



TECHNICKÁ UNIVERZITA V LIBERCI
Fakulta mechatroniky, informatiky
a mezioborových studií ■

Mobilní aplikace pro správu plynových zařízení

Bakalářská práce

Studijní program: B2646 – Informační technologie
Studijní obor: 1802R007 – Informační technologie

Autor práce: **Filip Rubeš**
Vedoucí práce: Ing. Jana Vitvarová, PhD.





TECHNICAL UNIVERSITY OF LIBEREC
Faculty of Mechatronics, Informatics
and Interdisciplinary Studies ■

Mobile application for gas equipment management

Bachelor thesis

Study programme: B2646 – Information technology
Study branch: 1802R007 – Information technology
Author: **Filip Rubeš**
Supervisor: Ing. Jana Vitvarová, PhD.





Zadání bakalářské práce

Mobilní aplikace pro správu plynových zařízení

<i>Jméno a příjmení:</i>	Filip Rubeš
<i>Osobní číslo:</i>	M17000087
<i>Studijní program:</i>	B2646 Informační technologie
<i>Studijní obor:</i>	Informační technologie
<i>Zadávající katedra:</i>	Ústav mechatroniky a technické informatiky
<i>Akademický rok:</i>	2019/2020

Zásady pro vypracování:

1. Seznamte se s problematikou správy plynových zařízení. Komunikujte s reprezentantem revizních pracovníků a sepište požadavky na mobilní aplikaci ulehčující jejich práci.
2. Proveďte rešerši existujících aplikací, které by mohly poskytovat požadovanou funkcionalitu. Specifikujte jejich omezení.
3. Navrhněte vlastní aplikaci na míru. Zaměřte se na intuitivní a příjemné uživatelské rozhraní. Využijte API geolokace a elektronického podpisu.
4. Aplikaci implementujte na operačním systému Android.
5. Aplikaci otestujte v terénu ve spolupráci s revizním pracovníkem.

Rozsah grafických prací:
Rozsah pracovní zprávy:
Forma zpracování práce:
Jazyk práce:

dle potřeby dokumentace
30–40 stran
tištěná/elektronická
Čeština



Seznam odborné literatury:

- [1] VÁVRŮ, Jiří a Miroslav UJBÁNYAI. Programujeme pro Android. Praha: Grada, 2013. ISBN 978-80-247-4863-4.
- [2] PHILLIPS, Bill a Brian HARDY. Android Programming: The Big Nerd Ranch Guide. Big Nerd Ranch Guides, 2013. ISBN 978-0321804334
- [3] 85/1978 Sb. Vyhláška o kontrolách, revizích a zkouškách plynových zařízení.

Vedoucí práce:

Ing. Jana Vitvarová, Ph.D.
Ústav mechatroniky a technické informatiky

Datum zadání práce:

10. října 2019

Předpokládaný termín odevzdání:

18. května 2020

prof. Ing. Zdeněk Plíva, Ph.D.
děkan



Kolář
doc. Ing. Milan Kolář, CSc.
vedoucí ústavu

V Liberci dne 10. října 2019

Prohlášení

Byl jsem seznámen s tím, že na mou bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci (TUL) nezasahuje do mých autorských práv užitím mé bakalářské práce pro vnitřní potřebu TUL.

Užiji-li bakalářskou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti TUL; v tomto případě má TUL právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Bakalářskou práci jsem vypracoval samostatně s použitím uvedené literatury a na základě konzultací s vedoucím mé bakalářské práce a konzultantem.

Současně čestně prohlašuji, že tištěná verze práce se shoduje s elektronickou verzí, vloženou do IS STAG.

Datum: 22.5.2020

Podpis: 

Mobilní aplikace pro správu plynových zařízení

Abstrakt

Tato práce řeší návrh a vývoj mobilní aplikace v operačním systému Android. Hlavním cílem práce je usnadnit a zefektivnit průběh revizí plynových zařízení. Vývoj probíhal v kooperaci s revizním technikem, který definoval a testoval požadované funkcionality. Specifikované požadavky byly implementovány prostřednictvím pomocných API, zajišťující elektronický podpis a geolokaci, a vhodných technologií.

Klíčová slova: mobilní aplikace, Android, Revize, Google Firebase

Mobile application for gas equipment management

Abstract

This thesis solves design and development of mobile application in Android operating system. The main aim of this thesis is to facilitate the process of revisions of gas equipment. The development was carried out in cooperation with a revision engineer who defined and tested the required functionalities. Specified requirements were implemented through auxiliary APIs, such as electronic signature and geolocation, and appropriate technologies.

Keywords: mobile app, Android, Revision, Google Firebase

Poděkování

Děkuji vedoucí mé bakalářské práce Ing. Janě Vitvarové, PhD. za cenné rady a vedení při tvorbě této práce. Dále bych rád poděkoval Martinu Chlumovi za důvěru a možnost vyvíjet aplikaci pro jeho firmu.

Obsah

Seznam zkratek	9
1 Úvod	10
2 Stávající proces zaznamenávání revizí	11
2.1 Nedostatky stávajícího procesu	11
3 Specifikace požadavků	12
3.1 Funkční požadavky	12
3.2 Nefunkční požadavky	13
3.3 Požadavky na systém	13
4 Existující aplikace	14
4.1 INSPECTO	14
5 Návrh řešení	16
5.1 User Experience	16
5.1.1 Wireframe	16
5.2 User Interface	19
5.2.1 Identita	19
5.3 Popis fungování aplikace	21
5.4 Zadávání dat	24
5.4.1 Zadávání datumu	24
5.4.2 Zadávání adresy	24
5.4.3 Specifikace závad	24
5.4.4 Elektronický podpis	24
5.4.5 Export do PDF formátu	24
5.4.6 Odeslání do úložiště	24
5.4.7 Notifikace expirací revizí	25
5.5 Registrace uživatele	25
5.6 Funkce admina	25
6 Implementace	26
6.1 Architektura aplikace	26
6.1.1 Activity	27
6.1.2 Fragment	27
6.1.3 Model	28

6.1.4	Adapter	28
6.2	Funkční prvky	29
6.2.1	Zadávání datumu	29
6.2.2	Definice závad	29
6.2.3	Automatické vyplnění adresy	30
6.2.4	Elektronický podpis	31
6.2.5	Export do formátu PDF	31
6.2.6	Seznam klientů	31
6.2.7	Notifikace	32
6.3	Datový model	32
6.4	Použité technologie	33
6.4.1	Programovací jazyk	33
6.4.2	Vývojové prostředí	33
6.4.3	Úložiště	34
6.4.4	Správa verzí	34
6.4.5	Zpřehlednění kódu	34
6.5	Zabezpečení	35
7	Testování	36
7.1	Testy v průběhu vývoje	36
7.1.1	Android emulátor	36
7.1.2	Mobilní telefon s Android	37
7.2	Testy v reálném použití	37
8	Nasazení	39
8.1	Google Play	39
8.2	Sdílený odkaz	39
9	Závěr	40
A	Přílohy	44
A.1	Formulář revize	44
A.2	Původní protokol	45

Seznam obrázků

2.1	Schéma procesu revize	11
5.1	Část wireframe modelu mobilní aplikace	17
5.2	Zadávání údajů	18
5.3	Vizuální identita aplikace	19
5.4	Barevná paleta	20
5.5	Úvod, Nová a úprava revize	21
5.6	Úvod, Nová a úprava revize	22
5.7	Karta klienta/Náhled dokumentu	23
5.8	Nastavení aplikace	23
6.1	Schéma MVC architektury	26
6.2	Bottom Navigation Bar	27
6.3	User a Client model	28
6.4	Adaptér ClientAdapter	29
6.5	Zadávání datumu/Definice závad	30
6.6	Elektronický podpis	31
6.7	Náhled notifikace	32
6.8	Datový mode	33
7.1	Testování polohy	37
A.1	Formulář revize	44

Seznam tabulek

4.1	Porovnání aplikací	15
5.1	Test wireframe modelu revizním technikem	18
6.1	Popis aktivit	27
7.1	Testování v reálném použití	38

Seznam zkratk

APK	Android Application Package
GPS	Global Positioning System
HTML	Hypertext Markup Language
IDE	Integrated Development Environment
INS	INSPECTO
JSON	JavaScript Object Notation
LIFO	Last In, First Out
MVC	Model-View-Controller
PDF	Portable Document Format
SDK	Software Development Kit
UI	User Interface
URL	Uniform Resource Locator
UX	User Experience

1 Úvod

Počet plynových zařízení v České republice stále stoupá, svědčí o tom tisková zpráva Českého plynárenského svazu.[1] Využití plynu při vytápění zásadním způsobem snižuje počet zdravotních rizik a napomáhá tak k lepší kvalitě ovzduší. Tento fakt potvrzuje počet vyhlášených smogových situací v topné sezóně 2020, který je na číslici 0, kdežto v roce 2017 se jednalo o počet 39.

Revizní kontroly jsou jedním z klíčových faktorů spolehlivého užívání plynových zařízení. Revize se provádí z důvodu ověření bezpečnosti a technického stavu zařízení. Kontrola se na základě vyhlášky č. 85/1978 Sb. provádí jednou za tři roky a to revizním technikem s odpovídajícím oprávněním.[2]

Momentálním modelem provádění revizní kontroly je použití papírových listů, které jsou následně ručně přepisovány do elektronické podoby. Tento model je pro dlouhodobé a časté provádění kontrol velmi nepraktický a zdoluhavý. Zdigitalizováním revizní kontroly dojde k značné úspoře času vykonávání, snadnější manipulaci s protokoly a údaji klienta a bezpochyby učiní kontrolu ekologičtější.

Cílem této práce je vyvinout mobilní aplikaci v koopearaci s revizním technikem, která umožní správu plynových zařízení. Bude možné zadávat revizní data o plynovém zařízení, vygenerovat revizní zprávu a tuto zprávu, včetně dat, odeslat na cloudové úložiště, ke kterému bude mít technik přístup z počítače, či jiného zařízení. Hlavními požadavky na aplikaci jsou zefektivnění práce revizního technika a funkčnost.

V první části práce je popsán stávající proces revize plynových zařízení a jsou specifikovány požadavky na aplikaci. V další části této práce je provedena rešerše existujících aplikací, jež by mohly splňovat zjištěné požadované funkce a jsou upřesněni jejich omezení.

Třetí kapitola popisuje návrh řešení. Je rozdělena na User Experience (UX), User Interface (UI) a popis fungování aplikace. UX řeší vzájemné působení uživatele a produktu, v tomto případě tedy revizního technika a aplikace. Je zde vyobrazen schématický náčrt rozmístění a interakce funkčních prvků, tzv. *wireframe*. UI část řeší uživatelské prostředí aplikace a definuje vizuální identitu. Následným tématem je popis fungování aplikace podložený snímky obrazovky a vymezení vkládání dat.

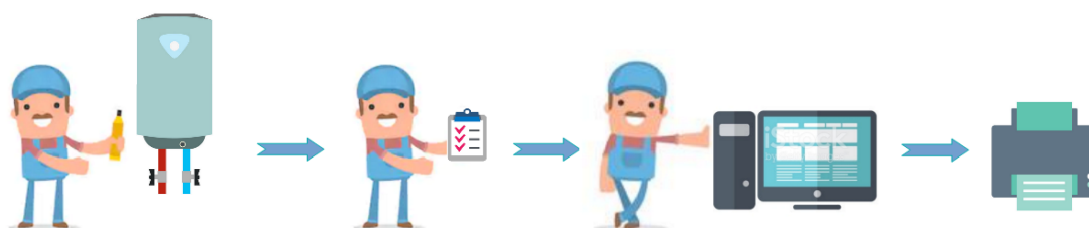
V následující kapitole práce pojednává o implementaci návrhu. Seznamuje s použitou technologií, řeší funkcionalitu a průběh vývoje mobilní aplikace. Poslední kapitola popisuje testování vyvinuté aplikace. Zahrnuje i výsledky testování revizním technikem v reálném prostředí.

2 Stávající proces zaznamenávání revizí

Každá revizní kontrola nyní vyžaduje tři postupné kroky k vytvoření revizního protokolu, které jsou zaznamenány na schématickém nákresu (Obr. 2.1). Revizní technik nejprve při průběhu revize plynového zařízení ručně vyplní formulář (viz příloha Obr. A.1), ve kterém zaznamená osobní údaje klienta, údaje o zapojení zařízení, údaje o spotřebiči, případné závady a stanoví posudek o bezpečnosti zařízení.

Dalším krokem je poté přepsání formuláře do protokolu v elektronické podobě. Výsledný protokol (příloha A.2) v sobě ukrývá všechny právní náležitosti, které stvrzují bezpečné použití plynového zařízení.

Posledním krokem je tisk protokolu a podepsání revizním technikem.



Obrázek 2.1: Schéma procesu revize ¹

2.1 Nedostatky stávajícího procesu

Jako největší nedostatek nýnějšiho modelu průběhu revizí je shledána nutnost vyplnění nového formuláře při každé nové revizi. To značně časově znáročňuje celý proces, v případě provádění revizí v panelovém domě, ve kterém se velká část dat klientů shoduje.

S tím také úzce souvisí spotřeba papírových listů, která se odvíjí od počtu provedených revizí. Další nedostatkem je poté nutnost převedení formuláře do protokolu v elektronické podobě, což je opět časově náročné.

Výše popsané nedostatky nepochybně ztěžují práci revizním technikům a podílí se na odůvodnění vývoje této práce.

¹zdroj ikon: https://www.123rf.com/profile_artenot

3 Specifikace požadavků

Analýza funkčních požadavků je prvním a nejdůležitějším krokem při vývoji mobilní aplikace. Od těchto požadavků se odvíjí náročnost projektu a technický návrh. Je tedy potřeba tyto požadavky správně definovat a pochopit.

3.1 Funkční požadavky

1. Vytvoření revizního protokolu

- (1.1) přidání osobních údajů o klientovi (jméno, tel. číslo, adresa) s využitím aktuální polohy
- (1.2) přidání data a evidenčního čísla revize
- (1.3) upřesnění zapojení zařízení a příslušných spotřebičů
- (1.4) specifikace závad bránících k bezpečnému provozu zařízení
- (1.5) specifikace závad nebránících k bezpečnému provozu zařízení
- (1.6) specifikace závad spotřebiče
- (1.7) doplňující informace (předešlé revize, doporučený termín příští revize..)
- (1.8) elektronický podpis dokumentu zúčastněné osoby ze strany klienta
- (1.9) export protokolu ve formátu PDF se zachovanou původní strukturou revizního protokolu
- (1.10) odeslání protokolu do cloudového úložiště
- (1.11) notifikace blížících se expirací revizí plynových zařízení

2. Správa klientů

- (2.1) vytvoření seznamu klientů
- (2.2) vytvoření klienta na základě vytvořené revize
- (2.3) udržovat kartu klienta s osobními údaji a nejaktuálnější provedenou revizí
- (2.4) možnost úpravy revize
- (2.5) možnost odstranění klienta
- (2.6) odeslání veškerých dat souvisejících s klientem do cloudové databáze

3. Správa uživatelů

- (3.1) registrace uživatele s využitím e-mailové adresy
- (3.2) možnost obnovy hesla uživatele

4. Funkce admina

- (4.1) přístup k veškerým provedeným revizím
- (4.2) přidání uživatelů
- (4.3) odstranění uživatelů
- (4.4) odstranění dat klienta

3.2 Nefunkční požadavky

Nefunkční, nebo také mimofunkční požadavky se netýkají funkcí aplikace, ale vlastností aplikace. Tyto vlastnosti bývají často rozhodujícím faktorem při použití aplikace. Pokud je aplikace nepřehledná a neintuitivní, tak i přes požadovanou funkcionálnost, je nepoužitelná.

- intuitivní a přívětivé uživatelské rozhraní

3.3 Požadavky na systém

Systémové požadavky definují minimální parametry zařízení, jenž musí aplikace podporovat. Mezi systémové požadavky patří požadavky na hardware, software a operační systém.

- zařízení s operačním systémem Android
- podpora verzí Android 5.0 (SDK 21) a vyš
- zařízení s vnitřním úložištěm
- zařízení s internetovým připojením

4 Existující aplikace

Prvotním impulsem pro vývoj mobilní aplikace pro správu plynových zařízení byl fakt, že na dnešním trhu neexistuje žádná aplikace, která by přesně splňovala nezbytné funkcionality, uvedené v kapitole 3. Nicméně jeden produkt je schopen, alespoň z části, požadavkům vyhovět. Jedná se o aplikaci INSPECTO.

4.1 INSPECTO

INSPECTO (INS) je webová aplikace, která je dostupná i ve verzi pro operační systém Android. Oproti této práci má tedy výhodu použití i na stolním počítači, či notebooku. Aplikace by měla být, dle slov vývojářů, dostupná na platformě Google Play, avšak tam nebyla nalezena. Proto nebylo možné produkt osobně odzkoušet a je nutné vycházet pouze z reprezentativních materiálů a návodů na webových stránkách produktu.

Zpozorované rozdíly jsou zaznamenány v tabulce 4.1, kde je možné si všimnout, že aplikace mohou, ze značné části, nabídnout stejné funkce. Avšak zásadním rozdílem a důvodem pro nemožnost použití aplikace INS je v druzích revize. INS nabízí pouze revize elektroinstalace, elektrických strojů, elektrických spotřebičů, zařízení pro obloukové svařování a systému ochrany před bleskem. Proto byla tato aplikace shledána jako nevyhovující k správě plynových zařízení.

Tabulka 4.1: Porovnání aplikací

	INSPECTO	Tato práce
Uživatelská přívětivost	ANO	ANO
Zádání základních údajů (1.1-1.3, 1.7)	ANO	ANO
Specifikace závad (1.4-1.6)	ANO	ANO
Export a uložení protokolu (1.9-1.10)	ANO	ANO
Notifikace revizí (1.11)	Nejspíše ANO	ANO
Editace revizí (2.4)	ANO	ANO
Odstranění klienta (2.5)	ANO	ANO
Odeslání dat do cloudového úložiště (2.6)	ANO	ANO
Registrace uživatele (3.1)	mimo aplikaci	ANO
Obnova hesla uživatele (3.2)	Nejspíše ANO	ANO
Funkce admina (4.1-4.4)	ANO	ANO
Vytvoření klienta na základě revize (2.2)	NE	ANO
Elektronický podpis klienta (1.8)	NE	ANO
Využití aktuální polohy k zadání adresy(1.1)	NE	ANO
Zachování původního protokolu	NE	ANO
Revize plynových zařízení	NE	ANO
Vytvoření klienta v předstihu	ANO	NE
Plánovač budoucích revizí	ANO	NE
Registrace měřících zařízení	ANO	NE
Definice firemních údajů	ANO	NE
Definice databáze závad	ANO	NE
Více typů revizí	ANO	NE

**V závorkách jsou zaznamenány požadované funkcionality.*

5 Návrh řešení

Návrh aplikace je poslední částí vývoje před samotnou implementací. Vychází ze získaných požadavků na produkt a z pozorování uživatele. Součástí návrhu je vytvoření prototypu aplikace, definice a rozmístění funkčních prvků, uživatelské prostředí a vizuální identita.

5.1 User Experience

User Experience (UX) design se zabývá různými aspekty interakce uživatele a aplikace. Často využívaným postupem je pozorování potenciačního uživatele v jeho přirozeném prostředí.

Při takovém pozorování je možné odhalit skutečnosti, na které si respondent při rozhovoru nevzpomene, protože některé úkony zkrátka dělá automaticky. Nesporným přínosem pozorování je také možnost dotazování při aktuálním dění a porozumnění proč určité úkony provádí jedním způsobem a ne druhým. [5]

5.1.1 Wireframe

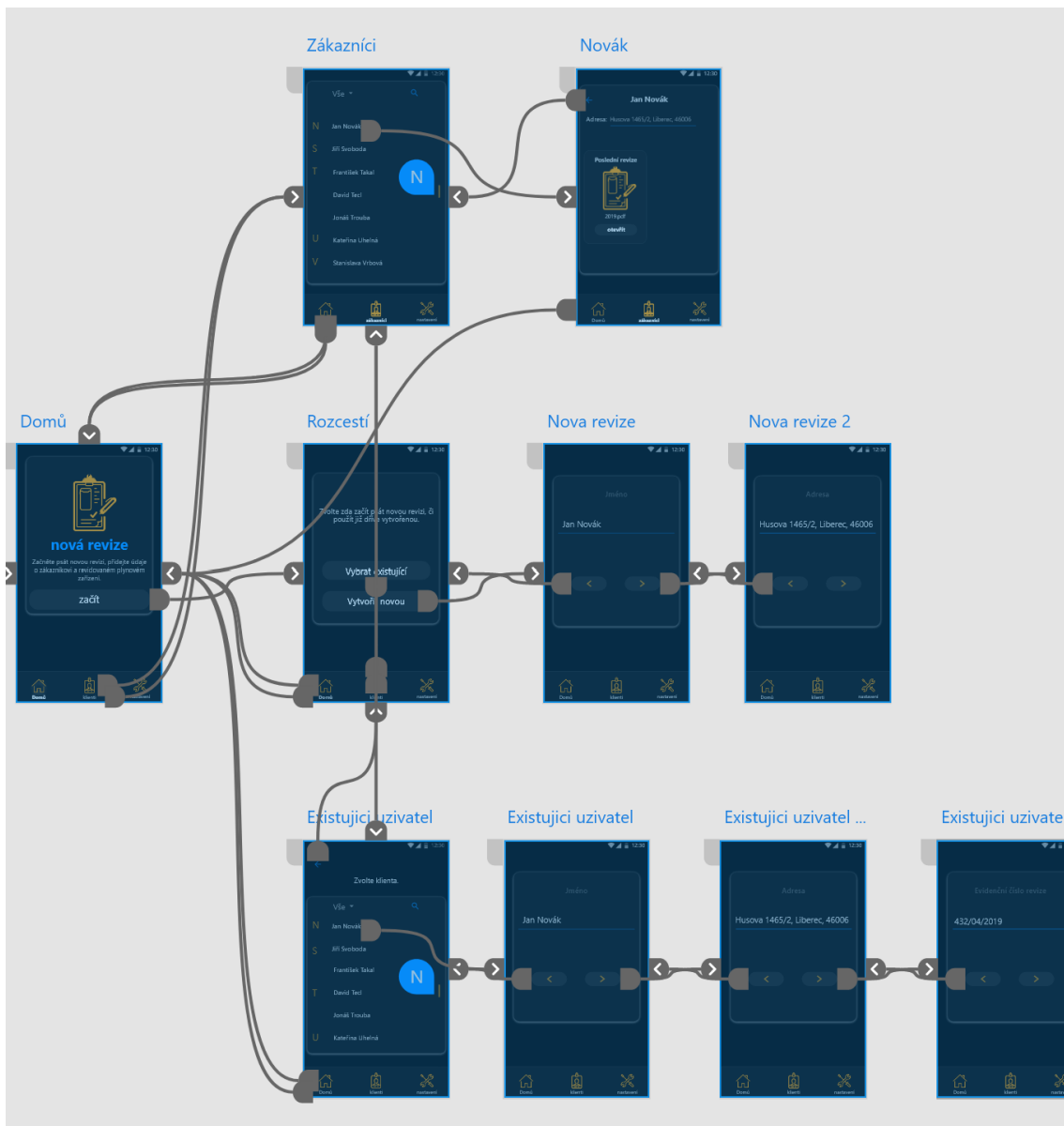
Hlavním nástrojem pro UX design je wireframe, neboli skica aplikace. Používá se při vývoji především jako náhled návrhu řešení, který specifikuje obsah a funkci mobilní aplikace. Wireframe nedefinuje grafickou podobu návrhu, ale pouze rozmístění funkčních prvků.

Může se jednat v podstatě o klikatelný prototyp, který dokáže nasimulovat přibližné fungování aplikace. Velkou výhodou prototypu je možné otestování běhu aplikace na uživatelích bez nutnosti programování aplikace a návrh lze jednoduše libovolně měnit. Důležitou součástí testování je plán úkolů a zápis poznámek při provádění těchto úkolů.

Wireframe se dá vytvořit v mnoha dostupných programech. Existují i webové aplikace, nicméně pro tuto práci byl zvolen program od společnosti Adobe Inc., konkrétně Adobe XD. Na obrázku 5.1 je poté k náhlednutí část wireframe modelu aplikace.

V tomto částečném návrhu je obsaženo pouze základní fungování aplikace, tzn. způsob zadávání dat, zobrazování dat a pohyb po aplikaci. Jsou zde zobrazeny obrazovky aplikace s rozmístěnými funkčními prvky a prvotní designové prvky uživatelského prostředí. Znázorněná propojení definují směr pohybu uživatele po stisku daného tlačítka. Wireframe model přenáší uživatele na domovskou obrazovku, kde

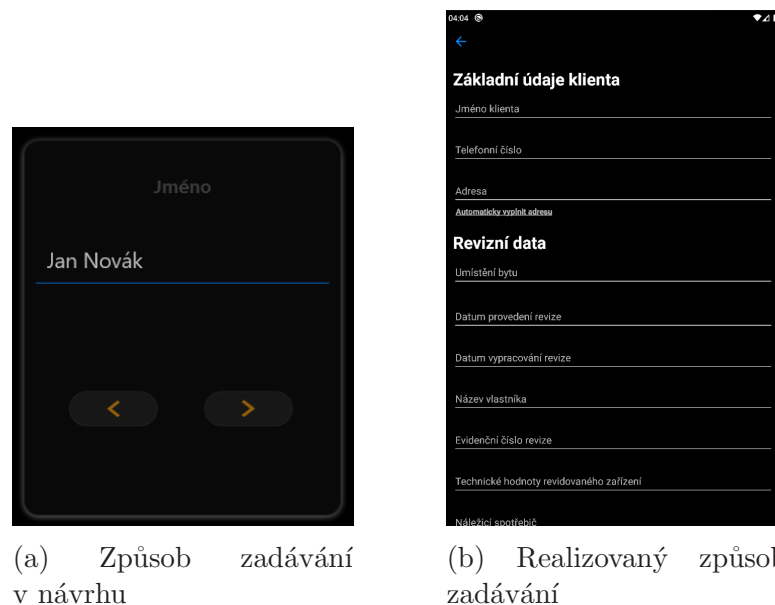
uživatel může začít psát revizi. Poté je dotázán, zda si přeje začít psát novou revizi, či upravit revizi. Při volbě nové revize je přesunut do dotazníkové části, kde zadává potřebné údaje k provedení revize. Při volbě úpravy je přesunut do seznamu klientů, ve kterém si vybere klienta k úpravě. Posléze je přesunut do dotazníkové části.



Obrázek 5.1: Část wireframe modelu mobilní aplikace

Při testování návrhu (viz tabulka 5.1) byly zaznamenávány poznatky od revizního technika, které vedly k úpravám na wireframe modelu. Technik neměl žádný problém s pohybem v aplikaci, dokázal najít a bezchybně otevřít kartu klienta, či se přesměrovat k nové revizi. Technik nebyl zcela spokojen se zadáváním dat, označil způsob za nepřehledný a proto bylo potřeba ho upravit. V původním návrhu se data zadávala postupně pomocí překlikatelných oken, viz obr. 5.2a. Značnou nevýhodou tohoto způsobu byl komplikovaný přechod na údaje zadané v první fázi dotazníku. Bylo potřeba vymyslet novou podobu zadávání, která by umožňovala celkový náhled vložených dat.

Novou variantou byl zvolen dotazník, který se rozprostírá přes celou obrazovku a je možno jej rolovat, tak aby mohl uživatel eventuálně prohlédnout celý dotazník jedním pohybem (Obr. 5.2b).



(a) Způsob zadávání v návrhu

(b) Realizovaný způsob zadávání

Obrázek 5.2: Zadávání údajů

Tabulka 5.1: Test wireframe modelu revizním technikem

úkon	vyhodnocení	poznámka
začít psát novou revizi	Bez problému	
zadávání údajů klienta	Bez problému, s výhradami	nepřehledné, složité navrácení zpět
upravit revizi	Bez problému	
otevřít seznam klientů	Bez problému	
otevřít kartu klienta	Bez problému	
otevřít revizi klienta	Bez problému	
upravit revizi klienta	Bez problému	

5.2 User Interface

User Interface (UI) design je velice významnou částí jakéhokoliv produktu v IT sféře. Jedná se o proces vytváření rozhraní v SW se zaměřením na vzhled nebo styl. Uživatelská rozhraní jsou v podstatě přístupové body, kde uživatel interaguje s návrhem. [6]

Při návrhu uživatelského návrhu této práce nebylo vycházeno z určitého stylu a byl kladen důraz na inovativnost.

5.2.1 Identita

Neméně důležitou součástí vývoje mobilní aplikace je vytvoření jednotné vizuální identity. Mezi jednotlivé prvky identity patří název aplikace, logo, styl písma a barvy.

Název aplikace by nám měl již na první pohled říci o jakou aplikaci se jedná. Neměl by být těžko vyslovitelný a samozřejmě už používaný. Vymyslet název pro tuto práci nebylo v konečném důsledku tak časově náročné, jak se zpočátku zdálo. Zvolen byl název „**Revizor**”. Název se vztahuje k řešenému segmentu, je lehce zapamatovatelný a nedělá problém jej vyslovit.

Při tvorbě loga aplikace se upřednostňovala jednoduchost před složitostí. Logo bylo vytvořeno z minimalistických písmen v kontrastní barvě aplikace, tedy světle modré (viz 5.3a). K aplikacím je nutné také vytvořit ikony. Ty vznikly přidáním černého podkladu a kulatě ohraničenou je možno vidět na obrázku 5.3b.



(a) Logo aplikace



(b) Kulatá ikona aplikace

Obrázek 5.3: Vizuální identita aplikace

Volba písma v mobilní aplikaci je odvozena od stylu písma v logu. Jedná se o písmo „Century Gothic”.

Zvolením barev se formuluje celková podoba výsledného designu a i volba barevného spektra má svá pravidla. Například není vhodné používat kontrastní barvu jako podkladovou, protože by text na ní byl komplikovaně čitelný. Značná část dnešních aplikací používá jako podkladovou barvu bílou, či podobně světlou barvu. Nejnovějším trendem v oblasti UI je přechod na „Dark Mode“, tzn. že mezi základní barvy patří barvy tmavé. Nepopíratelnou výhodou tohoto stylu je úspora energie na baterii mobilních zařízení.

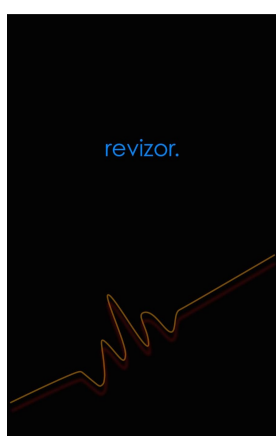
Pro tuto práci byl zvolen právě „Dark Mode“, ovšem nejedná se o jeho klasické provedení. „Dark Mode“ běžně nepoužívá zcela černou podkladovou barvu a volí v odstínech tmavě šedé. V tomto směru je tedy aplikace lehce inovativní a podkladovou barvou je černá. Doplnují ji tři odstíny šedivé, čistě bílá a dvě kontrastní barvy, konkrétně světle modrá a oranžová, která symbolizuje sounáležitost s fakultou mechatroniky, informatiky a mezioborových studií (viz 5.4).



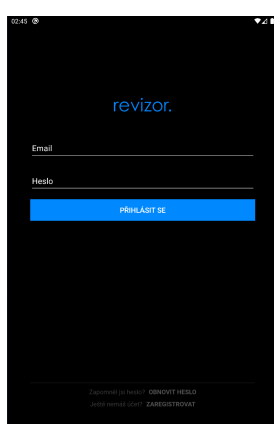
Obrázek 5.4: Barevná paleta

5.3 Popis fungování aplikace

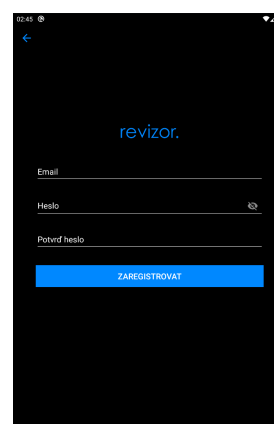
V rámci popisu fungování jsou jednotlivé části doplněny o odkaz na specifikované funkcionality aplikace. Mobilní aplikace neprve zobrazí uvítací obrazovku (Obr. 5.5a) a při prvním spuštění bude zobrazovat přihlašovací stránku (Obr. 5.5b), kde se uživatel po vyplnění dvou hodnot, e-mailu a hesla, přihlásí, pokud již má vytvořený účet. V opačném případě se uživatel bude moct přesunout do registračního formuláře (Obr. 5.5c), kde zadá svou e-mailovou adresu, heslo a následně heslo potvrdí a svůj účet zaregistruje (3.1). Poté bude přesunut zpět na přihlašovací formulář. Zde bude také možnost heslo resetovat (3.2)(Obr. 5.5d). To je vhodné při ztrátě hesla. Aplikace si bude pamatovat přihlášeného uživatele a při dalším spuštění již nebude vyžadováno heslo pro přihlášení.



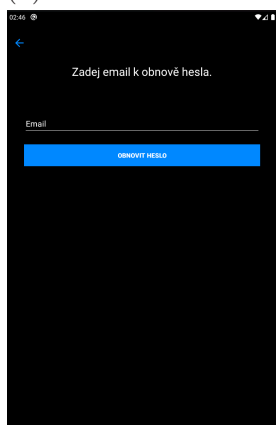
(a) Uvítací obrazovka



(b) Přihlášení uživatele



(c) Registrace uživatele

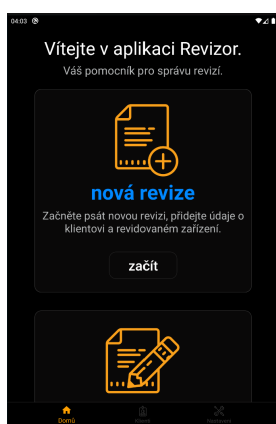


(d) Obnova hesla

Obrázek 5.5: Uvítání a Přihlášení/Registrace uživatele

Na úvodní obrazovce (Obr. 5.6a) se bude v dolní části nacházet tzv. *bottom navigation bar*, což je zjednodušeně menu pro navigaci v aplikaci, které je připevněné. Menu bude nabízet tři destinace. Úvod, list klientů a nastavení. V úvodu si uživatel bude moci zvolit, zda chce začít psát novou revizi nebo chce upravit již dříve vytvořenou. V případě volby pro novou revizi bude přesunut do dotazníku (Obr. 5.6b) pro zadání potřebných údajů (1.1-1.8) k vygenerování revize (1.9). Při úspěšném vygenerování revize se uživatel rozhodne, zda protokol uloží a odešle do cloudového úložiště (1.10, 2.6), či se navrátí zpět pro úpravu. Po uložení protokolu se vytvoří karta klienta (2.2).

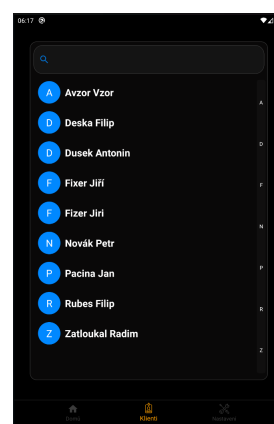
Po zvolení úpravy revize bude zobrazen seznam klientů (2.1) (Obr. 5.6c) a uživatel zvolí, kterému chce upravit revizi. Následně bude zobrazen dotazník s předvyplněnými daty (Obr. 5.6d) a uživatel se může rozhodnout, která data chce upravit, či neupravit (2.4). Poté bude vygenerován protokol (Obr. 5.6e) a uživatel jej bude moci uložit a odeslat na cloudové úložiště, nebo uložení zruší a navrátí se zpět do dotazníku.



(a) Úvodní obrazovka



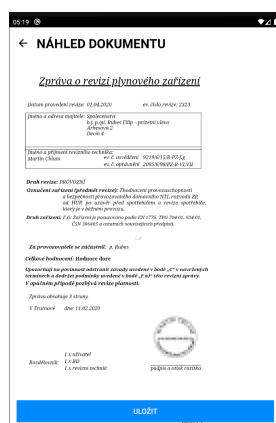
(b) Nová revize



(c) Seznam klientů



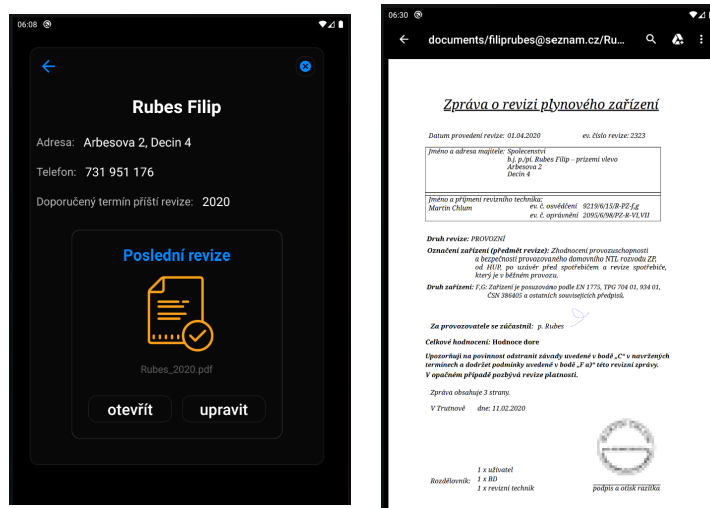
(d) Úprava revize



(e) Uložení protokolu

Obrázek 5.6: Úvod, Nová a Úprava revize

Na druhé destinaci se bude nacházet seznam klientů (Obr. 5.6c), ve kterém bude možné vyhledávat pomocí hledacího okénka. Při kliknutí na jméno klienta se zobrazí karta klienta (2.3) (Obr. 5.7a). Tato karta bude obsahovat adresu klienta, telefonní číslo klienta, doporučené datum pro příští revizi a bude zde možné zobrazit nejaktuálnější revizi (Obr. 5.7b) a případně být přesunut k úpravě revize. Karta bude obsahovat i tlačítko pro možnost odstranění klienta a všech jeho dat (2.5).

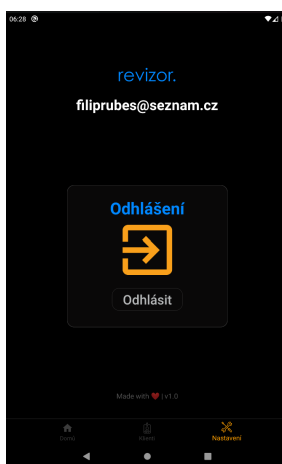


(a) Karta klienta

(b) Náhled dokumentu

Obrázek 5.7: Karta klienta/Náhled dokumentu

V poslední destinaci (Obr. 5.8) se bude nacházet zobrazení aktuálně přihlášeného uživatele a bude možné se zde odhlásit a přesunout k přihlašovací obrazovce.



Obrázek 5.8: Nastavení aplikace

5.4 Zadávání dat

Součástí zadávání dat je návrh a popis způsobů získávání požadovaných informací k úspěšnému provedení revizní kontroly.

5.4.1 Zadávání datumu

Uživateli se při kliknutí na kolonku datumu zobrazí kalednář s aktuálním datem a uživatel si poté zvolí požadované datum.

5.4.2 Zadávání adresy

Uživatel při vyplňování klientovy adresy bude mít na výběr, zda ji vyplní ručně, či využije detekce aktuální polohy. Detekování adresy bude zajištěno pomocí reverzního geokódování, což znamená převod GPS souřadnic na adresu. Zjištění aktuálních GPS souřadnic zařízení bude získáno díky GPS modulu v mobilním zařízení, s případnou kooperací s internetovým připojením.

5.4.3 Specifikace závad

Uživatel bude mít možnost vyplnit zjištěné závady ručně, nicméně z důvodu úspory času bude mít také možnost zadat závady výběrem z předpřipravených. Při kliknutí na kolonku se uživateli zobrazí dialog se závadami a po zvolení požadovaných se tyto závady zobrazí v náležití kolonce. Při zvolení závad, pro které je nutné vyplnit i nejzazší datum odstranění se zobrazí kalendář k zadání.

5.4.4 Elektronický podpis

Po úspěšném vyplnění všech požadových informací se na konci dotazníku bude nacházet prostor pro elektronický podpis. Zadávání bude zajištěno pomocí zobrazené obrazovky určené ke kreslení. Výsledek kreslení bude možné uložit, či v případě nespokojenosti smazat. Poté bude výsledek uložen jako obrázek a vložen do protokolu

5.4.5 Export do PDF formátu

Získaná data budou umístěna na patřičná místa v předpřipraveném dokumentu, který bude ve formátu HTML a poté bude dokument exportován do formátu PDF pomocí vhodné knihovny.

5.4.6 Odeslání do úložiště

Odeslání dokumentu do úložiště bude probíhat ihned po uložení protokolu. Bude odesláno do úložiště v cloudové platformě na účet, kterým byl uživatel přihlášen

do aplikace. Celý tento proces bude probíhat pomocí nástrojů zvolené cloudové platformy.

5.4.7 Notifikace expirací revizí

Uživatele bude aplikace notifikovat v případě blížící se expirace revize. Což znamená, že při ukládání revize se také bude ukládat datum pro oznámení. Přesně jeden měsíc před expirací revize bude uživateli tato skutečnost oznámena.

5.5 Registrace uživatele

Při prvotním spuštění aplikace bude zobrazena obrazovka pro vyplnění přihlašovací údajů. Registrace uživatele bude možná překlikem na registrovací obrazovku, kde uživatel zadá e-mailovou adresu a heslo(3.1). Registrovat uživatele bude také možné přes portál cloudové platformy, ovšem tuto možnost bude mít pouze admin aplikace, viz 5.6. Přes zmíněný portál bude probíhat i autentizace pomocí e-mailu. Mobilní aplikace bude nabízet, v případě potřeby, možnou obnovu hesla. Obnova bude probíhat pomocí zasláního tokenu na e-mailovou adresu uživatele(3.2).

5.6 Funkce admina

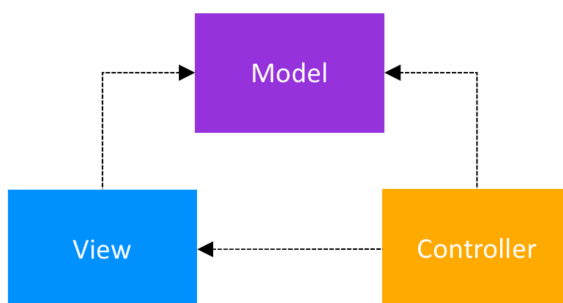
Na portálu cloudové platformy bude mít admin přístup k veškerým provedeným revizím(4.1) a bude mít možnost přidat, odstranit, či upravit uživatele(4.2, 4.3) a bude mu umožněno operovat nad daty klientů(4.4).

6 Implementace

Po zdárném návrhu řešení aplikace je dalším krokem implementace. Tato kapitola se zabývá architekturou aplikace, implementací funkčních prvků, jako je automatické vyplnění adresy, či elektronický podpis a dále jsou zde popsány použité technologie při vývoji aplikace.

6.1 Architektura aplikace

Tato práce je vyvíjena s využitím architektury MVC, která vyžaduje tři základní komponenty: *Model*, *View* a *Controller*. MVC architektura je vhodná pro použití v případě, že „logika“ uživatelského prostředí je měněna častěji než „business logika“ a je zapotřebí oddělení funkcí uživatelského prostředí. [7]



Obrázek 6.1: Schéma MVC architektury, zdroj: [7]

V této práci je za *View* považován Fragment (6.1.2), *Controller* je poté Activity (6.1.1) a *Model* je Model (6.1.3).

6.1.1 Activity

Aktivita představuje jednu obrazovku aplikace, jenž interaguje s uživatelem. V aplikaci se nachází jedna hlavní aktivita (*MainActivity*), která se zobrazuje uživateli ihned po spuštění aplikace. Aktivita může spouštět další aktivity, přičemž se předchozí aktivita zastaví a odešle do zásobníku, který funguje na principu LIFO (Last in - First out). V případě stisku tlačítka „zpět“ se obnoví ze zásobníku poslední vložená aktivita.[3]

V tabulce 6.1 jsou vyjmenovány a krátce popsány všechny aktivity této práce.

Tabulka 6.1: Popis aktivit

Název aktivity	Popis
ClientCardActivity	Zobrazuje kartu klienta
ClientsActivity	Zobrazuje seznam klientů
EditFormActivity	Zobrazuje úpravu dotazníku
FormActivity	Zobrazuje dotazník
LoginActivity	Zobrazuje přihlašovací formulář
MainActivity	Zobrazuje úvodní obrazovku
ResetPasswordActivity	Zobrazuje formulář k obnově hesla
SignUpActivity	Zobrazuje registrační formulář
WebViewActivity	Zobrazuje náhled revize
WelcomeActivity	Zobrazuje uvítací obrazovku

6.1.2 Fragment

Fragment představuje úplné nebo částečné uživatelské prostředí aktivity (*Activity*), včetně kompetentních metod. Může být také součástí jiného fragmentu, či aktivity.[4]

V této práci byl fragment využit k snadnější implementaci spodní navigační lišty (*Bottom Navigation Bar*) (Obr. 6.2), který slouží k navigaci po aplikaci. Celkem aplikace obsahuje tři základní fragmenty: *HomeFragment* - domovská stránka, *ClientFragment* - seznam klientů a *SettingsFragment* - stránka nastavení.

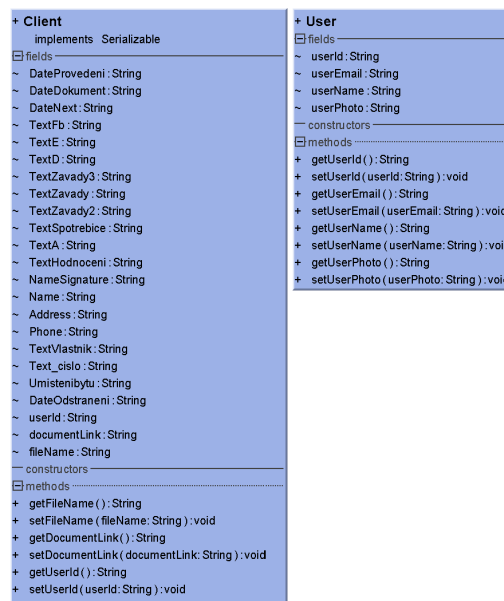


Obrázek 6.2: Bottom Navigation Bar

6.1.3 Model

Model je držitelem dat aplikace, jinými slovy modeluje věci, kterými se aplikace zabývá. Model se nikterak nezabývá uživatelským prostředím a jeho jedinou úlohou je uchovávání a správa dat.[4] V této práci jsou naimplementovány dva modely, konkrétně *Client* model a *User* model. Na obrázku 6.3 jsou vyobrazeny oba modely, jejich proměnné a metody.

Model *Client* v sobě uchovává všechny informace klienta, jako jsou jméno, adresa, či telefonní číslo, ale také informace k provedené revizní kontrole. Model poskytuje metody, které můžou s daty klienta pracovat. Na obr. 6.3 nejsou uvedeny všechny metody. *User* model v sobě uchovává základní informace o uživateli - identifikační číslo, e-mailovou adresu a příslušné metody.



Obrázek 6.3: Část User a Client modelu

6.1.4 Adapter

Adaptér je objekt kontroleru, který se, pomyslně, nachází mezi listem a daty, které má list zobrazit. Hlavní úlohou adaptéru je vytvoření potřebných *ViewHolderů*, jenž popisují zobrazení a umístění položek v listu, a tyto *ViewHoldery* svázat s daty z *Modelu*.[4]

V této práci je použit pouze jeden adaptér - *ClientAdapter*(Obr. 6.4), který slouží k vytvoření seznamu klienta a definuje, jak seznam bude vypadat a interagovat s uživatelem. Adaptér je rozšířen pomocí implementace *Filterable*, která zajišťuje filtraci položek a *SectionIndexer*, která slouží k rychlému rolování mezi abecedními sekcemi v seznamu.

```

+ ClientAdapter extends RecyclerView.Adapter
  implements Filterable
  SectionIndexer
fields
- final mContext: Context
- final mInflater: LayoutInflater
- mItem: ArrayList<Client>
- live_path_string: ArrayList<String>
- live_name_string: ArrayList<String>
- mSectionPositions: ArrayList<Integer>
constructors
+ ClientAdapter(c: Context)
methods
+ onCreateViewHolder(parent: ViewGroup, viewType: int): ItemHolder
+ getItemViewType(position: int): int
+ onBindViewHolder(holder: ItemHolder, position: int): void
+ getItemCount(): int
+ update(_list: ArrayList<Client>): void
+ updateList(newList: ArrayList<Client>): void
+ getFilter(): Filter
+ getSections(): Object[]
+ getPositionForSection(i: int): int
+ getSectionForPosition(i: int): int

```

Obrázek 6.4: Adaptér ClientAdapter

6.2 Funkční prvky

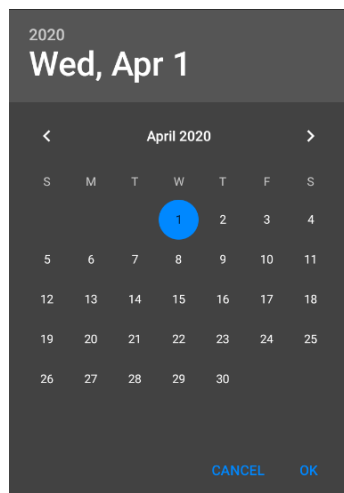
Tato podkapitola se zabývá implementací funkčních prvků, které přispívají k zajištění požadovaných funkcionalit. Jsou zde popsány zvolené pomocné knihovny a třídy.

6.2.1 Zadávání datumu

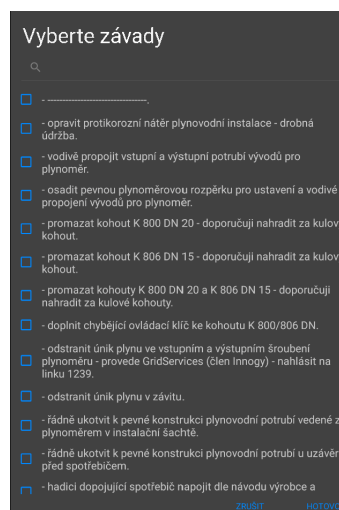
K zadávání datumu byl využit *DatePickerDialog* (Obr. 6.5a), jenž pomocí abstraktní třídy *Calendar* umožňuje volit datum z kalendáře. K získání aktuálního data je využita metoda třídy *Calendar*: *getInstance*. Nicméně revizní dotazník vyžaduje více formátů data a proto bylo nutné vytvořit *DatePickerDialogy* s různými formáty, které byly definovány pomocí třídy *SimpleDateFormat*.

6.2.2 Definice závad

MultiSelectDialog (Obr. 6.5b) je knihovna, která umožňuje vytvořit seznam prvků z nichž uživatel vybírá. *Dialog* disponuje vyhledávacím okénkem, který využívá zvýrazňování textu. Při implementaci bylo nutné definovat pravidla výběru, jako například minimální a maximální počet vybraných položek a také seznam položek k výběru pomocí *MultiSelectModelu*. [8]



(a) Zadávání datumu



(b) Volba závad pomocí MultiSelectDialogu

Obrázek 6.5: Zadávání datumu/Definice závad

6.2.3 Automatické vyplnění adresy

Zavedení automatického vyplnění adresy na základě aktuální polohy je řešeno pomocí *Location Manager*, což je třída, jež poskytuje přístup ke službám určování polohy zařízení. Tyto služby umožňují aplikacím získat pravidelné aktualizace geografické polohy zařízení nebo být upozorněny, když se zařízení nachází v blízkosti dané geografické polohy. Třída *LocationManager* generuje objekt *Location*, který představuje geografické umístění. Všechna umístění generovaná nástrojem *LocationManager* mají zaručenou platnou zeměpisnou šířku, délku a časové razítko.

Výše uvedené třídy zajišťují aktuální pozici zařízení popsanou pomocí zeměpisné šířky a délky. K převedení těchto souřadnic na adresu je využit *Geocoder*. Jedná se o třídu pro manipulaci s geokódováním a reverzním geokódováním. Geokódování je proces transformace adresy ulice nebo jiného popisu místa na souřadnice (zeměpisná šířka, délka). Reverzní geokódování je proces opačný, tzn. transformace souřadnic na adresu. K úspěšnému použití těchto metod jsou vyžadována oprávnění, konkrétně *ACCESS_COARSE_LOCATION* a *ACCESS_FINE_LOCATION*. [4]

6.2.4 Elektronický podpis

K implementaci elektronického podpisu byla využita knihovna *DrawView*[9], která není primárně určena k vytváření elektronického podpisu, nýbrž k prostému malování. Nicméně v této práci byla využita pouze její elementární funkčnost, tedy zaznamenávání pohybu prstu po vymezeném prostoru bez možnosti modifikace kreslicího nástroje. Vymezený prostor je doplněn o tři, takzvané *FloatingActionButton*, které umožňují uložení podpisu, vymazání podpisu a zrušení výkonávání podpisu (Obr. 6.6). Takto vytvořený podpis je poté uložen formou bitmapového obrázku, který je následně exportován do formátu PNG a zakódován v rámci Base64.



Obrázek 6.6: Elektronický podpis

6.2.5 Export do formátu PDF

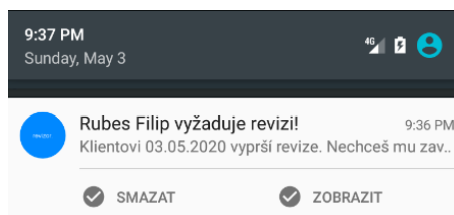
Původním zamýšleným postupem při exportování revizního protokolu do PDF formátu bylo využití určitých knihoven pro tvorbu PDF dokumentů. Nicméně při řešení takových knihoven nebyly shledány žádné, které by byly bezplatně dostupné, respektive by splňovaly požadované funkcionality pro tvorbu revizních protokolů. Z toho důvodu po úspěšném vyplnění všech informací v revizního dotazníku je vytvořen revizní protokol v HTML formátu, který je následně pomocí knihovny *PDF Generator*[10] exportován do formátu PDF.

6.2.6 Seznam klientů

Pro vytvoření přehledného a intuitivního seznamu klientů (Obr. 5.6c) byla využita knihovna *AlphabetIndex FastScroller* pro Android *RecyclerView*[11]. Tato knihovna rozšiřuje klasický *RecyclerView* o možnost vyhledávání položek pomocí vyhledávacího okénka a rychlému přechodu mezi abecedními sekcemi.

6.2.7 Notifikace

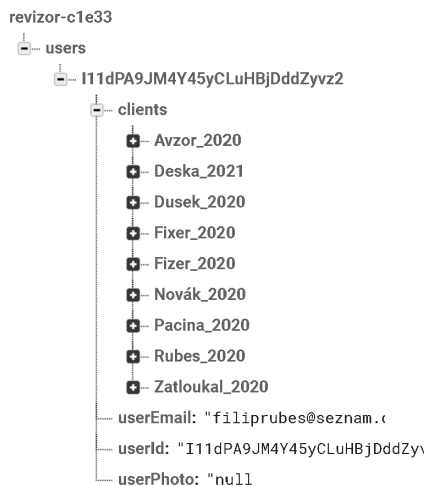
Notifikace blížících se expirací revizních kontrol je naimplementována využitím knihovny *NotifyMe*[12], pomocí které se definuje termín notifikace, její znění, tlačítka a cesty po jejich kliknutí. Oznámení o blížící se expiraci se uživateli objeví přesně jeden měsíc před vypršením revize, aby měl uživatel dostatek času o tom informovat svého klienta, v případě, že se klient již neobjednal na novou revizní kontrolu. Notifikace (Obr. 6.7) zobrazuje jméno klienta a datum expirace. Notifikaci je možné odstranit, či se přesunout do aplikace.



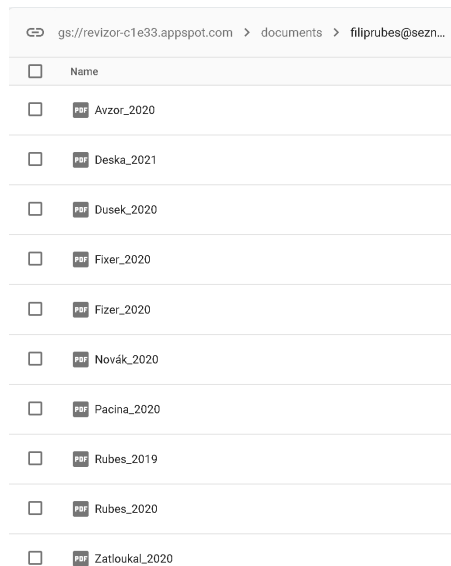
Obrázek 6.7: Náhled notifikace

6.3 Datový model

Mobilní aplikace využívá služeb Google Firebase, dále v 6.4.3. Ty umožňují komunikaci databáze s aplikací, přičemž je využívána stromová struktura databáze (Obr. 6.8a) ve formátu JSON. Kořenovým uzlem databáze je její název, který obsahuje uživatele aplikace, jako uzel první úrovně. Uživateli je při registraci přidělen unikátní klíč, který slouží jako uzel druhé úrovně, jenž v sobě uchovává uživatelovu e-mailovou adresu a uzel třetí úrovně - uživatelovi klienty. Každý klient má vytvořený vlastní uzel čtvrté úrovně v němž jsou všechny údaje uloženy pomocí typu *String*. Každý klient také obsahuje URL odkaz na vyexportovaný revizní protokol, jenž se nachází v *Firebase Storage* (Obr. 6.8b). Dokumenty jsou ukládány na základě vytvoření klientova uzlu a v případě úpravy, či smazání klienta je dokument upraven, respektive smazán. Nicméně v případě úpravy klienta, která slouží k obnovení revizní kontroly, tzn. že se mění rok provedení, je smazán původní klientův uzel, ale nikoliv původní dokument klienta. Jinými slovy v databázi se udržují pouze aktuální data klientů a ve *Storage* se nachází aktuální i předchozí dokumenty.



(a) Firebase databáze



(b) Firebase storage

Obrázek 6.8: Datový model

6.4 Použité technologie

Práce byla implementována s využitím vhodných pomocných technologií. V této podkapitole jsou popsány jednotlivé technologie a jejich využití v aplikaci.

6.4.1 Programovací jazyk

Java je objektově orientovaný programovací jazyk, jenž vychází z jazyka C a C++. Mezi jeho hlavní znaky patří jednoduchost, robustnost, přenositelnost a bezpečnost.[13]

Pro vývoj mobilní aplikace v operačním systému Android je možné volit ze dvou jazyků - Javy a Kotlinu. Nicméně z důvodu předešlých zkušeností s jazykem Java nebylo o volbě druhé možnosti uvažováno.

6.4.2 Vývojové prostředí

K vývoji mobilních aplikací v operačním systému Android je optimální využít Integrated Development Environment (IDE) od společnosti Google, tedy Android Studio. Tento program umožňuje nejen programovat aplikace nativně pro Android, avšak také obsahuje Software Development Kit (SDK) Managera, díky kterému je možné nainstalovat SDK, což je sada vývojových nástrojů. Zásadní výhodou tohoto vývojového prostředí je integrovaný Android emulátor, jenž slouží ke spuštění aplikace na zařízení a je možné následné testování na více verzích Androidu. [3]

V této práci bylo Android Studio využito jako primární vývojové prostředí a aplikace byla pomocí tohoto programu testována na verzích Android 5.0 až 10.0.

6.4.3 Úložiště

Google Firebase[14] je cloudová platforma od společnosti Google. Tato platforma slouží jako pomocný nástroj při vývoji iOS, Android a webových aplikací. Nabízí hned několik možných využití. Mezi jeho základní funkce patří realtime databáze, která je typem NoSQL a nevyžaduje konstantní připojení k internetu. Cloudové úložiště je možné využívat jen do určité velikosti paměti. Ta je určena na základě zvoleného cenového plánu. Firebase dále disponuje statistickými údaji využití aplikace, či autentizací uživatelů.

Google Firebase byl v této práci značně využíván. Registrace uživatelů je řešena přes autentizační portál s využitím e-mailové adresy. K vytvoření databáze uživatelů a klientů je využita funkce Realtime databáze, zejména z důvodu kontinuálně probíhajícího přenosu dat mezi uživateli a využití pouze jedné databáze. K vývoji aplikace byl zvolen startovací program, který nabízí využití až 1 GB úložiště, který při velikosti nahrávaných souborů 60 až 85 Kb, je dostačující. Aplikace využívá také možnosti automatického zasílání tokenů pro obnovu hesla, jejichž znění je předdefinované adminem.

6.4.4 Správa verzí

Ke správě zdrojových kódů aplikace byla využívána webová služba GitHub.[15] Tato služba spadá pod společnost Microsoft a používá verzovací nástroj Git. Mezi hlavní možnosti využití této služby patří interakce programátora s uživateli, kdy je možné sledovat reportované problémy, včetně požadavků na nové funkcionality. Dále také možnost vytváření dokumentace a nebo například hledání rozdílů mezi verzemi kódu.

V této práci byl využíván zejména jako ochrana před ztrátou dat. Nahrávání souborů do repozitáře bylo prováděno průběžně až v druhé polovině vývoje aplikace. Repozitář je definován jako privátní a není tedy přístupný veřejnosti.

6.4.5 Zpřehlednění kódu

Při vývoji mobilní aplikace je vhodné využít knihovny, které zpřehledňují kód a usnadňují jeho zápis. *ButterKnife*[16] je knihovna, jenž využívá *Annotation Processing*, což je proces zpracování anotací, který ověřuje jejich správné použití. Zjednodušuje práci s View prvky, které se nachází aktivitách a nebo také s událostmi *OnClick*, které definují událost po stisku tlačítka.

6.5 Zabezpečení

Mobilní aplikace, o které pojednává tato práce, je vyvíjena výhradně pro jednu společnost, a proto nebude možné její stažení z jakékoliv distribuční služby a její zdrojové soubory nebudou přístupné veřejnosti. Z toho důvodu není aplikace ošetřena propracovanějším zabezpečením a ochrana dat je na úrovni ochrany platformy Firebase a jsou pouze definována pravidla pro zápis a čtení databáze.

7 Testování

Testování programu je při vývoji mobilní aplikace a jiných SW podstatnou částí budování programu. Aplikace byla testována nespočetněkrát během vývoje. Ovšem takové testování nedokáže nahradit použití v reálném procesu. Proto je nezbytně nutné otestovat aplikaci uživatelem, který ji poté bude denně používat, aby byly nalezeny případné nedostatky a chyby.

7.1 Testy v průběhu vývoje

Při vývoji aplikace ve vývojovém prostředí Android Studio (6.4.2) se nabízí dvě možnosti testování aplikace. Aplikace je možné v kteroukoliv chvíli nainstalovat do emulátoru, či na mobilní telefon a aktuální verzi programu otestovat.

Obě varianty mají své kladné a záporné stránky. Mezi výhody testování aplikace na emulátoru patří možnost sledování dění na konzoli. Do konzole je možné vypisovat pomocné hlášení při běhu aplikace. To může usnadnit nalezení případných problémů. Dále emulátor umožňuje testování aplikace na různých zařízeních s různými verzemi operačního systému. Naopak jeho nevýhodou může být neautentická interakce s uživatelem a časté potíže při instalaci.

V případě testování na mobilním zařízení, který není aktuálně připojen k počítači, není možnost sledování běhu aplikace a případných chyb v konzoli. Naopak je zde možnost autentického použití aplikace, kdy je možné odhalit vzhledové nedokonalosti a vyzkoušet práci s aplikací na reálném zařízení.

7.1.1 Android emulátor

Pro testování byl zvolen emulátor, který se nachází přímo v programu Android Studio. Tento emulátor je nejdoporučovanější pro využití, v případě programování ve vývojovém prostředí Android Studio. Pro možné testování je nutné mít nainstalovaný SDK pro Android. K tomu napomáhá SDK Manager umístěný v Android Studiu. Nicméně instalace SDK může být v určitých případech problematická, protože je nutné dodržet jistá pravidla, například lokace SDK musí obsahovat pouze ASCII znaky.

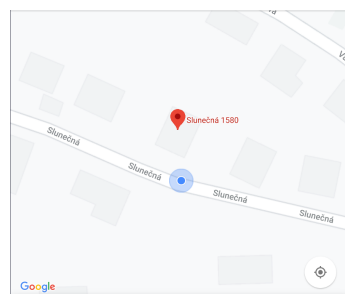
K testování byly zvoleny nejnovější systémy Android 9.0 s SDK 28 a Android 10.0 s SDK 29. Při testování nebyly shledány žádné funkční ani vzhledové chyby. Rozlišení bylo zvoleno na základě volby revizního technika, který požadoval rozlišení pro tablet, konkrétně se jednalo o rozlišení 1200 x 1920 px. Na tomto rozlišení se

aplikace zobrazovala bezchybně. Práce byla testována i na menším rozlišení, které však v určitých fázích nesplňovala požadovaný vzhled. Přizpůsobení aplikace i pro menší zařízení se nachází v možných rozšířeních aplikace.

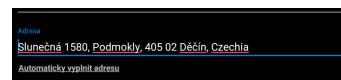
7.1.2 Mobilní telefon s Android

Pro testování aplikace na reálném zařízení nebylo možné využít zařízení s větším rozlišením. Využíván byl mobilní telefon s menším rozlišením, Huawei Mate 20 Pro s rozlišením 3120 x 1440 px a operačním systémem Android 10.0. Aplikace se na zařízení v určitých částech nezobrazovala správně a testování sloužilo z větší části pouze k posouzení fungování specifikovaných požadavků.

Některé funkce na Android emulátoru nebylo možné dostatečně vyzkoušet. Jednalo se kupříkladu o automatické vyplnění adresy v dotazníku, kde Android emulátor nabízel pouze lokaci v Severní Americe, Kalifornii. U mobilního telefonu se poté otestovalo správné posouzení při změně polovice, viz Obr. 7.1.



(a) Aktuální polovice na mapě



(b) Automaticky vyplněná adresa

Obrázek 7.1: Testování polovice

7.2 Testy v reálném použití

Po dokončení aplikace byla testována revizním technikem v reálném použití. K testování měl původně sloužit technikův tablet s úhlopříčkou 7" a rozlišením 1200 x 1920 px a operačním systémem Android 7.1. Nicméně z důvodu defektu na zařízení bylo nutné využít technikův mobilní telefon Lenovo Z2 s úhlopříčkou 5" a rozlišením 1080 x 1920 px a operačním systémem Android 5.0.2. Značnou výhodou při využití toho zařízení bylo možné otestování chodu a podpory aplikace na výrazně starší verzi systému Android. Aplikace fungovala bez jakýchkoliv obtíží a bylo možné ji otestovat v reálném použití. Testování v praxi se věnuje tabulka 7.1, kde jsou zaznamenány hodnocené oblasti testování.

Mezi testované oblasti patří intuitivnost, vzhled, funkčnost, výkon, podporovatelnost, bezpečnost a použitelnost aplikace. Revizní technik hodnotil aplikace obdobně jako ve škole, tzn. známkami 1-5 a případnými komentáři. Technik označil uživatelské rozhraní aplikace za velmi intuitivní a příjemné. Byl spokojen s funkčností aplikace,

jenž přesně splňuje specifikované požadavky. Dále hodnotil výkon aplikace, u kterého měl menší výhrady při načítání seznamu klientů. To však bylo způsobeno nejspíše nízkou kvalitou připojení v daném momentu, jelikož při změně pozice načítání nebylo nikterak zdoluhavé. Dalším bodem byla bezpečnost aplikace, kde byla hodnocena zejména technická odolnost, například na vstupech aplikace. Technik uvedl celkovou spokojenost s bezpečností, nicméně podotkl nutné seznámení s pravidly vstupu, které jsou však připomenuty v případě zadání nesprávného vstupu.

Následně revizní technik zhodnotil podporovatelnost aplikace, kdy neměl žádné obtíže při instalaci aplikace a hardwarovou či softwarovou konfigurací. Finálním hodnocením byla použitelnost aplikace v reálném použití. Technik vyhodnotil aplikaci jako zdařilou a vhodnou pro zefektivnění a ulehčení práce revizním technikům.

Tabulka 7.1: Testování v reálném použití

Oblast testování	Hodnocení*	Komentář
Intuitivnost	1	
Vzhled	1	
Funkčnost	1	
Výkon	1-	dlouhé načítání
Bezpečnost	1	pravidla vstupů
Podporovatelnost	1	
Použitelnost	1	

**Hodnocení jako ve škole 1-5, 1 nejlepší, 5 nejhorší.*

8 Nasazení

Ve chvíli, kdy je aplikace vyvinuta a náležitě otestována, je finálním krokem její publikace. Android aplikace je možné nasadit dvěma způsoby - na *Marketplace* nebo mimo něj.

8.1 Google Play

Google Play [17] je platforma sloužící k publikaci, distribuci a monetizaci aplikace po celém světě. Google Play nabízí analýzu stahování aplikace a napomáhá ke kontrole distribuce a identifikaci trhu. K nasazení aplikace na této platformě jsou však potřebné přípravné úkony, mezi které patří snímky obrazovky, grafika a další materiály k propagaci aplikace. Dále je nutné definovat parametry aplikace, jež slouží ke kategorizaci aplikace, a v neposlední řadě také země a jazyky, pro které bude aplikace dostupná.

8.2 Sdílený odkaz

V případě rozhodnutí nenabízet aplikaci na nějakém z *Marketplace* je možná její distribuce pomocí webových stránek nebo serveru. K tomu stačí vytvořit Android Application Package (APK) a ten umístit na web, či server a poskytnout uživatelům odkaz na stažení. APK soubor poté stačí stáhnout do zařízení. Nicméně pro automatickou instalaci je nutné nejprve na zařízení potvrdit instalace aplikací z portálu třetích stran.

Tato práce je distribuována pomocí sdíleného linku. Důvodem je použití aplikace pouze jednotkami uživatelů a celkové nastavení aplikace je tomuto faktu uzpůsobeno. Aplikace nedisponuje autentizací e-mailových adres u registrovaných uživatelů a v případě umístění aplikace na distribuční platformy a možného stažení aplikace kýmkoliv, by mohlo dojít k nadbytečnému vytížení databáze.

9 Závěr

Hlavním cílem této práce bylo navrhnout a vyvinout mobilní aplikaci, jež by zefektivnila práci revizním technikům. Prvními kroky při návrhu aplikace byly seznámení se s problematikou správy plynových zařízení. To bylo docíleno kooperací s revizním technikem a obeznámením se stávajícím procesem zaznamenávání revizí. Přitom byly definovány nedostatky aktuálního procesu a funkční, nefunkční a systémové požadavky na aplikaci.

Na základě specifikace funkčních a nefunkčních požadavků byla provedena rešerše existujících řešení. Při této rešerši byla zkoumána aplikace INSPECTO, jež by mohla splňovat požadované funkcionality. Nicméně aplikace nebyla schopna nabídnout základní požadavek, čímž byla podpora revizí plynových zařízení a proto bylo zapotřebí navrhnout a vytvořit novou aplikaci vyhovující nárokům.

V úvodu návrhu aplikace byl vytvořen wireframe model, jež byl výsledkem pozorování revizního technika v přirozeném prostředí. Dalším krokem bylo zpětné testování, při kterém byly zjištěny nedostatky na wireframe modelu a bylo nutné jej upravit. Po vyhovujícím návrhu wireframe modelu byla vytvořena vizuální identita aplikace, která se skládá z loga, písma, ikon, barevné palety a byl definován název „Revizor”. Na základě vizuální identity bylo vytvořeno uživatelské prostředí aplikace, jež bylo podporou při popisu fungování aplikace. Posledním krokem návrhu bylo navrhnout a popsat způsob plnění požadavků.

Aplikace byla vyvinuta v programovacím jazyce Java s využitím architektury MVC. V práci byly popsány části architektury a jejich konkrétní využití v aplikaci. K implementaci funkčních prvků byly využity například pomocné knihovny *NotifyMe*, jež slouží k vytváření notifikací a *PDF Generator* pro export dokumentu do formátu PDF.

V průběhu vývoje byla mobilní aplikace soustavně testována, což přispívalo k nalezení případných chyb ve funkčnosti, či grafickém vzhledu. Po dokončení byla aplikace testována revizním technikem, jehož zpětná vazba byla zaznamenána v dotazníku. Z té je patrné, že aplikace splňuje všechny funkční a nefunkční požadavky a je vhodným nástrojem pro zefektivnění práce revizních techniků.

Během implementační a testovací fáze této práce byla zaznamenána možná rozšíření v dalších verzích aplikace. Vhodným rozšířením aplikace by byla možnost vytvoření kalendáře s termíny sjednaných revizí. Aktuálně aplikace uživatele pouze upozorňuje na blížící se expiraci revizní kontroly, kdy tuto skutečnost může oznámit klientovi. Implementace plánovače revizí by měla za důsledek shromáždění veškeré aktivity, spojené s revizní kontrolou, na jedno místo.

Nynější verze aplikace je zaměřena pouze na jeden druh revize. V případě rozšíření o další druhy revizí se může aplikace stát prospěšnou i pro jiné revizní techniky.

Jak už bylo uvedeno v 8, aktuálně není v plánu aplikaci zveřejňovat na jakékoliv distribuční platformě. Je tak provedeno zejména z důvodu bezpečnosti. Dalším důvodem je prozatimní využití pouze malým počtem uživatelů a instalace aplikace na jednotky zařízení. V případě provedání aktualizací je možná instalace on-line. Ovšem upřednostňováno bude osobní setkání, což zaručuje bezproblémový proces a utužování vztahů s uživateli.

Nicméně v případě rozšíření aplikace pro využití vícero uživateli a případném zveřejnění na některé z distribučních platform by bylo nutné implementovat ověření e-mailové adresy při registraci uživatele, které by zamezilo eventuálnímu zahlcení databáze irelevantními uživateli.

Literatura

- [1] *Tisková zpráva - Češi dýchají čistší vzduch. Pomáhá přechod na vytápění plynem* [online], Český plynárenský svaz, [cit. 2020-04-07]. Dostupný z WWW: https://www.cgoa.cz/pages/pdfdoc/aktuality/2020/TZ_CPS_CHMU.pdf
- [2] Vyhláška č. 85/1978 Sb. *o kontrolách, revizích a zkouškách plynových zařízení*. Sbírka zákonů, 26.06.1978.
- [3] VÁVRŮ, Jiří a Miroslav UJBÁNYAI, *Programujeme pro Android*. Grada, Praha, 2013. ISBN: 978-80-247-4863-4
- [4] PHILLIPS, Bill a Brian Hardy, *Android Programming: The Big Nerd Ranch Guide*. Big Nerd Ranch Guides, 2013. ISBN: 978-0321804334
- [5] *UX testování* [online], Ackee, [cit. 2020-04-07]. Dostupný z WWW: <https://www.ackee.cz/blog/ux-testovani/>
- [6] *User Interface (UI) Design* [online], Interaction design foundation, [cit. 2020-05-01]. Dostupný z WWW: <https://www.interaction-design.org/literature/topics/ui-design>
- [7] *Android Architecture Patterns Part 1: Model-View-Controller* [online], Medium, [cit. 2020-05-01]. Dostupný z WWW: <https://cutt.ly/dyjaZlW>
- [8] *Android Multi Select Dialog* [online], GitHub, [cit. 2020-05-03]. Dostupný z WWW: <https://github.com/abumoallim/Android-Multi-Select-Dialog>
- [9] *DrawView* [online], GitHub, [cit. 2020-05-03]. Dostupný z WWW: <https://github.com/ByoxCode/DrawView>
- [10] *PDF Generator* [online], GitHub, [cit. 2020-05-03]. Dostupný z WWW: <https://github.com/UttamPanchasara/PDF-Generator>
- [11] *Fast Scroll RecyclerView* [online], GitHub, [cit. 2020-05-03]. Dostupný z WWW: <https://github.com/myinnos/AlphabetIndex-Fast-Scroll-RecyclerView>
- [12] *NotifyMe* [online], GitHub, [cit. 2020-05-03]. Dostupný z WWW: <https://github.com/jakebonk/NotifyMe>
- [13] *What is Java technology and why do I need it?* [online], Java, [cit. 2020-04-09]. Dostupný z WWW: https://java.com/en/download/faq/whatis_java.xml

- [14] *Google Firebase* [online], Google, [cit. 2020-04-09]. Dostupný z WWW: <https://firebase.google.com/docs>
- [15] *Github* [online], Microsoft, [cit. 2020-04-20]. Dostupný z WWW: <https://github.com/about>
- [16] *ButterKnife* [online], GitHub, [cit. 2020-05-04]. Dostupný z WWW: <http://jakewharton.github.io/butterknife/>
- [17] *Publish your app* [online], GitHub, [cit. 2020-05-20]. Dostupný z WWW: <https://developer.android.com/studio/publish>

A Přílohy

A.1 Formulář revize

Datum: 25.2.		Jméno: BENOLOVA ALEA (53)		Patro: 4. VLEVO VLEVO			
Adresa: 444		Zhodnocení		je schopno <input checked="" type="checkbox"/>		není schopno	
HUP	K800 DN <input checked="" type="checkbox"/>	KK DN <input checked="" type="checkbox"/>	materiál:	ocel	-měď	stoupačka:	ocel <input checked="" type="checkbox"/> měď <input checked="" type="checkbox"/>
Z01	opravit protikorozní nátěr plynovodní instalace - drobná údržba		Z11	řádně ukotvit plynovodní potrubí u uzávěru před spotřebičem			
Z02	vodivě propojit vstupní a výstupní potrubí vývodů pro plynoměr		Z12	hadici dopojující spotřebič napojit dle návodu výrobce a předpisů			
Z03	osadit pevnou plynoměrovou rozpěrku pro ustavení a vodivě		Z13	provést výměnu hadice dopojující spotřebič - překr. životnost 10 let			
Z04	promazat kohout K 800 DN 20 - dop. nahradit za kulový kohout		Z14	provést výměnu hadice dopojující spotřebič - život. 10 let, požár od.			
Z05	promazat kohout K 806 DN 15 - dop. nahradit za kulový kohout		Z15	provést výměnu hadice dopojující spotřebič, nebo protipož. armat.			
Z06	promazat kohouty K 800 DN 20 a K 806 DN 15 - dop. nahradit		Z16	doporučuji dopojit spotřebič hadicí pro lepší přístup a méně spojů			
Z07	doplnit chybějící ovládací klíč ke kohoutu K 800/806 DN.....		Z17	zhotovit kontr. a odvětr. otvory ve strop. podhledu - kontr., údr.			
Z08	odstranit únik ve vstupním a výstupním šroubení plynoměru		Z18	volně zpřístupnit rozeb. spoje potr. vedeného za - kontr., údr.			
Z09	odstranit únik plynu v závitě		Z19	volně zpřístupnit uzávěr před spotřebičem - ovládání, kontr., údr.			
Z10	řádně ukotvit plynovodní potrubí vedené v instal. šachtě		X	Z20	rozvod bez zjevných závad		
Spotřebič: HORA 2479			uzávěr: <input checked="" type="checkbox"/>	umístění: <input checked="" type="checkbox"/>			
Z21	celý spotřebič vyčistit a seřadit		Z25	vypálené rozdělovače plamene - vyměnit			
Z22	vyčistit a promazat ovládací kohoutky spotřebiče		Z26	řádně upevnit uvolněný ovládací knoflík.....hořáku			
Z23	zprovoznit termo-elektrickou pojistku plamene		Z27	doplnit chybějící ovládací knoflík.....hořáku			
Z24	zprovoznit piezo-zapalování hořáků spotřebiče		Z28	vzhledem ke stavu a stáří spotřebiče doporučuji výměnu			
			X	Z29	bez zjevných závad		
Zajistit:						Podpis:	

Obrázek A.1: Formulář revize

A.2 Původní protokol

Zpráva o revizi plynového zařízení

Datum provedení revize: 15.1.2020 ev. číslo revize: 590/01/2020

Jméno a adresa majitele:	Bytové družstvo Šestidomí b.j. p. Fizer Jiří – přízemí vlevo Papírenská 590 Trutnov
Jméno a příjmení revizního technika:	Martin Chlum
	ev. č. osvědčení 9219/6/15/R-PZ-f,g ev. č. oprávnění 2095/6/98/PZ-R-VI,VII

Druh revize: PROVOZNÍ

Označení zařízení (předmět revize): Zhodnocení provozuschopnosti a bezpečnosti provozovaného domovního NTL rozvodu ZP, od HUP, po uzávěr před spotřebičem a revize spotřebiče, který je v běžném provozu.

Druh zařízení: F,G: Zařízení je posuzováno podle EN 1775, TPG 704 01, 934 01, ČSN 386405 a ostatních souvisejících předpisů.

Za provozovatele se zúčastnil: p. Fizer

Celkové hodnocení: Zařízení je schopno bezpečného a spolehlivého provozu.

Upozorňuji na povinnost odstranit závady uvedené v bodě „C“ v navržených termínech a dodržet podmínky uvedené v bodě „F a)“ této revizní zprávy. V opačném případě pozbývá revize platnosti.

Zpráva obsahuje 3 strany.

V Trutnově dne: 22.1.2020

Rozdělovník: 1 x uživatel
1 x BD
1 x revizní technik



podpis a otisk razítka

TECHNICKÁ ZPRÁVA

- A. Technické hodnoty revidovaného zařízení
- B. Údaje o měření a zkouškách
- C. Zjištěné závady a nedostatky, návrh opatření a lhůt k odstranění.
- D. Údaje o odstranění závad z předchozích revizí, příp. kontrol.
- E. Záznam o ostatních revizích provedených na zařízení.
- F. Ostatní

A: Odběrní plynové zařízení je napojeno ze stávajícího provozovaného stoupačního NTL plynovodu odbočkou pr. 28, která je provedena v instalační šachtě na WC bytové jednotky. Tato odbočka je rovněž v šachtě na WC ukončena kulovým kohoutem DN 25, který slouží jako HU odběrního plynového zařízení. Za ním jsou provedeny vývody pro plynoměr. Od plynoměru je rozvod veden instalační šachtou, odkud poté prochází do kuchyně, kde je potrubí ukončeno uzávěrem, za kterým je dopojen spotřebič. Kubatura kuchyně je pro daný spotřebič vyhovující. Kuchyň je přímo větratelná a průtok vzduchu je zajištěn možností občasného nebo trvalého otevření okenního křídla do venkovního prostoru. Rozvod je proveden z trubek měděných spojovaných lisováním. Celý rozvod je ukotven v konzolách a třmenech.

Spotřebiče:

- 1 ks plynová deska Electrolux - umístěn v kuchyni
- kulový kohout DN 15

- B:**
- Kontrola těsnosti přístupných rozebiratelných spojů revidovaného zařízení byla provedena detektorem plynu Sewerin Snooper H mini - v.č. 02211002907
 - Kontrola funkce a úplnosti ovládacích a bezpečnostních armatur na zařízení
 - Kontrola spotřebičů (ovládací a bezpečnostní prvky, funkce)
 - Kontrola výskytu CO v místnosti detekčním přístrojem Testo T310 v.č. 42803364

Výsledek – zjištěné závady a případné úniky plynu uvedeny v bodě „C“ této revizní zprávy.

C: Závady bránící bezpečnému provozu

Závady nebránící bezpečnému provozu:

- provést výměnu hadice dopojující spotřebič - překročená životnost 10 let 20.6.2020
- Plynová deska Electrolux**
- bez zjevných závad

D: Závady z předchozí revize byly částečně odstraněny.

E: Ostatní doložené revize:

- zápis z předchozí provozní revize provedené na zařízení z roku 2017 – RT Martin Chlum

F: a) pro zajištění správné a bezpečné funkce spotřebičů je nutné udržovat dostatečný přívod spalovacího vzduchu do místnosti. Toto je zajištěno v technické zprávě popsáním opatřením - udržovat tyto cesty průchozí - odpovídá majitel (uživatel).

b) K revizi bylo doloženo:

- zápis z předchozí provozní revize provedené na zařízení z roku 2017 – RT Martin Chlum

K revizi nebylo doloženo:

- zápis o vpuštění plynu a seznámení obsluhy s provozem spotřebičů dle návodu výrobce-dohledat

- zpráva o výchozí revizi plynového zařízení – dohledat

- doklad o servisní prohlídce spotřebiče - dohledat

- projektová dokumentace – dohledat

Doporučený termín příští revize v roce 2022.

Tímto revizní zpráva končí.

Vypracoval: Filip Chlum

Kontroloval: Martin Chlum