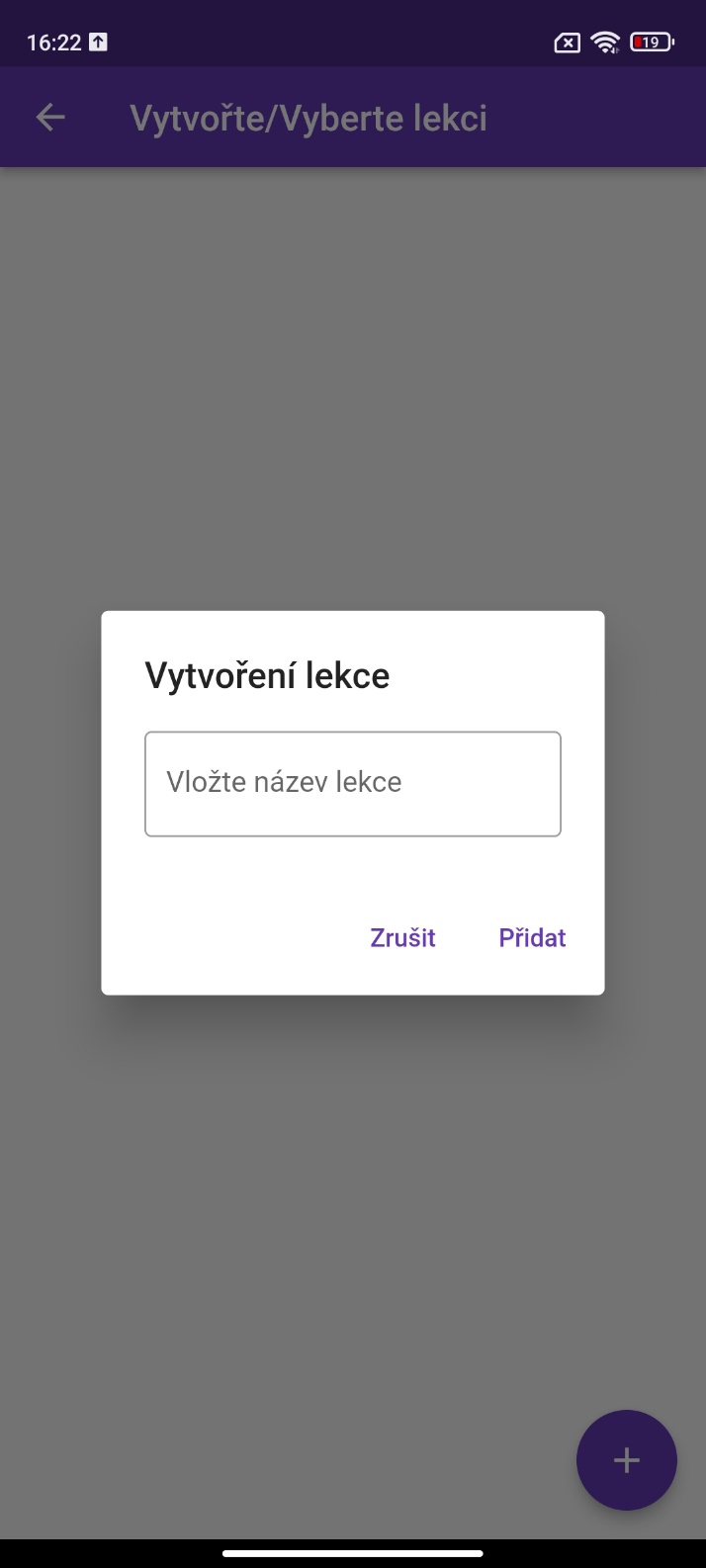
Na stránce pro vybrání či vytvoření lekce se v horní části nachází lišta s textem *Vytvořte/Vyberte lekci.* Ve spodní části se nachází tlačítko pro přidání nové lekce. Pokud uživatel dosud nemá žádnou lekci vytvořenou, objeví se pouze prázdný obsah stránky.



Obrázek 6 Stránka pro výběr/vytvoření lekce

**Zdroj: vlastní**

Pokud chce uživatel vytvořit novou lekci, musí kliknout na tlačítko přidání a objeví se mu dialogové okno, kde musí zadat název lekce. Dialogové okno obsahuje textové pole s textem *Vložte název lekce* a pod textovým polem dvě textová tlačítka *Zrušit* a *Přidat.* Tlačítko *Zrušit* zavře dialogové okno bez ukládání vloženého textu. Tlačítko *Přidat* uloží vložený název do textového pole a zavře dialogové okno.



Obrázek 7 Dialogové okno pro vytvoření lekce

**Zdroj: vlastní**

Po vytvoření nové lekce se objeví na stránce karta s uloženým názvem. Karta obsahuje v levé části název lekce a v pravé části šipku, kterou uživatel vybere lekci, do které chce zapisovat.



Obrázek 8 Vytvořená lekce

**Zdroj: vlastní**

Pokud uživatel vytvořil více lekcí, tak se karty budou řadit abecedně pod sebe.



Obrázek 9 Více lekcí

**Zdroj: vlastní**

## Vytvoření nového data

Po vybrání lekce, do které chce uživatel zapisovat, se aplikace vrátí na úvodní stránku a název vybrané lekce se zobrazí v horní liště. Pokud uživatel lekci teprve vytvořil, bude stránka prázdná, jelikož k vytvořenému kroužku nejsou přidána žádná data.



Obrázek 10 Hlavní stránka s vybranou lekcí

**Zdroj: vlastní**

Pokud už uživatel lekci vytvořenou měl a přidával do ní data, objeví se mu karty s vytvořenými daty. Po kliknutí na tlačítko plus se vytvoří nové aktuální datum.

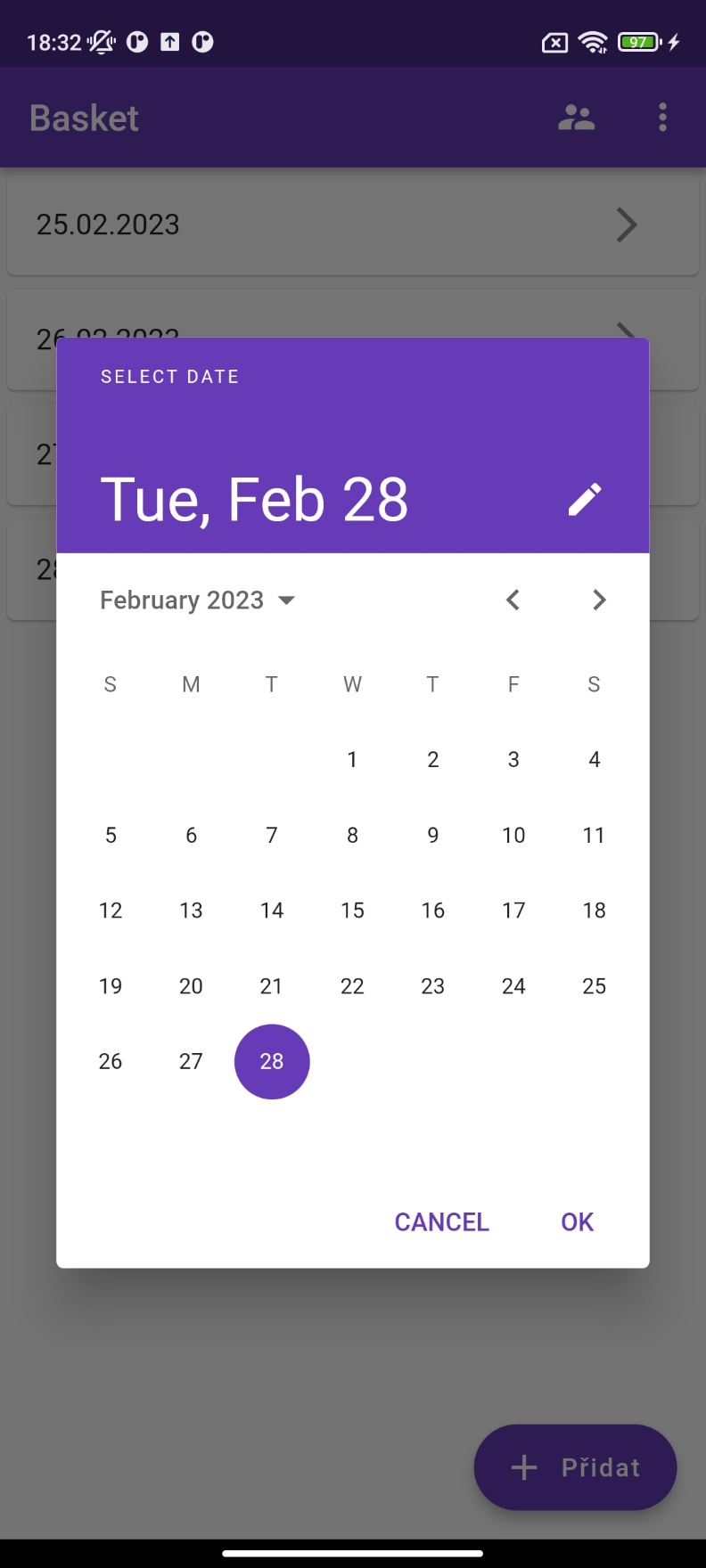
Obsah obrázku text, snímek obrazovky, spotřebič

Popis byl vytvořen automaticky

Obrázek 11 Vytvořené datum

**Zdroj: vlastní**

Pokud chce uživatel přidat datum zpětně, v horním menu vybere danou položku a otevře se mu kalendář, ve kterém si datum vybere. Tlačítkem *OK* uživatel potvrdí svůj výběr a vytvoří se nové datum. Tlačítkem *Cancel* se zavře kalendář a žádné datum se nevytvoří.



Obrázek 12 Kalendář pro zpětné vytvoření data

**Zdroj: vlastní**

## Vytvoření žáci

Po kliknutí na ikonu *Lidé* se otevře stránka se seznamem vytvořených žáků. V horní liště stránky se nachází z leva ikona pro filtrování dle vybraných parametrů, ikona + pro přidání nového žáka a ikona pro návrat na úvodní stránku. Pokud uživatel nemá vytvořený žádný záznam žáka, bude obsah stránky prázdný.



Obrázek 13 Prázdný seznam lidí

**Zdroj: vlastní**

Pokud již uživatel lidi vytvořené má, objeví se na kartách, které obsahují jméno a příjmení a pod ním název lekce, do které lidé chodí. V pravé části každé karty se nachází ikona pro editaci údajů člověka.



Obrázek 14 Vytvořený člověk

**Zdroj: vlastní**

Pro výpis veškerých údajů o daném člověku stačí kliknout na kartu a objeví se ze spodní části obrazovky karta, která obsahuje jméno, příjmení, bydliště, kontakt 1, kontakt 2, název kroužku, který daný člověk navštěvuje, zaplacenou částku, souhlas s GDPR a případně doplňující poznámku. Pro zavření stačí kliknout kdekoliv do prostoru mimo otevřenou kartu.

Obsah obrázku stůl

Popis byl vytvořen automaticky

Obrázek 15 Veškeré informace o člověku

**Zdroj: vlastní**

Pokud některé informace u žáka chybí, na kartě se u každé nedoplněné informace zobrazí text *Nedoplněno*, který je zbarven do červena. Pokud žák nemá potvrzeno GDPR, zobrazí se červený text *Zakázáno*.

Obsah obrázku stůl

Popis byl vytvořen automaticky

Obrázek 16 Nedoplněné informace o člověku

**Zdroj: vlastní**

### Filtrování seznamu žáků

Uživatel může filtrovat hodnoty vytvořených žáků, např. zda dosud nemá souhlas s GDPR nebo pokud ještě neuhradil částku. Položka *Vše* zobrazí seznam všech žáků.

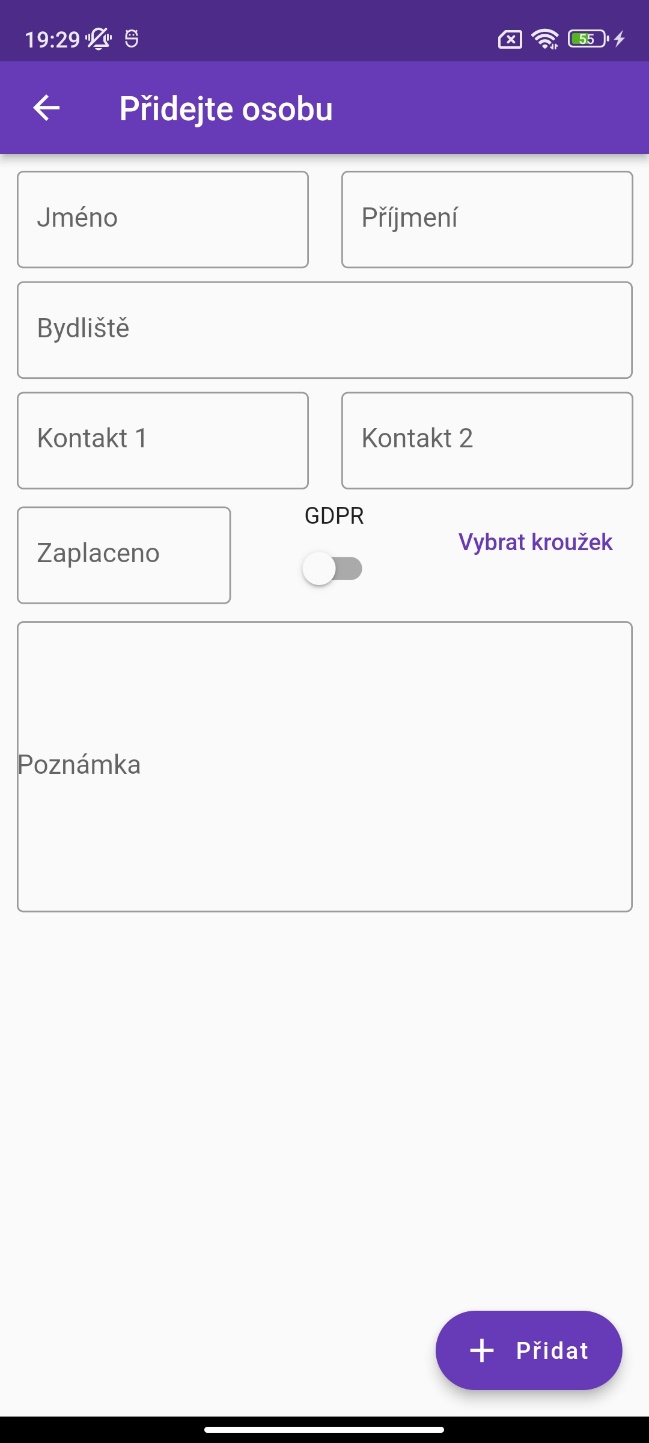


Obrázek 17 Menu pro filtrování

**Zdroj: vlastní**

## Přidávání žáka

Pro vytvoření nového žáka klikneme na stránce se seznamem lidí na ikonu + v horní liště. Otevře se nám stránka s textovými poli, které v sobě mají napsáno, co do nich uživatel musí vložit. Formulář obsahuje pole pro vložení jména, příjmení, bydliště, kontakt 1, kontakt 2, zaplacenou částku, tlačítko pro potvrzení GDPR, tlačítko pro výběr kroužku, které otevře dialogové okno se seznamem uživatelem vytvořených kroužků a pole pro přidání poznámky. Pro vytvoření záznamu žáka se v levé dolní části obrazovky nachází tlačítko *+ Přidat,* které přidá žáka do seznamu.



Obrázek 18 Formulář pro přidání člověka

**Zdroj: vlastní**

### Podmínky vytvoření žáka

Pokud uživatel nezadá jméno a bude chtít přidat žáka, aplikace vypíše chybovou hlášku, že textové pole *jméno* nesmí zůstat prázdné.

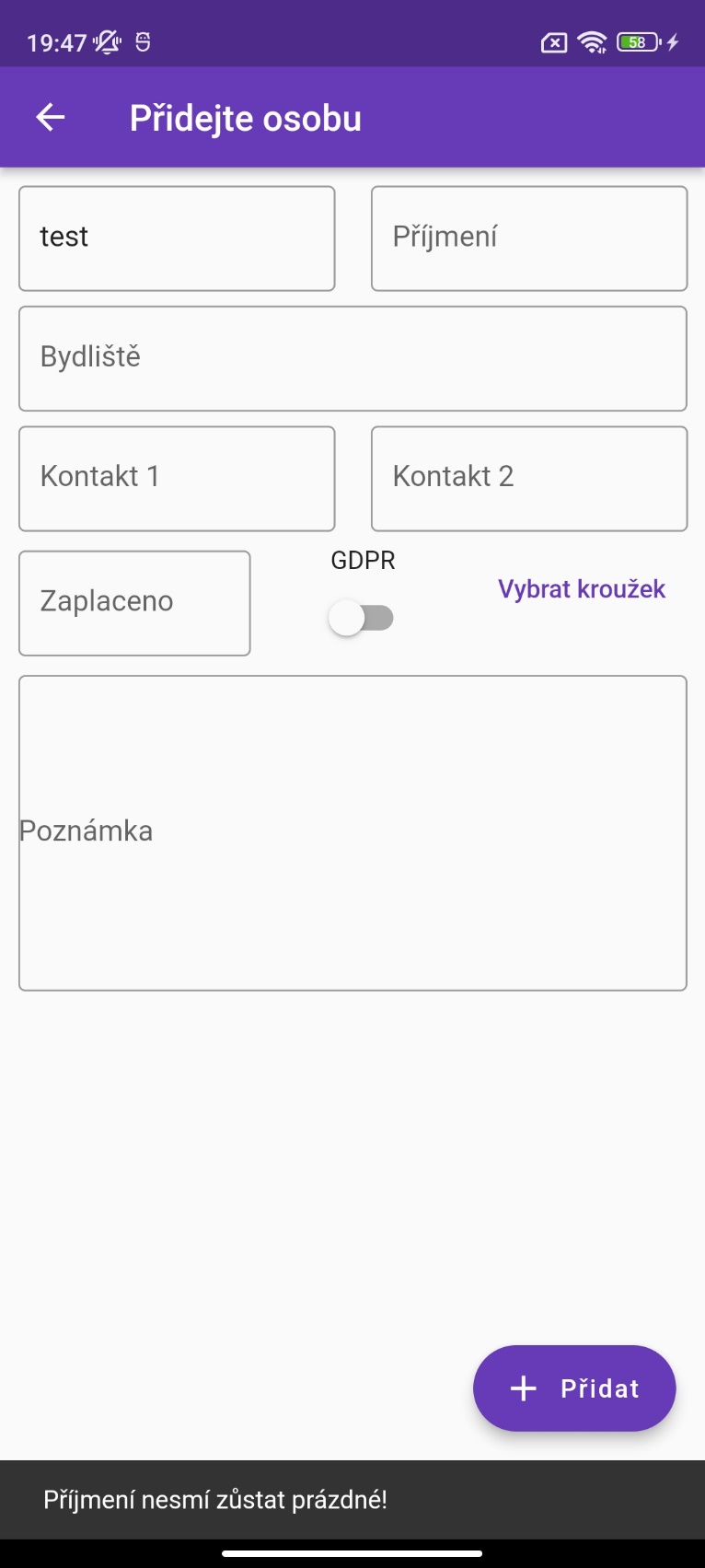
Obsah obrázku text

Popis byl vytvořen automaticky

Obrázek 19 Hláška pro nedoplněné jméno

**Zdroj: vlastní**

Pokud uživatel nevloží příjmení, aplikace vypíše chybovou hlášku, že textové pole *příjmení* nesmí zůstat prázdné.

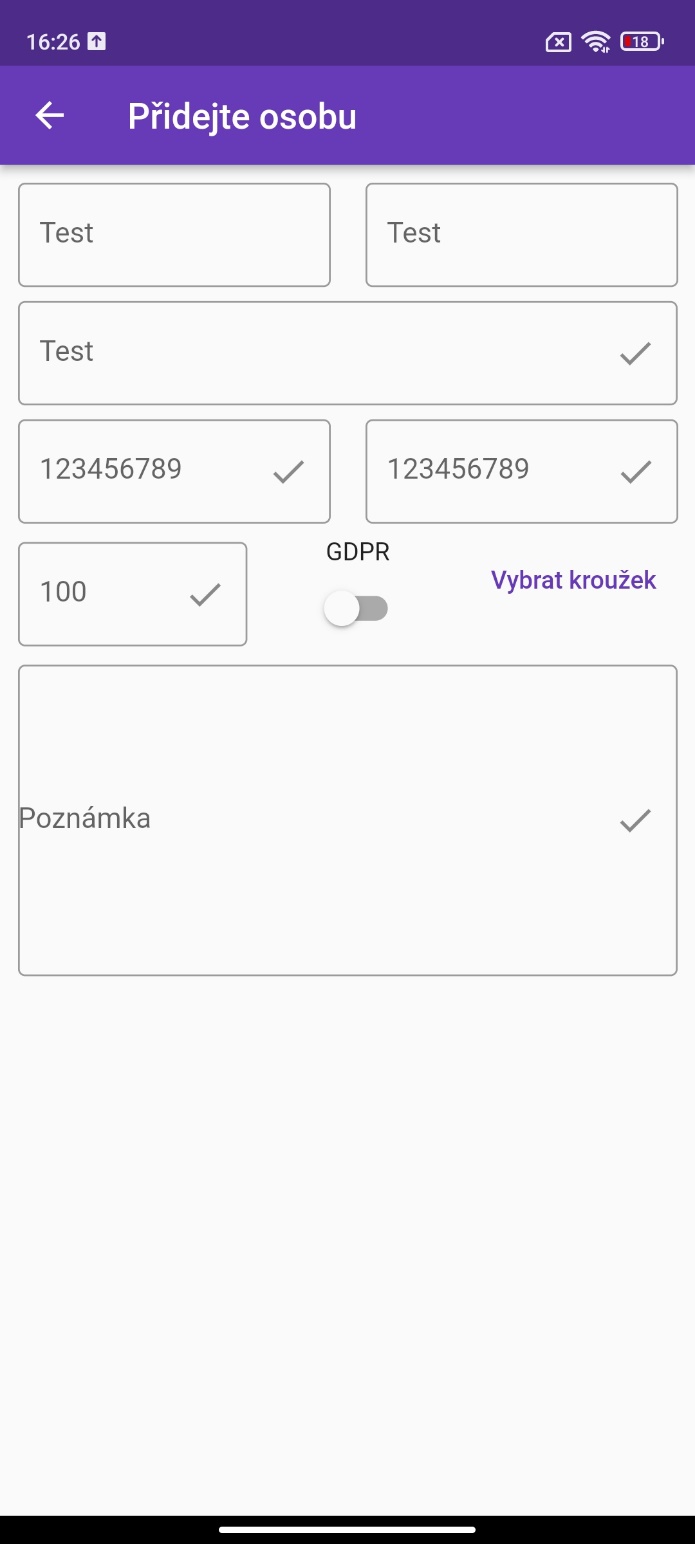


Obrázek 20 Hláška pro nedoplněné příjmení

**Zdroj: vlastní**

## Editace žáka

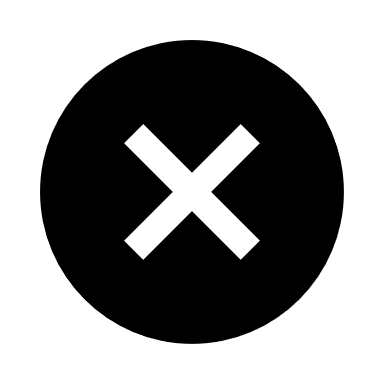
Editaci informací žáka uživatel otevře pomocí ikony na pravé straně karty žáka. Otevře se stránka podobná vytváření nového žáka, ale s rozdílem, že v každém textovém poli se nachází ikona pro upravení informace. V poli se nachází již vytvořená informace. Uživatel může upravovat všechno kromě jména a příjmení.



Obrázek 21 Formulář pro editaci člověka

**Zdroj: vlastní**

## Zápis docházky

Pro zápis docházky uživatel vybere vytvořené datum, do kterého bude chtít zapisovat. Pro vybrání konkrétního data uživatel klikne na šipku v pravé části karty. Po vybrání se uživateli otevře stránka se seznamem lidí, které uživatel přidal do dané lekce. Každý žák má potom na pravé straně dvě ikony. Ikona  slouží pro potvrzení docházky a ikona  slouží pro zapsání, že daný žák chybí. Po rozřazení si uživatel může zkontrolovat – nabídkou v horní liště - zda je žák přítomen nebo není. Volbu je možno upravit. Při každém zápisu se ve spodní části objeví hláška se zelenou barvou, že daný žák je přítomen, nebo hláška s červenou, že daný žák není přítomen.



Obrázek 22 Zápis docházky

**Zdroj: vlastní**

Pokud uživatel zapsal žákovi, že je přítomen, objeví se v příslušném seznamu přítomných. Pro změnu docházky žáka je na pravé straně karty daného žáka ikona pro přesunutí do stavu nepřítomen.



Obrázek 23 Zapsání přítomnosti s hláškou

**Zdroj: vlastní**

Pokud uživatel zapsal žákovi, že je nepřítomen, objeví se v příslušném seznamu nepřítomných. Pro změnu docházky žáka je na pravé straně karty daného žáka ikona pro přesunutí do stavu přítomen.



Obrázek 24 Zapsání nepřítomnosti s hláškou

**Zdroj: vlastní**

# Programátorská dokumentace

## Firebase

Firebase je platforma poskytovaná společností Google, která umožňuje vývojářům rychle vytvářet mobilní i webové aplikace s cloudovými funkcemi. Firebase nabízí celou řadu služeb, včetně autentifikace uživatelů, databází v reálném čase, cloudového uložiště, zasílání zpráv, analýzy chování uživatelů a mnoho dalšího.

Firebase poskytuje také nástroje pro rychlé a jednoduché vytváření serverových funkcí, které umožňují vývojářům psát kód pro zpracování událostí v aplikaci (např. přidání nového uživatele nebo vytvoření nového objektu v databázi), které jsou spouštěny na serveru Firebase.

Firebase je vhodné pro vývojáře mobilních i webových aplikací, kteří chtějí rychle a efektivně vytvářet aplikace s pokročilými funkcemi, které jsou propojené s cloudovými službami a umožňují zpracování uživatelského chování. Firebase je také výhodné pro malé a střední projekty, protože nabízí bezplatný tarif s omezenými funkcemi a poměrně nízkými náklady na další tarify.

### Použité pluginy

Pluginy pro Flutter slouží k rozšíření funkcionality aplikace o externí knihovny a služby. Flutter je framework, který umožňuje vytvářet multiplatformní mobilní aplikace pro iOS a Android, ale také i pro desktopové a webové aplikace. Nicméně, Flutter nemá v sobě všechny funkcionality, které by vývojáři mohli potřebovat, proto pluginy umožňují přidání nových funkcí, jako jsou například přístup k fotoaparátu, geolokace, push notifikace, přehrávání zvuku a mnoho dalších.

|  |  |
| --- | --- |
| **firebase\_auth** | Slouží pro implementaci autentizace uživatelů do aplikace, která využívá Firebase jako backend službu. |
| **cloud\_firestore** | Slouží pro komunikaci mezi aplikací a databází Firestore |
| **intl** | Slouží k formátování čísel, dat, měn a překládání textu |
| **google\_fonts** | Umožňuje používat písma z Google Fonts |

Tabulka 1 Pluginy

**Zdroj: vlastní**

### Implementace pluginů

Všechny pluginy se implementují v souboru *pubspec.yaml.*

Obsah obrázku text

Popis byl vytvořen automaticky

Obrázek 25 Pluginy

**Zdroj: vlastní**

Po implementaci pluginů se vybrané volají na každou stránku aplikace zvlášť. Plugin *materiál.dart* obsahuje všechny informace o widgetech ve Flutteru a je již standartně předinstalovaný ve frameworku.

Obsah obrázku text

Popis byl vytvořen automaticky

Obrázek 26 Volání pluginů

**Zdroj: vlastní**

### Použité Firebase služby

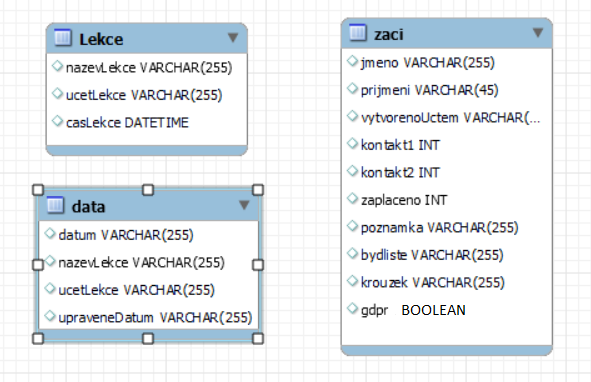
|  |  |
| --- | --- |
| **Firestore Database** | poskytuje NoSQL dokumentovou databázi s funkcemi ukládání a synchronizace dat v reálném čase |
| **Authentication** | umožňuje uživatelům přihlásit se pomocí různých metod, jako jsou například e-mailové adresy, telefonní čísla, Google účty, Facebook účty a další. |

Tabulka 2 Firebase služby

**Zdroj: vlastní**

## Databáze

Databáze Firestore je noSQL, tudíž neobsahuje žádné relace mezi tabulkami. Databáze obsahuje 3 tabulky, které ukládají data jakožto dokumenty. V ER diagramu můžeme vidět, co která tabulka obsahuje.



Obrázek 27 Diagram databáze

**Zdroj: vlastní**

## Metody

Po spuštění aplikace je volána metoda *main,* která zavolá metodu *WidgetsFlutterBinding.ensureInitialized(),* která inicializuje vazbu mezi widgety a enginem ve Flutteru. Následně se zavolá asynchronní metoda *Firebase.initializeApp(),* která inicializuje Firebase. Po této metodě můžeme začít používat služby Firebase.

Obsah obrázku text

Popis byl vytvořen automaticky

Obrázek 28 Metoda Main

**Zdroj: vlastní**

### Metoda *Prihlasovani*

Tato metoda odesílá údaje zadané uživatelem do služby Firebase Authentication. Jelikož se metoda volá později než obsah stránky, je definována jako asynchronní. Údaje se ukládají do stringu *email* a *password.*

Obsah obrázku text

Popis byl vytvořen automaticky

Obrázek 29 Metoda *Prihlasovani*

**Zdroj: vlastní**

### Metoda *Dispose* u přihlášení

Tato metoda zastaví asynchronní proces vkládání textu do controlleru pro email a heslo potom, co je stisknuto tlačítko *Přihlásit*.

Obsah obrázku text

Popis byl vytvořen automaticky

Obrázek 30 Metoda *Dispose* u příhlášení

**Zdroj: vlastní**

### Metoda *Dispose* u registrace

Tato metoda zastaví asynchronní proces vkládání textu do controlleru pro email, heslo a metodu pro kontrolu hesla, zda jsou obě stejná potom, co je stisknuto tlačítko *Registrovat*.

Obsah obrázku text

Popis byl vytvořen automaticky

Obrázek 31 Metoda *Dispose* u registrace

**Zdroj: vlastní**

¨

### Metoda *Registrace*

Tato metoda vytváří uživatele pomocí údajů zadané uživatelem do formuláře pro registraci. Jelikož se metoda volá později než obsah stránky, je definována jako asynchronní. Údaje se ukládají do stringu *email* a *password.* Tato metoda obsahuje podmínku, která provede svůj obsah na základě hodnoty, kterou vrátí metoda *hesloSpravne.*

Obsah obrázku text

Popis byl vytvořen automaticky

Obrázek 32 Metoda *Registrace*

**Zdroj: vlastní**

### Metoda \_*ukazKalendar*

Tato metoda zobrazí widget *showDatePicker,* který následně slouží k vybrání data pro zapsání docházky zpětně a následné uložení data do databáze.

Obsah obrázku text

Popis byl vytvořen automaticky

Obrázek 33 Metoda \_*ukazKalendar*

**Zdroj: vlastní**

### Metoda *hesloSpravne*

Metoda kontroluje, zda jsou obě hesla ve formuláři pro registraci stejná. Pokud ne, aplikace vypíše hlášku, že se hesla neshodují. Metoda vrátí hodnotu true či false.

Obsah obrázku text

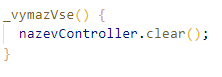
Popis byl vytvořen automaticky

Obrázek 34 Metoda *hesloSpravne*

**Zdroj: vlastní**

### Metoda *\_vymazVse*

Tato metoda slouží pro vymazání obsahu controlleru, do kterého se ukládá název lekce zadané uživatelem.



Obrázek 35 Metoda *\_vymazVse*

**Zdroj: vlastní**

## Třídy

### Třída *Uvodni*

Tato třída obsahuje základní nastavení pro celou aplikaci jako barvu widgetů v aplikaci, třídu, která se zavolá po spuštění aplikace a skrytí štítku debug v pravém horním rohu.

Obsah obrázku text

Popis byl vytvořen automaticky

Obrázek 36 Třída *Uvodni*

**Zdroj: vlastní**

### Třída *Cesty*

Třída *Cesty* rozhoduje, která třída se bude volat v závislosti na tom, zda je uživatel přihlášen. Třída pošle dotaz do Firebase. Pokud má metoda *StreamBuilder* data od Firebase že je uživatel přihlášen, zavolá třídu *HlavniStranka* a nastaví proměnnou *nazevLekce* na „Lekce není vybraná“*,* pokud není přihlášen, zavolá třídu *Autorizacni.*

Obsah obrázku text

Popis byl vytvořen automaticky

Obrázek 37 Třída *Cesty*

**Zdroj: vlastní**

### Třída *Autorizacni*

Třída *Autorizacni* slouží pro zobrazování stránek přihlašovacího a registračního formuláře. Hodnota boolean určuje, která stránka s třídou se zobrazí. Pokud se hodnota boolean rovná true, zobrazí se přihlašování. Toto je nastaveno jako výchozí. Pokud se ale boolean rovná false, zobrazí se registrace. Metoda *prehodStranky* pak udělá negaci hodnoty boolean.

Obsah obrázku text

Popis byl vytvořen automaticky

Obrázek 38 Třída *Autorizacni*

**Zdroj: vlastní**

### Třída *Prihlaseni*

Tato třída obsahuje formulář pro přihlášení uživatele, barvu pro pozadí obsahu stránky a odkaz pro přesunutí uživatele na stránku registrace. Hodnotu, kterou uživatel vložil, ukládá do příslušného controlleru.

Obsah obrázku text

Popis byl vytvořen automaticky

Obrázek 39 Widget – Textové pole pro heslo

**Zdroj: vlastní**

Tento widget obsahuje ikonu pro skrytí a odhalení hesla v textovém poli. Hodnotu, kterou uživatel vložil, ukládá do příslušného controlleru.



Obrázek 40 Widget – Textové pole pro heslo

**Zdroj: vlastní**

Tento widget slouží jako tlačítko pro přihlášení uživatele, který při kliknutí volá metodu pro přihlášení.

Obsah obrázku text

Popis byl vytvořen automaticky

Obrázek 41 Widget – Tlačítko pro přihlášení

**Zdroj: vlastní**

Widget *Row* obsahuje text a odkaz na stránku pro registraci.

Obsah obrázku text

Popis byl vytvořen automaticky

Obrázek 42 Widget – Odkaz na stránku registrace

**Zdroj: vlastní**

### Třída *Registrace*

Třída obsahuje formulář se třemi textovými poli pro vložení emailu a dvou shodných hesel.

Tento widget obsahuje textové pole pro vložení emailu. Hodnotu, kterou uživatel vložil, ukládá do příslušného controlleru.

Obsah obrázku text

Popis byl vytvořen automaticky

Obrázek 43 Widget – Textové pole pro registraci emailem

**Zdroj: vlastní**

Tento widget obsahuje formulář pro vložení hesla s ikonou pro jeho skrytí a odhalení.



Obrázek 44 Widget – Textové pole pro heslo s možností ukázání hesla

**Zdroj: vlastní**

### Třída *HlavniStranka*

Tato třída se uživateli zobrazí, pokud je již přihlášen. Pokud má vybraný kroužek, objeví se mu na stránce již přidaná data. Obsahuje horní navigační lištu, builder, který vypisuje záznamy z databáze a tlačítko ve spodní části které přidá nové aktuální datum, které přidá záznam do databáze.

Tlačítko v navigační liště, které uživatele přesměruje na stránku se seznamem přidaných lidí.

Obsah obrázku text

Popis byl vytvořen automaticky

Obrázek 45 Widget - Tlačítko pro volání třídy *SeznamLidi*

**Zdroj: vlastní**

Tlačítko pro otevření menu, které obsahuje odkaz na stránku pro vybrání či vytvoření lekce, přidání data zpětně a odhlášení uživatele.



Obrázek 46 Widget – Menu na hlavní stránce

**Zdroj: vlastní**

Widget *StreamBuilder* zobrazuje data z Firestore databáze v předdefinovaném widgetu, v našem případě *Card*. Ve *StreamBuilderu* si potom vytvoříme proměnnou, do které se budou ukládat jednotlivá data z databáze „*nejakaData“,*  a ty poté každé zobrazíme pod sebou a budou se vypisovat hodnoty, které jsou v databázi pod vybraným názvem proměnné.



Obrázek 47 Widget – StreamBuilder s výpisem dat

**Zdroj: vlastní**

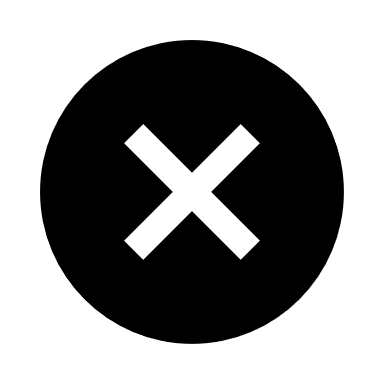
Do proměnné v hlavičce třídy nastavíme, ze které kolekce se budou vypisovat data a popř. nastavíme filtrování určitých hodnot.

Obsah obrázku text

Popis byl vytvořen automaticky

Obrázek 48 Proměnná pro výpis dat z databáze

**Zdroj: vlastní**

Tlačítko pro přidání nového data mění svoji ikonu v závislosti na vybrané lekci. Pokud lekce není vybrána, ukáže se v tlačítku . Pokud je lekce vybrána, ukáže se v tlačítku + a po stisknutí se uloží do databáze nový záznam s aktuálním datem.

Obsah obrázku text

Popis byl vytvořen automaticky

Obrázek 49 Widget – Tlačítko pro vytváření nového data

**Zdroj: vlastní**

### Třída *VytvoreniLekce*

Třída obsahuje navigační lištu, *StreamBuilder* pro vypisování vytvořených kroužků uživatelem a tlačítko pro přidání nového kroužku. Zde je výpis filtrován dle uživatele, který své lekce vytvořil.

Obsah obrázku text

Popis byl vytvořen automaticky

Obrázek 50 Proměnná pro výpis dat z databáze s filtrem podle uživatele

**Zdroj: vlastní**

Po stisknutí tlačítka přidání se na obrazovce objeví dialogové okno s textovým polem a tlačítky pro zavření dialogového pole nebo uložení zadaných hodnot do databáze a následné zavření okna.

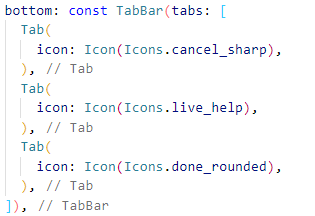


Obrázek 51 Widget – Okno pro vytvoření nové lekce

**Zdroj: vlastní**

### Třída *ZapsaniDochazky*

Tato třída zobrazuje seznam žáků v daném kroužku a ke každému se po kliknutí na stav docházky každého z nich nastaví v databázi stav přítomen či nepřítomen. V menu v navigační liště si poté může uživatel překliknout na seznam, kdo v daném kroužku je a kdo ne. Uprostřed lišty je záložka, kde se zobrazí všichni žáci a uživatel je v této záložce rozřazuje. Pokud uživatel udělá chybu, může jak ve zmíněném seznamu, tak v seznamu s již přiřazenou docházkou změnit stav docházky každého žáka.



Obrázek 52 Widget – Spodní menu navigační lišty

**Zdroj: vlastní**

### Třída *SeznamLidi*

Třída obsahuje navigační lištu s filtry pro zobrazení žáků, kterých chybí informace ohledně GDPR a zda mají už zaplaceno.

Tento widget obsahuje menu s položkami pro filtrování. Zvolená položka otevře stránku s kartami lidí, kteří zvolený filtr nesplňují.



Obrázek 53 Widget – Menu pro vybrání filtrování nedoplněné hodnoty

**Zdroj: vlastní**

Pokud chce uživatel zobrazit veškeré informace o žákovi, stačí kliknout na vybraného žáka a otevře se karta s informacemi. Pokud je některá hodnota prázdná, místo informace se vypíše červeně podbarvený text „*Nedoplněno“*. Pokud má žák povolené GDPR, tak bude zbarveno zeleně.



Obrázek 54 Widget – Karta s výpisem všech informací o uživateli

**Zdroj: vlastní**

U každého žáka je na pravé straně karty ikona, která uživatele přesměruje na formulář pro editaci uživatele. S přesměrováním se na stránku odešlou i data k danému uživateli.

Obsah obrázku text

Popis byl vytvořen automaticky

Obrázek 55 Data odeslaná pro editaci

**Zdroj: vlastní**

Na samotné kartě žáka se nachází jeho jméno a příjmení a pod ním se nachází název kroužku, který navštěvuje.



Obrázek 56 Obsah karty s žákem

**Zdroj: vlastní**

### Třída *PridatLidi*

Tato třída obsahuje formulář, který obsahuje textové pole pro zapsání jména, příjmení, bydliště, kontaktu 1, kontaktu 2, částky, zda zaplatil, zda potvrdil GDPR, tlačítko pro výběr kroužku a pole pro případnou poznámku.

Například kód pro textové pole na vložení bydliště žáka vypadá následovně.

Obsah obrázku text

Popis byl vytvořen automaticky

Obrázek 57 Widget – Textové pole pro vložení adresy bydliště žáka

**Zdroj: vlastní**

Pro nastavení hodnoty souhlasu GDPR slouží přepínač. Na začátku je nastaven na false a po přepnutí se nastaví na true a podbarví se zeleně.

Obsah obrázku text

Popis byl vytvořen automaticky

Obrázek 58 Widget – Tlačítko pro potvrzení GDPR

**Zdroj: vlastní**

V tlačítku pro odesílání dat do databáze se nachází podmínka, která nedovolí uživateli přidat data bez vyplnění políčka se jménem a příjmením. Pokud uživatel některé pole vynechá, vypíše se hláška s daným problémem. Když má obě tato políčka vyplněná, odešlou se data do databáze.

Obsah obrázku text

Popis byl vytvořen automaticky

Obrázek 59 Podmínky v tlačítku pro odeslání dat do databáze

**Zdroj: vlastní**

### Třída *Editace*

Tato třída je podobná na třídu pro vytvoření nového žáka, ale s rozdílem, že u každého pole s výjimkou jména a příjmení je ikona, která odešle nově vložený text a přepíše předešlý. Předešlý text je vepsaný v textovém poli hned po otevření stránky.



Obrázek 60 Widget – Textové pole pro editaci informace o žakovi

**Zdroj: vlastní**

### Třída *Nezaplaceni*

Tato třída obsahuje *StreamBuilder*, který filtruje a zobrazuje data, kde se v databázi u nich rovná proměnná zaplaceno na 0.

### Třída *NeniGDPR*

Třída obsahuje *StreamBuilder*, který filtruje a zobrazuje data, kde se v databázi u nich rovná proměnná true nebo false.

## Ikona aplikace



Obrázek 61 Ikona aplikace

**Zdroj:** [**https://www.visualpharm.com/free-icons/calendar-595b40b65ba036ed117d3cfb**](https://www.visualpharm.com/free-icons/calendar-595b40b65ba036ed117d3cfb)

**Licence: Linkware License**

Závěr

Cílem této práce bylo udělat jednoduší a efektivnější aplikaci pro zápis docházky.

Aplikaci je možné používat na všech mobilních zařízeních s operačním systémem Android.

seznam použité literatury

1. Job ready Flutter complete course with Firebase and Dart. *Job ready Flutter complete course with Firebase and Dart* [online]. [cit.2023-03-12]. Dostupné z: <https://www.udemy.com/course/job-ready-flutter-crash-course-with-firebase/learn/lecture/27871220#overview>
2. AlertDialog class. *AlertDialog class* [online]. [cit. 2023-03-12]. Dostupné z: <https://api.flutter.dev/flutter/material/AlertDialog-class.html>
3. AlertDialog with a TextField in Flutter. *AlertDialog with a TextField in Flutter* [online]. [cit. 2023-03-12]. Dostupné z: <https://www.appsdeveloperblog.com/alert-dialog-with-a-text-field-in-flutter/>
4. Flutter Show Array Elements in ListView Builder Android iOS Example.  [online]. [cit. 2023-03-12]. Dostupné z: <https://flutter-examples.com/flutter-show-array-elements-in-listview/>
5. Flutter Cloud Firestore Example - Firebase Cloud Firestore Read & Write with Flutter.  [online]. [cit. 2023-03-12]. Dostupné z: <https://www.youtube.com/watch?v=TcwQ74WVTTc&t=520s>
6. *How to Add ʺ⋮ʺ Popup Menu on Flutter AppBar* [online]. [cit. 2023-03-12]. Dostupné z: <https://www.youtube.com/watch?v=TcwQ74WVTTc&t=520s>
7. *Flutter: how to make a TextField with HintText but no Underline?* [online]. [cit. 2023-03-12]. Dostupné z: <https://stackoverflow.com/questions/49040679/flutter-how-to-make-a-textfield-with-hinttext-but-no-underline>
8. *3 Ways to Set Height and Width of TextField in Flutter [2023]* [online]. [cit. 2023-03-12]. Dostupné z: <https://www.flutterbeads.com/set-height-and-width-of-textfield-in-flutter/>
9. *Switch class* [online]. [cit. 2023-03-12]. Dostupné z: <https://api.flutter.dev/flutter/material/Switch-class.html>
10. *Flutter – Pixels Overflow Error while Launching Keyboard* [online]. [cit. 2023-03-12]. Dostupné z: <https://www.geeksforgeeks.org/flutter-pixels-overflow-error-while-launching-keyboard/>
11. *Modal Bottom Sheet In Flutter* [online]. [cit. 2023-03-12]. Dostupné z: <https://medium.flutterdevs.com/modal-bottom-sheet-in-flutter-dae05debbed2>
12. *Flutter ListTile (tileColor) conditionally* [online]. [cit. 2023-03-12]. Dostupné z: <https://stackoverflow.com/questions/68919040/flutter-listtile-tilecolor-conditionally>
13. *Display a snackbar* [online]. [cit. 2023-03-12]. Dostupné z: <https://docs.flutter.dev/cookbook/design/snackbars>
14. *Placing two trailing icons in ListTile* [online]. [cit. 2023-03-12]. Dostupné z: <https://stackoverflow.com/questions/54548853/placing-two-trailing-icons-in-listtile>
15. *How to update field in document on cloud firestore using Flutter?* [online]. [cit. 2023-03-12]. Dostupné z: <https://stackoverflow.com/questions/58383016/how-to-update-field-in-document-on-cloud-firestore-using-flutter>
16. *How to add clear button to TextField Widget* [online]. [cit. 2023-03-12]. Dostupné z: <https://stackoverflow.com/questions/48927928/how-to-add-clear-button-to-textfield-widget>
17. *Change Text Color in Flutter  –  The RIGHT Way [2023]* [online]. [cit. 2023-03-12]. Dostupné z: <https://www.flutterbeads.com/change-text-color-in-flutter/>
18. *Implement if statement inside subtitle in ListTile* [online]. [cit. 2023-03-12]. Dostupné z: <https://stackoverflow.com/questions/60512360/implement-if-statement-inside-subtitle-in-listtile>
19. *Flutter SnackBar Background Color* [online]. [cit. 2023-03-12]. Dostupné z: <https://googleflutter.com/flutter-snackbar-background-color/>
20. *How can I change the app display name build with Flutter?* [online]. [cit. 2023-03-12]. Dostupné z: <https://stackoverflow.com/questions/49353199/how-can-i-change-the-app-display-name-build-with-flutter>
21. *How to change the application launcher icon on Flutter?* [online]. [cit. 2023-03-12]. Dostupné z: <https://stackoverflow.com/questions/43928702/how-to-change-the-application-launcher-icon-on-flutter>
22. *Flutter give container rounded border* [online]. [cit. 2023-03-12]. Dostupné z: <https://stackoverflow.com/questions/57777737/flutter-give-container-rounded-border>
23. *DATE PICKER • Flutter Widget of the Day #25* [online]. [cit. 2023-03-12]. Dostupné z: <https://www.youtube.com/watch?v=JK3zztXnDxs&list=PL_NdaDBZ1owbQ534FAuPV1iGbLPcBvepw&index=5&t=12s>
24. *📱Sign Up Users • Firebase x Flutter Tutorial ♡* [online]. [cit. 2023-03-12]. Dostupné z: <https://www.youtube.com/watch?v=Mfa3u3naQew&list=PL_NdaDBZ1owbQ534FAuPV1iGbLPcBvepw&index=3&t=564s>
25. *📱Login & Logout • Firebase x Flutter Tutorial ♡* [online]. [cit. 2023-03-12]. Dostupné z: <https://www.youtube.com/watch?v=TkuO8OLgvkk&t=598s>
26. *Futures and Streams (Flutter FutureBuilder, StreamBuilder)* [online]. [cit. 2023-03-12]. Dostupné z: <https://www.youtube.com/watch?v=lkpPg0ieklg>
27. *Firebase UI for Auth (Package of the Week)* [online]. [cit. 2023-03-12]. Dostupné z: <https://www.youtube.com/watch?v=jtFaMv2vGF0&list=WL&index=16>
28. *FlutterIcon - Flutter custom icons generator* [online]. [cit. 2023-03-12]. Dostupné z: <https://www.fluttericon.com>

Seznam obrázků

[Obrázek 1 Obrazovka přihlášení 11](#_Toc129535636)

[Obrázek 2 Obrazovka registrace 12](#_Toc129535637)

[Obrázek 3 Chybné heslo 13](#_Toc129535638)

[Obrázek 4 Úvodní stránka 14](#_Toc129535639)

[Obrázek 5 Menu úvodní obrazovky 15](#_Toc129535640)

[Obrázek 6 Stránka pro výběr/vytvoření lekce 16](#_Toc129535641)

[Obrázek 7 Dialogové okno pro vytvoření lekce 17](#_Toc129535642)

[Obrázek 8 Vytvořená lekce 18](#_Toc129535643)

[Obrázek 9 Více lekcí 19](#_Toc129535644)

[Obrázek 10 Hlavní stránka s vybranou lekcí 20](#_Toc129535645)

[Obrázek 11 Vytvořené datum 21](#_Toc129535646)

[Obrázek 12 Kalendář pro zpětné vytvoření data 22](#_Toc129535647)

[Obrázek 13 Prázdný seznam lidí 23](#_Toc129535648)

[Obrázek 14 Vytvořený člověk 24](#_Toc129535649)

[Obrázek 15 Veškeré informace o člověku 25](#_Toc129535650)

[Obrázek 16 Nedoplněné informace o člověku 26](#_Toc129535651)

[Obrázek 17 Menu pro filtrování 27](#_Toc129535652)

[Obrázek 18 Formulář pro přidání člověka 28](#_Toc129535653)

[Obrázek 19 Hláška pro nedoplněné jméno 29](#_Toc129535654)

[Obrázek 20 Hláška pro nedoplněné příjmení 30](#_Toc129535655)

[Obrázek 21 Formulář pro editaci člověka 31](#_Toc129535656)

[Obrázek 22 Zápis docházky 32](#_Toc129535657)

[Obrázek 23 Zapsání přítomnosti s hláškou 33](#_Toc129535658)

[Obrázek 24 Zapsání nepřítomnosti s hláškou 34](#_Toc129535659)

[Obrázek 25 Pluginy 37](#_Toc129535660)

[Obrázek 26 Volání pluginů 38](#_Toc129535661)

[Obrázek 27 Diagram databáze 39](#_Toc129535662)

[Obrázek 28 Metoda Main 40](#_Toc129535663)

[Obrázek 29 Metoda *Prihlasovani* 40](#_Toc129535664)

[Obrázek 30 Metoda *Dispose* u příhlášení 41](#_Toc129535665)

[Obrázek 31 Metoda *Dispose* u registrace 41](#_Toc129535666)

[Obrázek 32 Metoda *Registrace* 42](#_Toc129535667)

[Obrázek 33 Metoda \_*ukazKalendar* 42](#_Toc129535668)

[Obrázek 34 Metoda *hesloSpravne* 43](#_Toc129535669)

[Obrázek 35 Metoda *\_vymazVse* 43](#_Toc129535670)

[Obrázek 36 Třída *Uvodni* 44](#_Toc129535671)

[Obrázek 37 Třída *Cesty* 45](#_Toc129535672)

[Obrázek 38 Třída *Autorizacni* 46](#_Toc129535673)

[Obrázek 39 Widget – Textové pole pro heslo 47](#_Toc129535674)

[Obrázek 40 Widget – Textové pole pro heslo 48](#_Toc129535675)

[Obrázek 41 Widget – Tlačítko pro přihlášení 49](#_Toc129535676)

[Obrázek 42 Widget – Odkaz na stránku registrace 49](#_Toc129535677)

[Obrázek 43 Widget – Textové pole pro registraci emailem 50](#_Toc129535678)

[Obrázek 44 Widget – Textové pole pro heslo s možností ukázání hesla 51](#_Toc129535679)

[Obrázek 45 Widget - Tlačítko pro volání třídy *SeznamLidi* 52](#_Toc129535680)

[Obrázek 46 Widget – Menu na hlavní stránce 53](#_Toc129535681)

[Obrázek 47 Widget – StreamBuilder s výpisem dat 54](#_Toc129535682)

[Obrázek 48 Proměnná pro výpis dat z databáze 55](#_Toc129535683)

[Obrázek 49 Widget – Tlačítko pro vytváření nového data 55](#_Toc129535684)

[Obrázek 50 Proměnná pro výpis dat z databáze s filtrem podle uživatele 56](#_Toc129535685)

[Obrázek 51 Widget – Okno pro vytvoření nové lekce 56](#_Toc129535686)

[Obrázek 52 Widget – Spodní menu navigační lišty 57](#_Toc129535687)

[Obrázek 53 Widget – Menu pro vybrání filtrování nedoplněné hodnoty 58](#_Toc129535688)

[Obrázek 54 Widget – Karta s výpisem všech informací o uživateli 59](#_Toc129535689)

[Obrázek 55 Data odeslaná pro editaci 60](#_Toc129535690)

[Obrázek 56 Obsah karty s žákem 60](#_Toc129535691)

[Obrázek 57 Widget – Textové pole pro vložení adresy bydliště žáka 61](#_Toc129535692)

[Obrázek 58 Widget – Tlačítko pro potvrzení GDPR 61](#_Toc129535693)

[Obrázek 59 Podmínky v tlačítku pro odeslání dat do databáze 62](#_Toc129535694)

[Obrázek 60 Widget – Textové pole pro editaci informace o žakovi 63](#_Toc129535695)

[Obrázek 61 Ikona aplikace 64](#_Toc129535696)

# Seznam Tabulek

[Tabulka 1 Pluginy 36](#_Toc129383961)

[Tabulka 2 Firebase služby 37](#_Toc129383962)

seznam Příloh

* Instalační soubor docházka.apk
* Visual Studio Code projekt aplikace