DELTA – Střední škola informatiky a ekonomie, s.r.o.

Ke Kamenci 151, Pardubice

Prohlašuji, že jsem maturitní projekt vypracoval(a) samostatně, výhradně s použitím uvedené literatury.

Málek, Ondřej

4.A

Informační technologie 18-20-/01

2023/2024

Chore Master

Aplikace pro rodinné úkoly

V Pardubicích 30.3.2024

(vlastnoruční podpis)

Obsah

[Úvod 6](#_Toc162810966)

[1.Porovnání s ostatními aplikacemi 7](#_Toc162810967)

[1.1.Sweepy: Home Cleaning Schedule 7](#_Toc162810968)

[1.2.Nipto: Split Household Chores 7](#_Toc162810969)

[2.Technologie 8](#_Toc162810970)

[2.1.Java 8](#_Toc162810971)

[2.2.XML 8](#_Toc162810972)

[2.3.Firebase authentication 8](#_Toc162810973)

[2.4.Firebase Firestore Database 8](#_Toc162810974)

[2.5.Firebase Hosting 9](#_Toc162810975)

[2.6.HTML, JavaScript 9](#_Toc162810976)

[3.Aplikace 10](#_Toc162810977)

[3.1.Architektura aplikace 10](#_Toc162810978)

[3.2.Adresářová struktura 10](#_Toc162810979)

[3.3.Implementace 10](#_Toc162810980)

[3.3.1.Firestore Databáze 10](#_Toc162810981)

[3.3.2.Správa uživatelů 13](#_Toc162810982)

[3.3.3.Firebase Hosting 14](#_Toc162810983)

[4.Funkce aplikace 17](#_Toc162810984)

[4.1.Správa účtů 17](#_Toc162810985)

[4.1.1.Přihlášení 17](#_Toc162810986)

[4.1.2Registrace 17](#_Toc162810987)

[4.1.3.Zapomenutí hesla 17](#_Toc162810988)

[4.1.4.Vytvoření uživatelského jména a skupiny 18](#_Toc162810989)

[4.2.Hlavní část aplikace 18](#_Toc162810990)

[4.2.1.Hlavní aktivita 18](#_Toc162810991)

[4.2.2.Task fragment 18](#_Toc162810992)

[4.2.3.Calendar fragment 19](#_Toc162810993)

[4.2.4.Stats Fragment 20](#_Toc162810994)

[4.2.5.Setings Fragment 21](#_Toc162810995)

[4.2.6.Create Task Activity 22](#_Toc162810996)

[4.2.8.Create Group Activity 24](#_Toc162810997)

[4.2.9.Manage Group Activity 24](#_Toc162810998)

[4.2.10.Manage Profile Activity 24](#_Toc162810999)

[5.Závěr 25](#_Toc162811000)

[6.Literatura 26](#_Toc162811001)

[7.Přílohy 27](#_Toc162811002)

**Anotace**

Aplikace Chore Master slouží pro jednoduchou a efektivní správu úkolů a umožňuje uživatelům lepší organizaci svého času a zvýšení produktivity jak v osobním, tak v profesním životě. Tato aplikace představuje platformu pro vytváření, správu a sledování úkolů na individuální i skupinové úrovni, podporuje spolupráci mezi členy týmu a správu těchto týmových skupin. Cílem této práce je nabídnout uživatelům jednoduché, ale efektivní řešení pro správu jejich každodenních povinností, a tím přispět k jejich lepší organizovanosti a celkové spokojenosti.

**Klíčová slova**

Android aplikace, Java, Cloudová databáze, Správa úkolů, Sdílení informací

**Anotation**

The Chore Master app is used for simple and efficient task management, allowing users to better organize their time and increase productivity in both their personal and professional lives. The app provides a platform for creating, managing and tracking tasks at an individual and group level, and supports collaboration between team members and the management of these groups. The aim of this work is to offer users a simple but effective solution to manage their daily responsibilities, thus contributing to their better organization and overall satisfaction.

**Key words**

Android application, Java, Cloud database, Task management, Information sharing

# Úvod

V dnešní době, kdy se tempo života neustále zrychluje a množství povinností roste, se stává efektivní organizace času nejen potřebou, ale i výzvou. V tomto kontextu přichází aplikace ChoreMaster jako praktické řešení, které se zaměřuje na zjednodušení správy úkolů a podporu vzájemné spolupráce mezi uživateli. Aplikace se soustředí na efektivní přidělování a sledování úkolů v rámci skupin, což uživatelům umožňuje lépe se organizovat, vzájemně si pomáhat v dosahování cílů a sledovat jejich plnění. Aplikace je navržena tak, aby podporovala rychlou orientaci v úkolech, čímž se stává ideálním řešením pro každodenní organizaci a vzájemnou spolupráci.

# 1.Porovnání s ostatními aplikacemi

## 1.1.Sweepy: Home Cleaning Schedule

Sweepy je ideální mobilní aplikace na rozložení a rozvrhnutí domácích prací. Aplikace uživatele rychle a jednoduše provede nastavením všech potřebných věcí, jako je vytváření nových nebo připojování se k již existujícím teamům, nastavením všech místností a prací, kterou chce uživatel v daných místnostech vykonávat. Dále po stanovených časových intervalech aplikace posílá oznámení o potřebné práci, za jejíž vykonání jsou uživateli přidělovány body. Členové týmu tak mezi sebou mohou soutěžit o co největší počet získaných bodů, které vidí na leaderboardu. Sweepy nabízí i variantu různých dalších funkcí, mezi které patří například teamové challenge, streaky, divoké karty, které se získají za dobré plnění úkolů a mohou se použít na přeskočení dne na uklízení bez ztráty streaku, časovou linku, na které se vše zobrazuje a mnoho dalšího. Aplikace toho sice umí dost, ale hodně z funkcí se nachází za paywallem a v základu dokonce nepodporuje ani základní funkci teamů.

## 1.2.Nipto: Split Household Chores

Tato aplikace funguje velice podobně jako Sweepy, ale oproti ní má o něco méně funkcí a působí o trochu více nudněji, ovšem její velkou výhodou je, že i když se určité funkce také nachází za paywallem, tak v základu má více funkcí zadarmo než Sweepy. Nipto také podporuje vytváření teamů, získávání skóre a jeho porovnávání s ostatními, vzájemné plnění úkolů, upozornění a mnoho dalšího. Jedná se o skvělou volbu, pokud se člověku nechce platit za pár funkcí navíc a lepší vzhled.

# 2.Technologie

## 2.1.Java

Java je běžně používaný, objektově orientovaný programovací jazyk navržený tak, aby měl co nejnižší závislosti na implementaci. Díky své platformově nezávislé povaze - "Napiš jednou, běž kdekoli" (WORA) - mohou být aplikace napsané v Javě spuštěny na jakékoli platformě, která je vybavena Java Virtual Machine (JVM). Java je známá svou robustností, bezpečností a výkonností, což ji činí populární volbou pro vývoj různých typů softwarových aplikací.

## 2.2.XML

XML (Extensible Markup Language) je značkovací jazyk používaný pro reprezentaci hierarchicky strukturovaných dat. Ve vývoji aplikací, zejména pro Android, se XML často využívá k definování layoutů uživatelských rozhraní a konfiguraci aplikace, jako je manifest aplikace a zdrojové soubory. Jeho deklarativní povaha umožňuje oddělit design uživatelského rozhraní od kódu aplikace, což usnadňuje vývoj a údržbu.

## 2.3.Firebase authentication

Firebase Authentication poskytuje komplexní systém pro autentizaci uživatelů, který podporuje různé metody přihlášení, včetně e-mailu a hesla, sociálních médií jako je Google, Facebook a Twitter, a dokonce i telefonního čísla. Umožňuje vývojářům snadno integrovat bezpečné přihlašování do jejich aplikací a spravovat uživatelské účty. Díky propojení s dalšími službami Firebase, jako jsou Cloud Firestore nebo Realtime Database, mohou vývojáři rychle vytvářet všechny potřebné funkce. Firebase Authentication také zjednodušuje správu uživatelských účtů a přístupových oprávnění, což pomáhá udržovat vysokou úroveň bezpečnosti a ochrany soukromí v aplikacích.

## 2.4.Firebase Firestore

Firebase Firestore je flexibilní, škálovatelná databáze pro mobilní, webové a serverové vývojové aplikace od Firebase a Google Cloud Platform. Je to NoSQL cloudová databáze, která ukládá data ve formátu dokumentů a kolekcí, což umožňuje rychlý vývoj a jednoduchou integraci s aplikacemi. Firestore nabízí živé synchronizace dat a automatické ukládání offline, což zajišťuje, že aplikace jsou responzivní a spolehlivé i bez internetového připojení. Díky real-time aktualizacím a možnostem zabezpečení na úrovni dokumentů poskytuje Firestore výkonný základ pro tvorbu kolaborativních a dynamických aplikací.

## 2.5.Firebase Hosting

Firebase Hosting poskytuje rychlý a bezpečný web hosting pro webové aplikace. S integrovanou podporou pro HTTPS a automatickým poskytováním SSL certifikátů zajišťuje vysokou úroveň zabezpečení. Díky globálně distribuované síti obsahu (CDN) jsou weby rychle dostupné odkudkoliv na světě s minimální latencí.

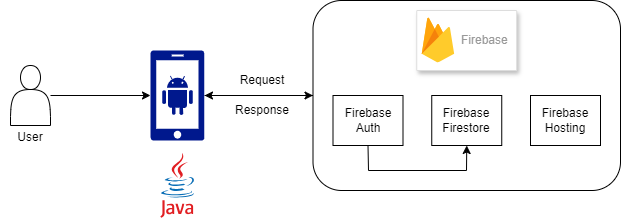
## 2.6.HTML, JavaScript

HTML je standardní značkovací jazyk pro vytváření webových stránek a aplikací. Umožňuje strukturovat obsah a layout webových stránek pomocí různých elementů jako jsou nadpisy, odstavce, odkazy a obrázky. JavaScript je skriptovací programovací jazyk, který se používá pro přidání interaktivních prvků na webové stránky. Společně tvoří základní technologie pro webový vývoj, kde HTML definuje strukturu stránky a JavaScript obohacuje uživatelské rozhraní o dynamické a interaktivní funkce.

# 3.Aplikace

## 3.1.Architektura aplikace

Architekturu aplikace tvoří dva základní prvky. Samotná androidí aplikace a firebase se službami authetifikace, firestore databáze a web hosting, jak je vidět na obrázku číslo 1.



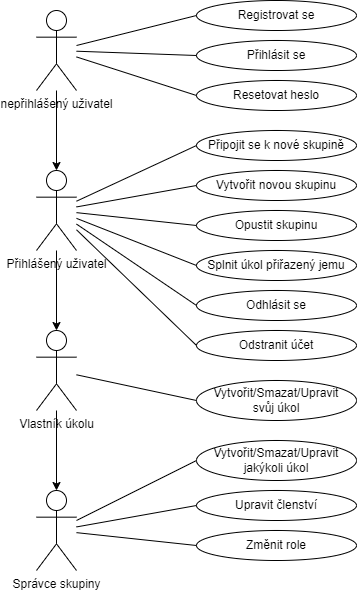
Obrázek 1 Architerktura aplikace

Jak lze z obrázku poznat, nejedná se vůbec o složitou architekturu. Aplikace samotná byla celá napsána v Javě a komunikuje s již zmíněnými firebase službami. Díky využití ne velkého množství technologií, je aplikace poměrně jednoduchá na pochopení a správu. Tento minimalistický přístup také usnadňuje další rozvoj a přidávání nových funkcí do aplikace. Navzdory své na pohled jednoduché architektuře, aplikace vyždímává velké množství nabýzených funkcí, pro všechny potřebné úkony.

## 3.2.Adresářová struktura

Všechny java třídy se nachází v cestě app/src/main/java/project/taskmaster/choremaster. Dále v složce main/res/drawable se nachází všechny ikony používané aplikací. V res/layout jsou všechny xml soubory na rozložení obrazovky. V res/menu se nachází xml soubor na navigační lištu. Ve složce app se také nachází důležitý soubor google-services.json, pro připojení aplikace k firebase. Ve složce main se také nachází soubor androidManifest.xml, který definuje základní atributy aplikace, její komponenty a požadovaná oprávnění.

## 3.3.Implementace

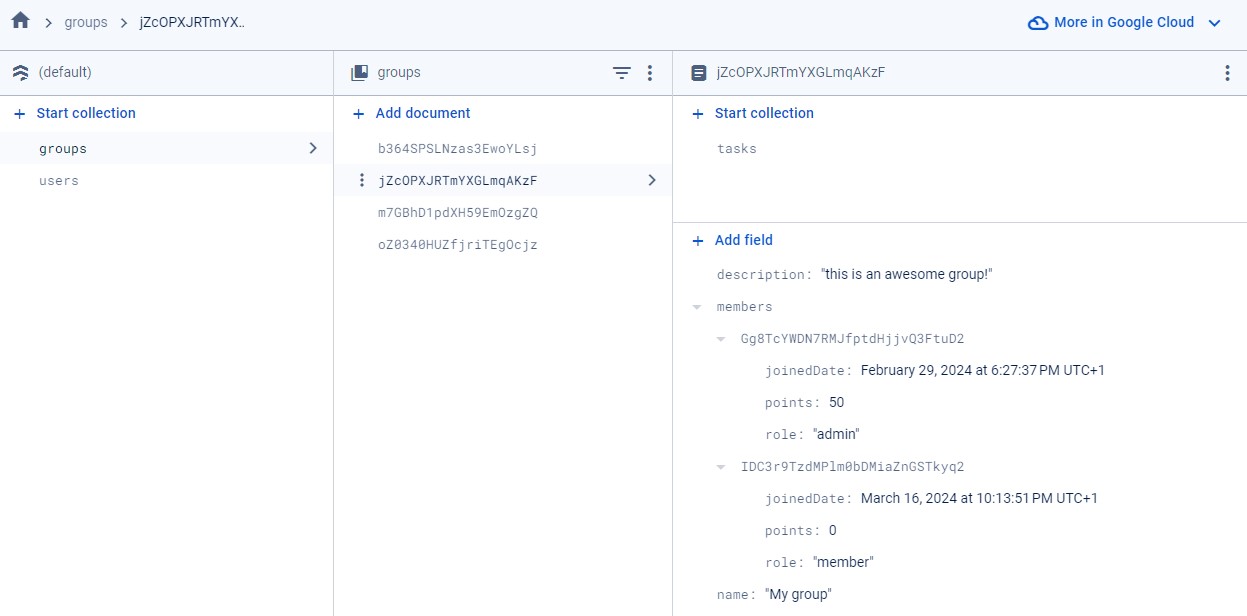


Obrázek 2 Use case diagram

### 3.3.1.Firestore Databáze

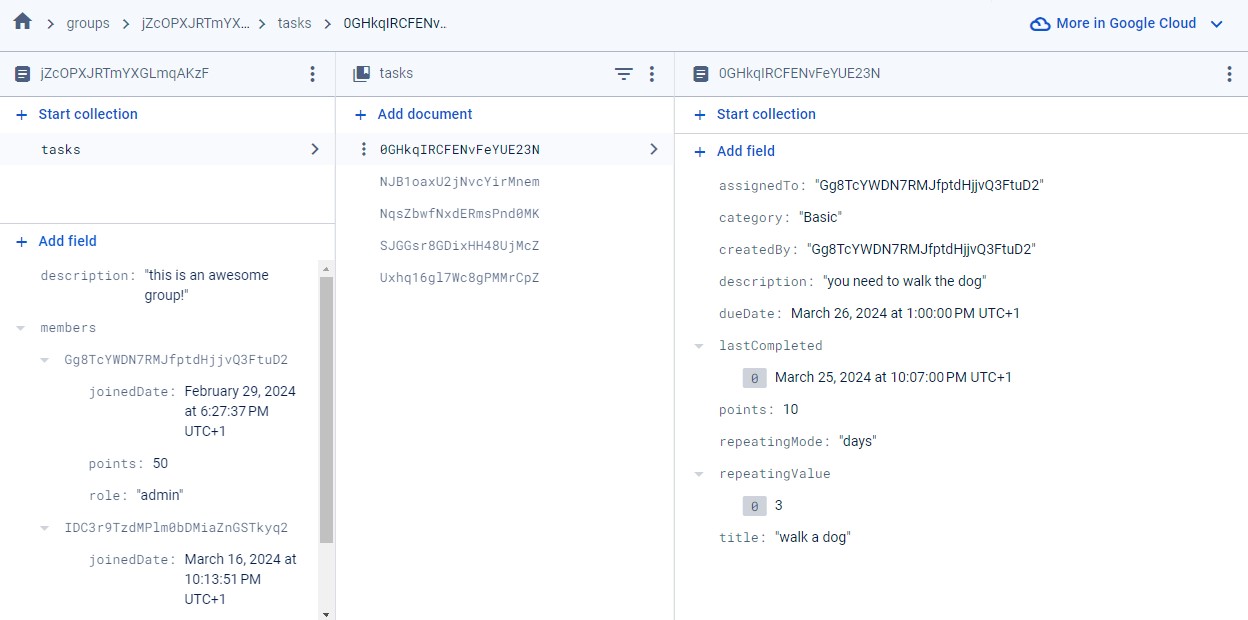
Pro tento projekt byla zvolena databáze firestore. Kvůli velkému počtu build-in funkcí a NOSQL architektuře, dobře uspokojuje všechny potřeby aplikace.

Jelikož aplikace má umožňovat uživatelům pracovat ve skupinách, bylo potřeba vytvořit kolekci groups, do které se budou ukládat. Dokumenty v této kolekci mají automaticky vygenerované UID a obsahují data o jméně a popisu skupiny. Dále se zde ukládají id všech uživatelů zapsaných do konkrétní skupiny s informacemi, o tom kdy se připojili, kolik se jim už podařilo nabírat v rámci skupiny bodů a jakou mají ve skupině roli, tedy buď administrátor nebo pouze member.



Obrázek 3 kolekce groups

Dále se také v každém dokumentu skupiny nachází subkolekce tasks, pro ukládání úkolů v dané skupině. Tento přistup byl zvolen z toho důvodu, že úkolů může být velice hodně a tak by dokument skupiny mohl být velmi velký, což by s ním způsobilo ztíženou manipulaci a delší dobu na stažení či nahrání. Tímto způsobem je všechno hezky rozdělené a jednoduše si můžeme stáhnout pouze to co zrovna potřebujeme.



Obrázek 4 subkolekce tasks

Každý úkol pak v sobě obsahuje informace o tom, jaké má jméno, detailnější popisek a kolik uživatel dostane bodů za jeho splnění. Dále kým byl vytvořen a komu byl přiřazen, také se do něj ukládá, do jaké kategorie spadá, na kdy má být hotový, jestli se opakuje a pokud ano tak jakým způsobem a po kolika dnech nebo v jakých dnech v týdnu.

Obsah obrázku text, snímek obrazovky, Písmo, dokument

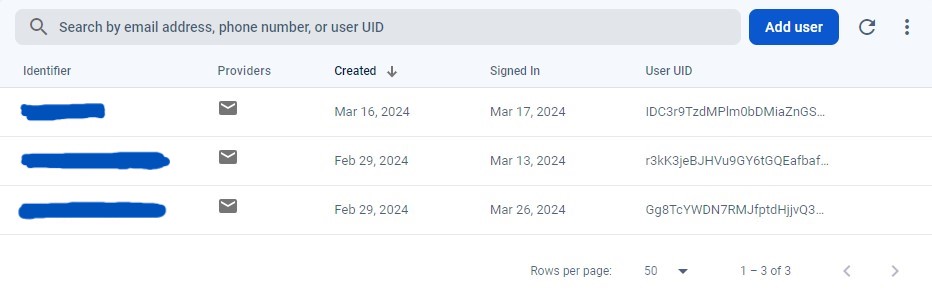
Popis byl vytvořen automaticky

Obrázek 5 kód na vypočítání dalšího dueDate

Tento kód slouží pro vypočítání dalšího „dueDate“, neboli data na kdy má být úkol hotový. Funkce přijímá za parametry kalendář dueDate, režim opakování, což bude buď „weekly“ nebo „days“, hodnota „none“ se zde nemůže objevit nikdy, protože pro ni se ani tato metoda nevolá z toho důvodu, že u ní se stejně další datum počítat nemá. Do dalších parametrů patří opakovací hodnota, tedy v jakých dnech, nebo po kolika dnech se úkol opakuje a flag s boleanovou hodnotou. Poté se vypočítá další datum podle zadaných parametrů a vrátí se zpět.

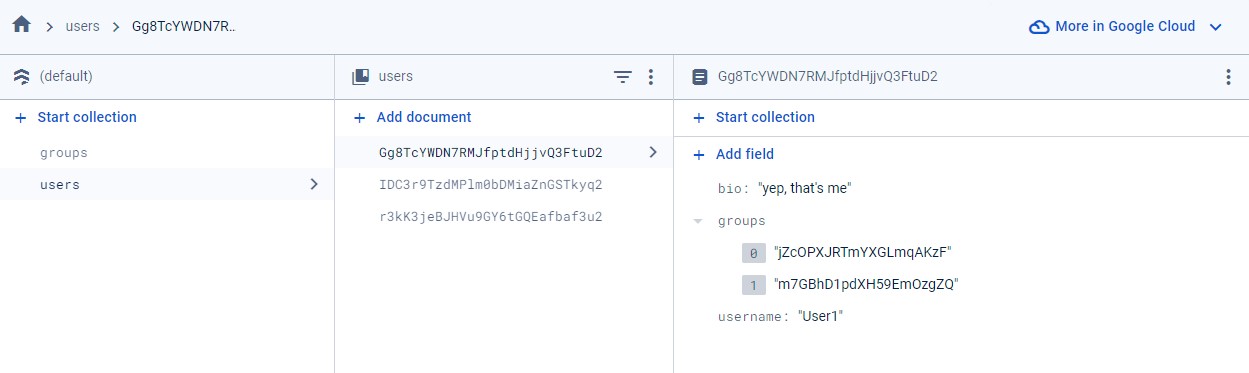
### 3.3.2.Správa uživatelů

Aplikace disponuje funkcemi pro přihlašování, registraci, odstraňování uživatelů a podporou pro změnění hesla při jeho zapomenutí. Tyto funkce jsou zprostředkovány balíčkem firebase-auth, který aplikaci umožňuje využívat služby firebase authentication pro správu účtů v cloudu. Pro využití těchto služeb je nutné vytvořit takzvaný provider, který určuje způsob přihlášní. V tomto projektu je používán provider Email/Password. To znamená, že akce s účtem jsou prováděny přes email a heslo a jsou založené přímo ve firebase.



Obrázek 6 Firebase authentication interface

Firebase authentication nám sice umožňuje správu uživatelů, ale už neumožňuje ukládání nějakých dodatečných dat o uživatelích. A zde přichází na řadu integrace s firestore databází. Při procesu vytváření účtu se totiž vezme id uživatele z firebase authentication a pod tímto id se vytvoří dokument ve firestore databázi v kolekci users. Jak je vidět na obrázku číslo 6, do kolekce se ukládají pouze dokumenty, které reprezentují dané uživatele. Do tohoto dokumentu se ukládá uživatelské jméno, bio, a později se zde bude ukládat i id skupin do kterých daný uživatel patří.



Obrázek 7 kolekce users

### 3.3.3.Firebase Hosting

Skutečnost, že se v projektu používá i hosting na webové stránky se může zprvu zdát poněkud zvláštní, když se čistě jedná o aplikaci na android. Abychom pochopili, proč je toto pro aplikaci nezbytné musíme si nejdříve ujasnit, jak funguje přidávání uživatelů do skupin. Toto přidávání funguje přes takzvaný deeplink s id té skupiny, do jaké vede. Deeplink funguje podobně jako obyčejný link na webovou stránku, s tím rozdílem, že android při kliknutí na daný link ho nemusí nutně otevřít ve webovém prohlížeči, ale podle jeho struktury vybere příslušnou aplikaci, která ho umí otevřít. Příklad takového deeplinku může vypadat takto:

app://appname.example/data?object=null

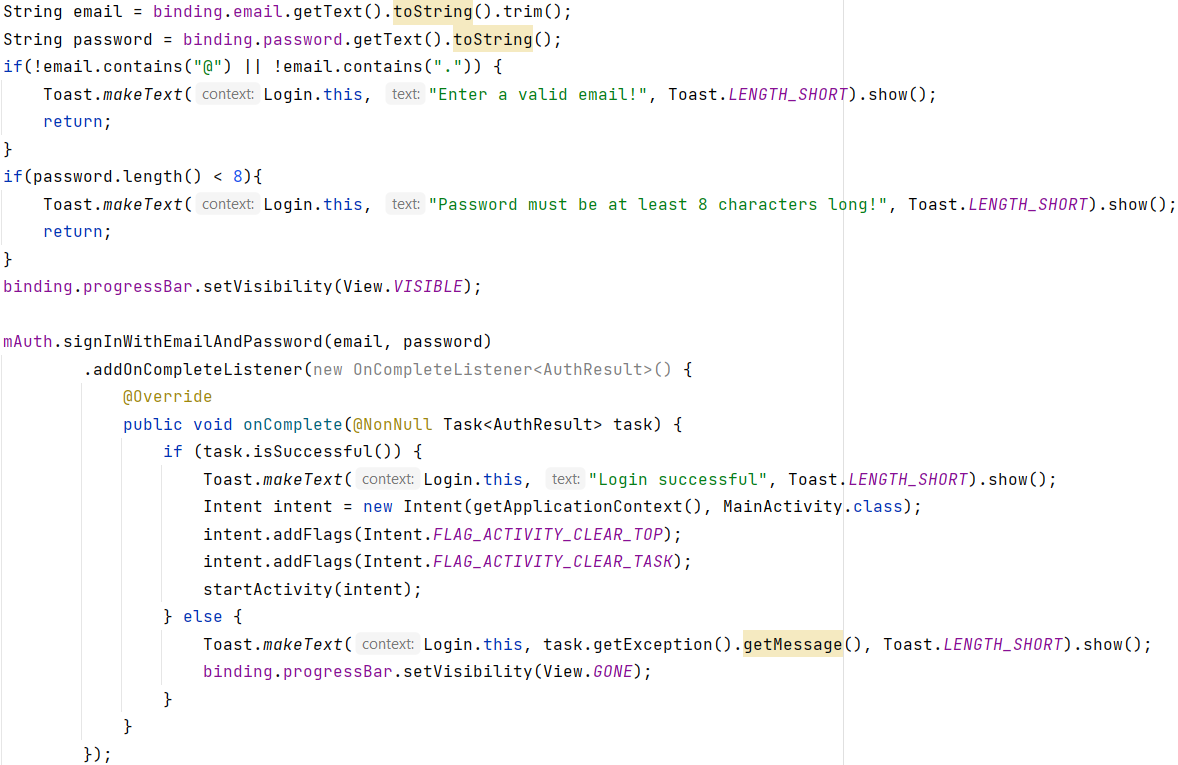
Takto by daný link fungoval bez problému, když by aplikace byla nastavená na jeho přijímání. Problém ale nastává sdílením tohoto linku. Některé služby jako třeba Gmail totiž neuznávají jiné schémata, kromě https nebo http, a tak do klikatelného linku nezahrnou vlastní schéma jako například app:// a tím pádem tímto způsobem není možné otevřít nic jiného než webovou stránku. Proto je potřeba vytvořit si vlastní webovou stránku, na kterou každý link povede a až ten web může spustit deeplink, který otevře aplikaci a přidá uživatele do požadované skupiny.

Obsah obrázku text, snímek obrazovky, Písmo, dokument

Popis byl vytvořen automaticky

Obrázek 8 kód webové stránky

Obrázek číslo 7, je ukázka kódu webové stránky, pro zpracování deeplinku. Po načtení stránky to vezme groupId z linku pod kterým byl web spuštěn, vloží ho to do nyní už funkčního linku pro zapnutí aplikace a na ten to uživatele přesměruje. Pokud by k přesměrování nedošlo do 2.5 sekund, na obrazovku se zobrazí text napsaný v těle stránky <body>.



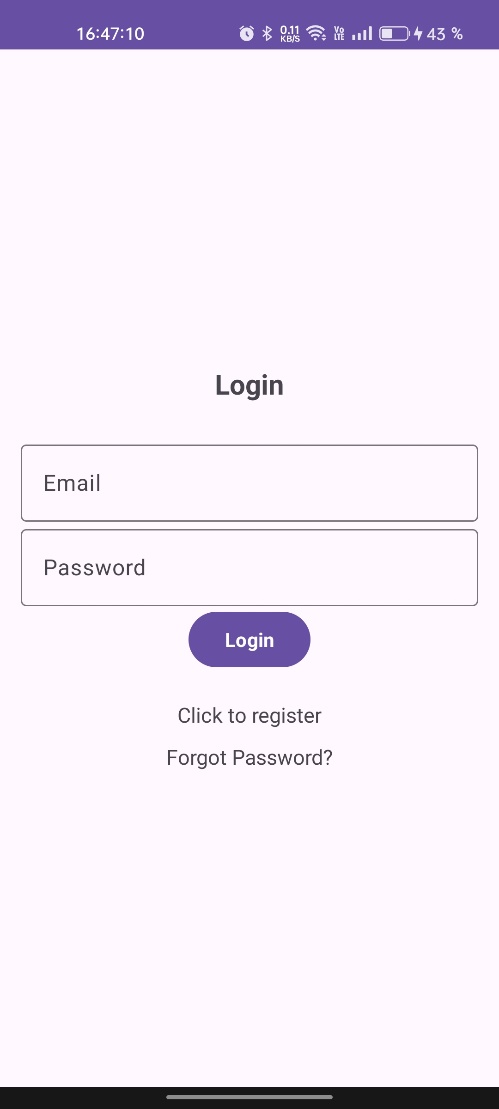
Obrázek 9: Ukázka přihlášení uživatele

Tento kus kódu slouží pro validaci e-mailové adresy a hesla a následné přihlášení uživatele. Nejdříve kód zkontroluje zdali zadaný e-mail obsahuje zavináč a tečku. I když se toto může zdát zbytečné, protože by firebase stejně žádného uživatele nenašel, když by byla adresa špatně zadaná, tak to zachrání spoustu zbytečných žádostí na databázi a ušetří čas. Poté se email s zaheshovaným heslem pošle na firebase a zpátky se vrátí jestli se zadaná data shodují s nějakým záznamem v databázi a pokud ano, na obrazovce se zobrazí zpráva s oznámením, že se podařilo uživatele přihlásit a uživatel je přesměrován na hlavní aktivitu, bez možnosti vrácení se zpátky, díky odstranění informace z intentu, odkud je uživatel přesměrováván.

# 4.Funkce aplikace

## 4.1.Správa účtů

### 4.1.1.Přihlášení

Když uživatel zapne aplikaci, tak jako první se zapne Login aktivity, která slouží pro přihlášení uživatele. Nachází se zde tlačítka pro registraci a zapomenutí hesla, která po kliknutí uživatele přesměrují na příslušné aktivity. Dále tu jsou 2 textová pole do kterých uživatel zadává svůj e-mail a heslo. Po kliknutí na tlačítko Login, aktivita ověří zda se jedná o validní e-mailovou adresu a jestli je heslo dlouhé alespoň 8 znaků. Následně pokud uživatel s danými přihlašovacími údaji existuje, aplikace uživatele přihlásí a přesměruje ho na hlavní aktivitu (Main aktivity).

Obrázek 10 Login obrazovka

### Obsah obrázku text, snímek obrazovky, Písmo, číslo Popis byl vytvořen automaticky4.1.2Registrace

Obrázek 11 Register obrazovka

Na této aktivitě se nachází textová pole pro zadání e-mailu, hesla a zopakování hesla. Po kliknutí na tlačítko register, aktivita ověří, zda se jedná o validní e-mailovou adresu a jestli se zadaná hesla shodují. Pokud s tím není problém, tak se se uživatelské informace uloží, aktivita uživatele automaticky přihlásí a přesměruje ho na aktivitu pro zadání uživatelského jména a bia.

### 4.1.3.Zapomenutí hesla

Obrázek 12 Zapomenutí hesla obrazovka

Obsah obrázku text, snímek obrazovky, Písmo, řada/pruh

Popis byl vytvořen automatickyNa této aktivitě je uživatel vyzván k zadání jeho e-mailové adresy a po kliknutí na tlačítko „Send email“ se na zadanou emailovou adresu pošle email s linkem pro změnění hesla a uživatel je mezitím přesměrován zpět na Login aktivitu.

Obsah obrázku text, snímek obrazovky, Písmo

Popis byl vytvořen automatickyObsah obrázku text, Písmo, snímek obrazovky

Popis byl vytvořen automaticky

Obrázek 13 webový formulář na reset hesla

Obrázek 14 email na reset hesla

Po kliknutí na odkaz se otevře tento webový formulář, který po zadání nového hesla, heslo aktualizuje a uživatel je v aplikaci ihned schopen se přihlásit s novým heslem.

### 4.1.4.Vytvoření uživatelského jména a skupiny

Obsah obrázku text, snímek obrazovky, Písmo, řada/pruh

Popis byl vytvořen automatickyTato aktivita se zapne pokaždé, když je uživatel přihlášený, ale nemá ještě uživatelské jméno. Na této aktivitě je uživatel vyzván k zadání uživatelského jména, které musí být aspoň 4 znaky dlouhé a popřípadě popisku profilu (bia). Po kliknutí tlačítka se zadané údaje uloží. Také se automaticky vytvoří skupina s názvem „My group“, do které to vloží tohoto uživatele. Po úspěšném provedení těchto akcí, je uživatel přesměrován na hlavní aktivitu.

Obrázek 15 nastavení jména obrazovka

## 4.2.Hlavní část aplikace

### 4.2.1.Hlavní aktivita

Hlavní aktivita (main aktivity) je hlavní částí aplikace. Slouží pro počáteční stažení dat a navigaci mezi ostatními částmi aplikace. Na dolní části se nachází navigační lišta, která slouží pro přepínání mezi fragmenty. A jako první aktivita zapne „TaskFragment“, pro zobrazení všech úkolů.

### 4.2.2.Task fragment

Tento fragment, jak už bylo řečeno dříve, slouží pro zobrazení úkolů ve skupině. Fragment automaticky aktualizuje list úkolů, takže je vždy aktuální a odpovídá reálným změnám v databázi jinými uživateli. V seznamu úkolů se uživateli zobrazuje pouze název, popisek, kategorie a datum splnění daného úkolu. Každý úkol se dá rozkliknout, pro zobrazení podrobnějších informací. Nahoře fragmentu jsou 3 tlačítka, která slouží pro filtrování zobrazovaných úkolů. První zobrazuje všechny úkoly ve skupině, druhé zobrazuje pouze úkoly přiřazené přihlášenému uživateli a třetí zobrazuje úkoly vytvořené aktuálním uživatelem. Dále je zde tlačítko na vytvoření nového úkolu, které po kliknutí spustí aktivitu Obsah obrázku text, elektronika, snímek obrazovky, software

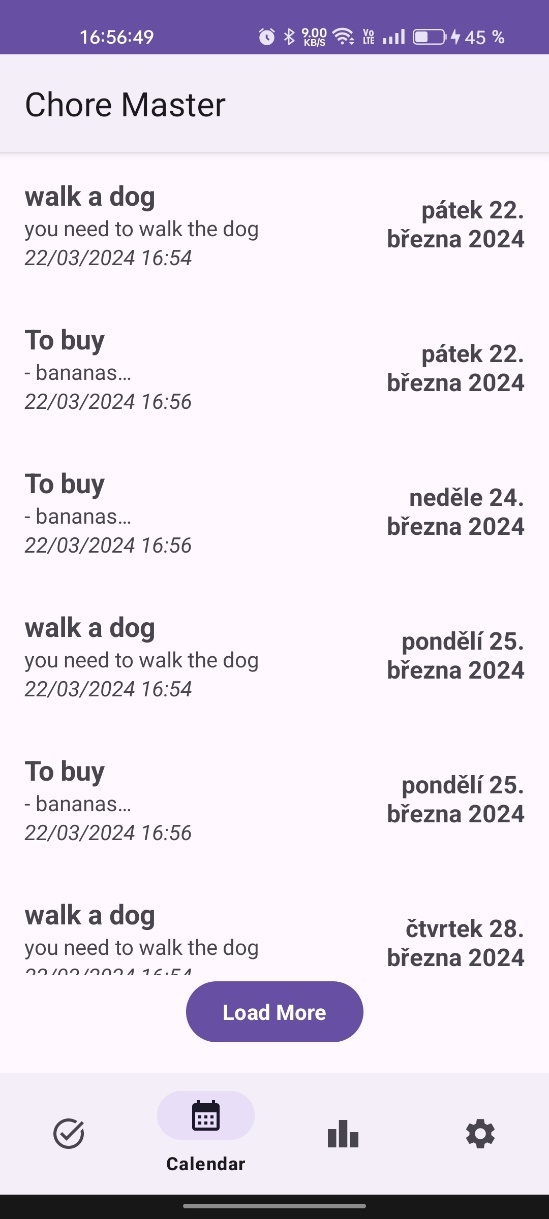
Popis byl vytvořen automaticky„Create Task“.

Obrázek 16 tasks fragment

### 4.2.3.Calendar fragment

Tento fragment je opět dostupný z navigační lišty hlavní aktivity. Fragment slouží pro chronologické zobrazení úkolů podle data, kdy mají být splněny, a to do neomezené budoucnosti. A navíc pokud má úkol nastavené opakování, zobrazí se zde pro každé datum, na kdy má být v budoucnu splněn. V základu fragment zobrazí pouze úkoly do konce týdne, ale po kliknutí tlačítka na načtení dalších, které se nachází na dolní části obrazovky, se zobrazí úkoly i na dalších 7 dní (další týden), po dalším kliknutí, dalších 7 dní, a tak dále. Úkoly se zobrazují se stejnými informacemi jako v „Task fragmentu“ až na kategorii, místo které se zobrazuje, na jaký den v týdnu a jaké datum to má být splněno. Fragment se také stará o to, aby se všem úkolům, co mají datum dalšího splnění v minulosti, změnilo v databázi na nejbližší termín v budoucnosti podle toho jak je nastavené opakování daných úkolů.

### 4.2.4.Stats Fragment

Tento fragment, opět dostupný z navigační lišty hlavní aktivity, neslouží pro nic jiného než pouhé zobrazení bodů jednotlivých uživatelů v dané skupině. Jeho alternativní použití může být zjištění jací uživatelé jsou přidáni ve skupině.

Obrázek 17 Calendar fragment

### 4.2.5.Setings Fragment

Obrázek 18 Stats fragment

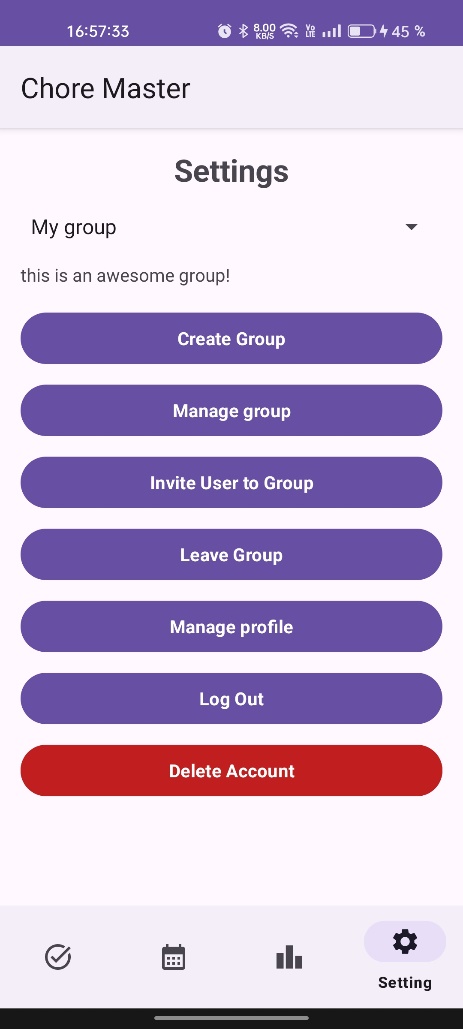
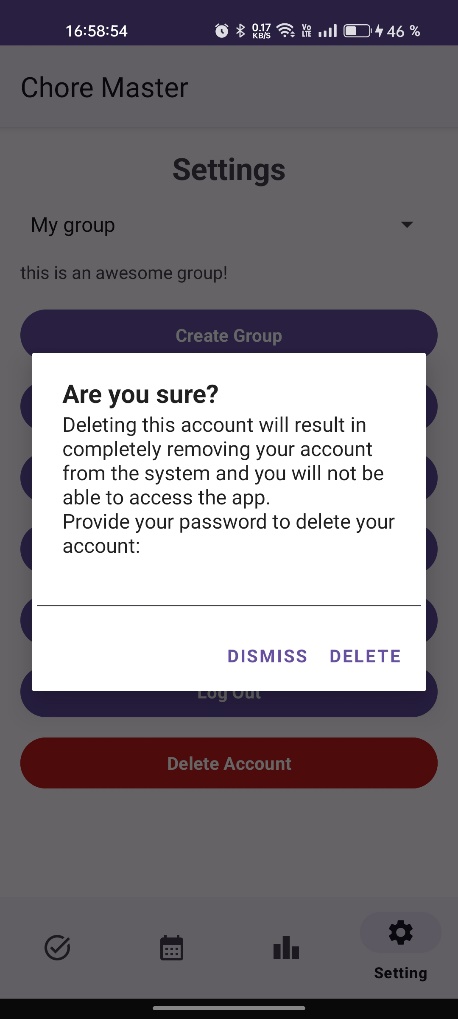
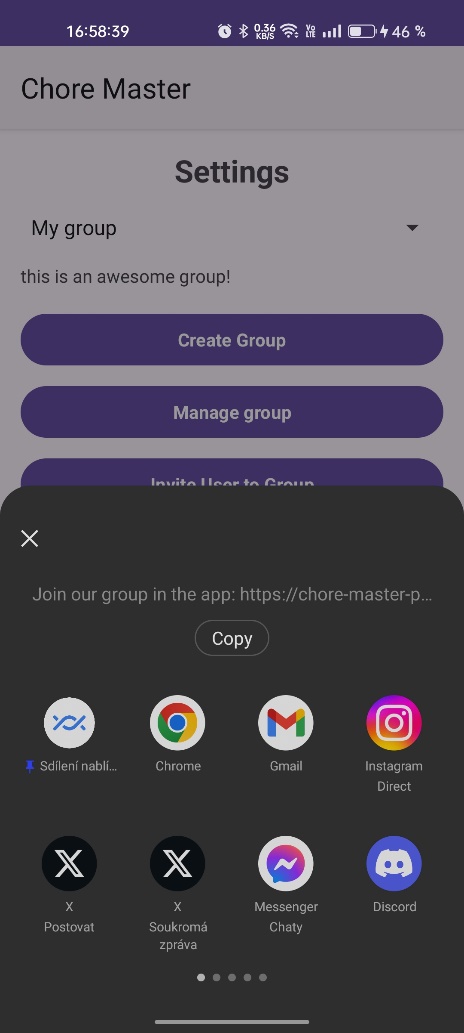
V tomto fragmentu opět dostupného z navigační lišty hlavní aktivity, je uživatel schopen provádět akce týkající se jeho účtu a skupin. Nachází se zde dropdown menu nahoře obrazovky se jmény všech skupin ve kterých je uživatel zapsán s předvybraným jménem momentálně aktivní skupiny. Zde si uživatel přepíná mezi aktivními skupinami. Po vybrání skupiny, se pod daném menu nastaví text s popiskem zvolené skupiny, aby se dalo lépe rozlišit mezi případně stejnojmennými skupinami.

Dále se zde nachází tlačítka na vytvoření skupiny, správu profilu a správu skupiny, které po kliknutí uživatele přesměrují na patřičné aktivity. Ovšem tlačítko správa skupiny bude fungovat pouze v případě, pokud je uživatel adminem v dané skupině, ostatní uživatele to nikam nepustí.

Také se zde nachází tlačítko na pozvání uživatelů do dané skupiny. To po kliknutí vygeneruje odkaz a zobrazí možnosti sdílení. Na fragmentu je i tlačítko pro opuštění skupiny, které uživatele ze skupiny vymaže a také změní aktivní skupinu na první v seznamu.

Dále se zde nachází tlačítko pro odhlášení, které odhlásí uživatele a přesměruje ho na login activitu, bez možnosti vrátit se zpět.

A poslední tlačítko co se nachází na tomto fragmentu je tlačítko na odstranění účtu. To kvůli bezpečnostním důvodům, po kliknutí vyzve uživatele pro zadání hesla v dialogovém okně. A po zadání správného hesla a kliknutí tlačítka „delete“, aplikace uživatelský účet smaže a vyhodí ho ze všech skupin.



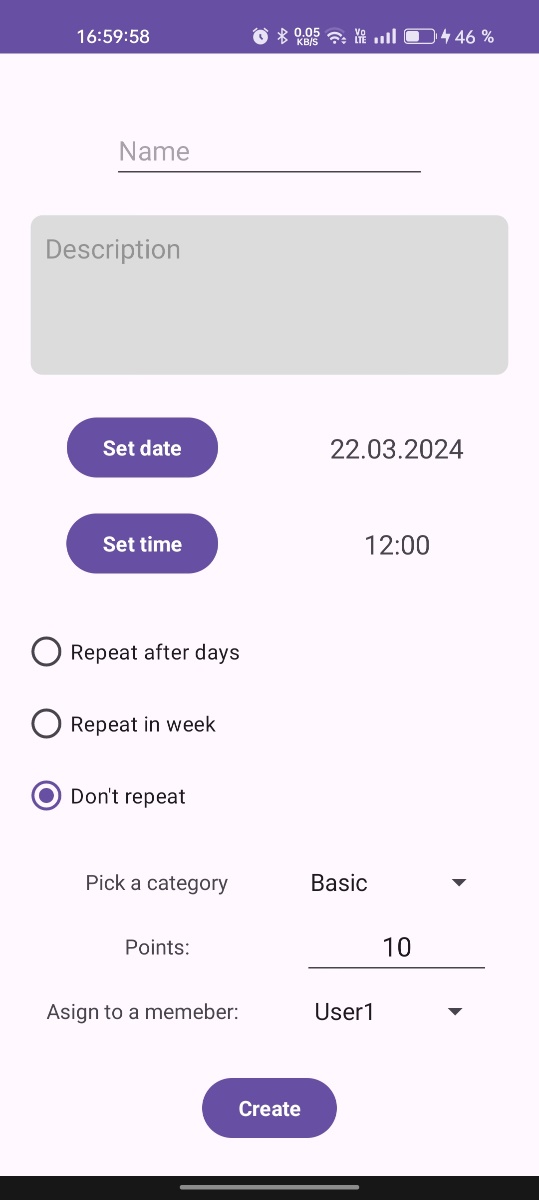
Obrázek 19 Link sharing

Obrázek 20 Konfirmace odstranění účtu

Obrázek 21 Settings fragment

### 4.2.6.Create Task Activity

Na této aktivitě je uživatel schopen vytvářet úkoly pro členy ve zvolené skupině. Uživatel zadává textově jméno úkolu a popis, dále tlačítkem „Set date“ v datumovém dialogu vybere datum, na kdy chce úkol zadat a tlačítkem Set time v časovém dialogu vybere na jaký čas chce úkol zadat. Dále je potřeba aby si uživatel vybral jakým stylem se bude úkol opakovat jestli vůbec. Jsou zde na výběr 3 možnosti. Neopakovat, opakovat v týdnu, opakovat po dnech. Při zvolení první možnosti, není nic dalšího vyžadováno. Ale pokud uživatel vybere opakovat v týdnu, zobrazí se vedle dalších 7 tlačítek pro vybrání v jakých dnes se má úkol opakovat. Podobně, když uživatel vybere možnost opakovat po dnech, tak se místo tlačítek zobrazí textové pole, do kterého uživatel musí zadat počet dní po kolika dnech se bude úkol opakovat.

Po této části, je uživatel povinen vybrat kategorii pro úkol, podle svého uvážení. Dále vybrat počet bodů, které budou uživateli přidány, když úkol splní. A jako poslední vybere, jakému uživateli ve skupině chce úkol přiřadit. Po kliknutí tlačítka „Create“ se úkol vytvoří a začne se zobrazovat v seznamu úkolů, kam je i uživatel ihned přesměrován. 4.2.7.Task Detail Activity

Obrázek 22 Vytvoření úkolu obrazovka

Tato aktivita, podobná aktivitě na vytvoření úkolu, slouží k zobrazení podrobnějších informací o rozkliknutém úkolu. Opět se zde nachází textová pole na zobrazení jména a popisku, dále datum a čas na splnění, způsob opakování a jeho specifikace, vybraná kategorie, body a uživatelské jméno člověka, komu je úkol přiřazen. Také se zde zobrazuje poslední datum splnění úkolu, pokud byl někdy splněn. A po kliknutí tlačítka „show more“ se zobrazí všechny datumy, kdy byl úkol splněn. Ze začátku se aktivita otevře v čtecím režimu, kde je možné si informace pouze prohlédnout a pokud je uživateli, který ten úkol rozklikl i úkol přiřazen, zobrazí se mu tlačítko na plnění úkolu. Dále pokud je uživatel vlastníkem úkolu nebo administrátorem skupiny, tak se nahoře zobrazí i tlačítka „Edit“ a „Delete“. Po kliknutí tlačítka na odstranění, se otevře dialogové okno pro potvrzení akce a po konfirmaci se úkol smaže. A po kliknutí tlačítka na úpravu úkolu, se uživateli umožní všechno na úkolu měnit a místo tlačítka na splnění úkolu se objeví tlačítko na uložení změn.

### 4.2.8.Create Group Activity

Zde si uživatel může vytvořit svoji novou skupinu, zadáním jejího jména a popřípadě i popisku do textových polí co se na této aktivitě zobrazují. Kliknutím tlačítka „Create group“ se skupina vytvoří a automaticky to uživatele na ni přesměruje do „Task fragnent“.

### 4.2.9.Manage Group Activity

Tato aktivita je dostupná pouze pro administrátory skupiny, ostatní členové se na ní nedostanou. Je zde možné měnit název a popis skupiny a kliknutím na tlačítko „Save changes“ změny uložit. Dále tato aktivita zobrazuje seznam členů, které je možné po konfirmaci dialogem ze skupiny vyhazovat, nebo po změnění režimu tlačítkem „Change mode“ přidávat a odebírat práva administrátora.

### 4.2.10.Manage Profile Activity

Na této poslední časti aplikace na kterou ještě nedošlo slovo, si může uživatel měnit svoje veřejné uživatelské jméno zobrazované ostatním a jemu samotnému a bio (popisek profilu).

# 5.Závěr

V závěru této práce lze říci, že aplikace Chore Master úspěšně splňuje svůj účel a nabízí uživatelům efektivní nástroj pro skupinovou správu úkolů. Díky jednoduchému, ale funkčnímu designu a využití moderních cloudových technologií poskytuje aplikace dobrý uživatelský zážitek. Implementace funkcí jako je spolupráce v týmech, správa skupin, správa účtů a úkolů všemožnými způsoby z aplikace dělá nástroj, který dokáže dostatečně uspokojit běžné uživatele. V budoucnu je možné aplikaci dále rozvíjet o nové funkce a vylepšení, které zvýší její užitečnost a přitažlivost pro širší spektrum uživatelů. Tato práce tak představuje solidní základ a přináší do světa aplikací pro správu úkolů novou perspektivu zaměřenou na jednoduchost, efektivitu a spolupráci.

Do tohoto projektu jsem šel s téměř nulovou znalostí Javy a Firebase služeb a úplně nulovou znalostí vytváření aplikací na android. Řekl bych, že jsem si úspěšně rozšířil obzory a naučil se spoustu nových věcí čehož je existence této aplikace důkazem.

# 6.Přílohy

## 6.1.Literatura

Wikipedia contributors. "Java (programming language)." *Wikipedia, The Free Encyclopedia*. Wikipedia, The Free Encyclopedia, 30 Mar. 2024. Web. 31 Mar. 2024.

Extensible Markup Language. (24. 11. 2023). *Wikipedie: Otevřená encyklopedie*. 31. 03. 2024 z <https://cs.wikipedia.org/w/index.php?title=Extensible_Markup_Language&oldid=23405998>

Adekola Olawale „Firebase Authentication“ Medium, 31.03.2024  
<https://medium.com/@Adekola_Olawale/firebase-authentication-413626c5234d>

Wenable „Firebase – Cloud Firestore“ Medium, 31.03.2024 <https://medium.com/@Adekola_Olawale/firebase-authentication-413626c5234d>

Weekly-webtips „Firebase Hosting“ Medium, 31.3.2024 <https://medium.com/weekly-webtips/firebase-hosting-60328efd71cf>

Hackernoon „Learn basic HTML“ Medium, 31.03.2024 <https://medium.com/hackernoon/learn-basic-html-be230361457>

Abdevelops „Building a Simple Reactive System with JavaScript“ Medium, 31.03.2024 <https://medium.com/@abdevelops/building-a-simple-reactive-system-with-javascript-80fb62d67c1b>

AndroidDevelopers „The deep links crash course, Part 1: Introduction to deep links“ Medium, 31.03.2024 <https://medium.com/androiddevelopers/the-deep-links-crash-course-part-1-introduction-to-deep-links-2189e509e269>

## 6.2.Obrázky

[Obrázek 1 Architerktura aplikace 10](#_Toc162820381)

[Obrázek 2 Use case diagram 11](#_Toc162820382)

[Obrázek 3 kolekce groups 12](#_Toc162820383)

[Obrázek 4 subkolekce tasks 12](#_Toc162820384)

[Obrázek 5 kód na vypočítání dalšího dueDate 13](#_Toc162820385)

[Obrázek 6 Firebase authentication interface 14](#_Toc162820386)

[Obrázek 7 kolekce users 14](#_Toc162820387)

[Obrázek 8 kód webové stránky 15](#_Toc162820388)

[Obrázek 9: Ukázka přihlášení uživatele 16](#_Toc162820389)

[Obrázek 10 Login obrazovka 17](file:///C:\Users\Ondra\Desktop\Dokumentace.docx#_Toc162820390)

[Obrázek 11 Register obrazovka 17](file:///C:\Users\Ondra\Desktop\Dokumentace.docx#_Toc162820391)

[Obrázek 12 Zapomenutí hesla obrazovka 17](file:///C:\Users\Ondra\Desktop\Dokumentace.docx#_Toc162820392)

[Obrázek 13 webový formulář na reset hesla 18](file:///C:\Users\Ondra\Desktop\Dokumentace.docx#_Toc162820393)

[Obrázek 14 email na reset hesla 18](file:///C:\Users\Ondra\Desktop\Dokumentace.docx#_Toc162820394)

[Obrázek 15 nastavení jména obrazovka 18](file:///C:\Users\Ondra\Desktop\Dokumentace.docx#_Toc162820395)

[Obrázek 16 tasks fragment 19](file:///C:\Users\Ondra\Desktop\Dokumentace.docx#_Toc162820396)

[Obrázek 17 Calendar fragment 20](file:///C:\Users\Ondra\Desktop\Dokumentace.docx#_Toc162820397)

[Obrázek 18 Stats fragment 21](file:///C:\Users\Ondra\Desktop\Dokumentace.docx#_Toc162820398)

[Obrázek 19 Link sharing 22](file:///C:\Users\Ondra\Desktop\Dokumentace.docx#_Toc162820399)

[Obrázek 20 Konfirmace odstranění účtu 22](file:///C:\Users\Ondra\Desktop\Dokumentace.docx#_Toc162820400)

[Obrázek 21 Settings fragment 22](file:///C:\Users\Ondra\Desktop\Dokumentace.docx#_Toc162820401)

[Obrázek 22 Vytvoření úkolu obrazovka 23](file:///C:\Users\Ondra\Desktop\Dokumentace.docx#_Toc162820402)