



## Diplomová práce

### Aplikace pro revizní zprávy

*Studijní program:*

N2612 Elektrotechnika a informatika

*Studijní obor:*

Informační technologie

*Autor práce:*

**Bc. Filip Rubeš**

*Vedoucí práce:*

Mgr. Jiří Vraný, Ph.D.

Ústav nových technologií a aplikované informatiky

Liberec 2023



## Zadání diplomové práce

### Aplikace pro revizní zprávy

<i>Jméno a příjmení:</i>	<b>Bc. Filip Rubeš</b>
<i>Osobní číslo:</i>	M20000172
<i>Studijní program:</i>	N2612 Elektrotechnika a informatika
<i>Studijní obor:</i>	Informační technologie
<i>Zadávající katedra:</i>	Ústav nových technologií a aplikované infor- matiky
<i>Akademický rok:</i>	2021/2022

#### Zásady pro vypracování:

1. Seznamte se s problematikou tvorby progresivních webových aplikací a problematikou tvorby revizních zpráv pro plynová zařízení, včetně aktuální legislativy.
2. Na základě získaných znalostí navrhnete další rozšíření existujícího prototypu aplikace pro tvorbu revizních zpráv pro plynová zařízení.
3. Vytvořený návrh prakticky implementujte, a vytvořte progresivní webovou aplikaci pro tvorbu revizních zpráv.

*Rozsah grafických prací:* dle potřeby dokumentace  
*Rozsah pracovní zprávy:* 40-50 stran  
*Forma zpracování práce:* tištěná/elektronická  
*Jazyk práce:* Čeština

### **Seznam odborné literatury:**

- [1] ATER, Tal. Building progressive web apps: bringing the power of native to the browser. Beijing: O'Reilly, 2017. ISBN 978-1491961650.  
[2] MDN JavaScript [online]. Mozilla foundation [cit. 2021-10-1]. Dostupné z:  
<https://developer.mozilla.org/en-US/docs/Web/JavaScript>

*Vedoucí práce:* Mgr. Jiří Vraný, Ph.D.  
Ústav nových technologií a aplikované informatiky

*Datum zadání práce:* 12. října 2021  
*Předpokládaný termín odevzdání:* 22. května 2023

prof. Ing. Zdeněk Plíva, Ph.D.  
děkan

L.S.

Ing. Josef Novák, Ph.D.  
vedoucí ústavu

V Liberci dne 19. října 2021

## Prohlášení

Prohlašuji, že svou diplomovou práci jsem vypracoval samostatně jako původní dílo s použitím uvedené literatury a na základě konzultací s vedoucím mé diplomové práce a konzultantem.

Jsem si vědom toho, že na mou diplomovou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci nezasahuje do mých autorských práv užitím mé diplomové práce pro vnitřní potřebu Technické univerzity v Liberci.

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

Současně čestně prohlašuji, že text elektronické podoby práce vložený do IS/STAG se shoduje s textem tištěné podoby práce.

Beru na vědomí, že má diplomová práce bude zveřejněna Technickou univerzitou v Liberci v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů.

Jsem si vědom následků, které podle zákona o vysokých školách mohou vyplývat z porušení tohoto prohlášení.

## Aplikace pro revizní zprávy

### Abstrakt

Tato práce řeší návrh a vývoj webové aplikace pro revizní zprávy, která usnadňuje celkovou agendu revizního technika. Webová aplikace využívá principy progresivních webových aplikací a je naimplementována pomocí JavaScriptové knihovny React. Vývoj probíhal na základě definovaných požadavků a provedeného návrhu UX a UI. Aplikace byla po implementaci řádně otestována revizním technikem v reálném použití.

**Klíčová slova:** PWA, webová aplikace, revize, plynová zařízení, React, Firebase

## Application for revision reports

### Abstract

This thesis solves design and development of a web application for revision reports, which makes the overall agenda of revision technician easier. The web application uses the principles of progressive web applications and was implemented with JavaScript React library. Development is based on defined requirements and executed UX and UI design. After implementation, the application was properly tested by a revision technician in real use.

**Keywords:** PWA, web application, revision, gas devices, React, Firebase

## Poděkování

Děkuji vedoucímu mé diplomové práce Mgr. Jiřímu Vranému, PhD. za cenné rady a vedení při tvorbě této práce. Dále bych rád poděkoval Filipu Chlumovi za poskytnutí informací týkajících se problematiky plynových zařízení a asistenci při testování aplikace.

# Obsah

Seznam zkratek . . . . .	11
<b>1 Úvod</b>	<b>12</b>
<b>2 Analýza</b>	<b>14</b>
2.1 Plynová zařízení . . . . .	14
2.2 Revize plynových zařízení . . . . .	16
2.2.1 Výchozí revize . . . . .	17
2.2.2 Provozní revize . . . . .	19
2.3 Progresivní webové aplikace . . . . .	20
2.3.1 Pilíře . . . . .	20
2.3.2 Manifest . . . . .	20
2.3.3 Service Workers . . . . .	20
2.3.4 Cache . . . . .	22
2.3.5 Výhody PWA . . . . .	23
2.3.6 Nevýhody PWA . . . . .	25
<b>3 Požadavky na aplikaci</b>	<b>26</b>
3.1 Funkční požadavky . . . . .	26
3.2 Ostatní požadavky . . . . .	28
<b>4 Návrh řešení</b>	<b>29</b>
4.1 User Experience a User Interface . . . . .	29
4.1.1 Kontextuální rozhovor . . . . .	29
4.1.2 Prototyping . . . . .	30
4.1.3 Testování použitelnosti . . . . .	33
4.1.4 Vizuální identita . . . . .	35
4.1.5 Grafické uživatelské rozhraní . . . . .	37
4.2 Revizní zpráva . . . . .	38
4.2.1 Provozní revize . . . . .	38
4.2.2 Výchozí revize . . . . .	40
4.2.3 Zápis o tlakové zkoušce . . . . .	40
<b>5 Implementace</b>	<b>41</b>
5.1 Použité technologie . . . . .	41
5.1.1 Frontendová část . . . . .	41
5.1.2 Backendová část . . . . .	43

5.2	Architektura aplikace . . . . .	45
5.2.1	Frontend . . . . .	45
5.2.2	Backend . . . . .	47
5.3	Funkční prvky . . . . .	48
5.3.1	Autentizace uživatelů . . . . .	48
5.3.2	Nejčastější závady na zařízení . . . . .	49
5.3.3	Definování závad . . . . .	49
5.3.4	Elektronický podpis a razítko . . . . .	49
5.3.5	Export revizní zprávy . . . . .	49
5.3.6	Uložení a načtení vzoru zpráv . . . . .	50
5.3.7	Odesílání revizních zpráv pomocí emailu . . . . .	50
5.3.8	Evidence revizních zpráv . . . . .	50
5.3.9	Využití map . . . . .	50
5.3.10	Kalendář . . . . .	50
5.3.11	Statistiky . . . . .	51
5.4	PWA prvky . . . . .	52
5.4.1	Instalace aplikace . . . . .	52
5.4.2	Notifikace . . . . .	52
5.4.3	Offline mód . . . . .	53
5.5	Datový model . . . . .	54
5.6	Zabezpečení . . . . .	54
5.6.1	Frontendová část . . . . .	54
5.6.2	Backendová část . . . . .	55
<b>6</b>	<b>Testování</b>	<b>56</b>
6.1	Testování v průběhu vývoje . . . . .	56
6.2	Uživatelské testování . . . . .	57
6.2.1	Průběh testování . . . . .	57
6.2.2	Výsledek testování . . . . .	57
6.3	PWA testování . . . . .	58
6.3.1	Lighthouse . . . . .	59
<b>7</b>	<b>Závěr</b>	<b>60</b>
	<b>Použitá literatura</b>	<b>62</b>
<b>A</b>	<b>Přílohy</b>	<b>63</b>
A.1	Plán testu použitelnosti . . . . .	64
A.2	Zápis o testu použitelnosti . . . . .	65
A.3	Porovnání uživatelského rozhraní . . . . .	66



## Seznam obrázků

2.1	Zjednodušený zápis revize . . . . .	16
2.2	Životní cyklus Service Workers . . . . .	21
4.1	Porovnání desktopového a mobilního zobrazení . . . . .	31
4.2	Klikatelný prototyp aplikace - vytvoření revizní zprávy . . . . .	33
4.3	Porovnání návrhu před a po testu použitelnosti . . . . .	35
4.4	Vizuální identita aplikace . . . . .	36
4.5	Grafické uživatelské rozhraní na mobilním zařízení . . . . .	37
4.6	Porovnání stran revizní zprávy . . . . .	39
4.7	Porovnání zápisů o zkoušce . . . . .	40
5.1	Architektura Redux . . . . .	45
5.2	Schéma frontendové části . . . . .	47
5.3	Schéma backendové části . . . . .	48
6.1	Porovnání zobrazení menu . . . . .	56
6.2	Klient s více adresami . . . . .	58
6.3	Hodnocení Lighthouse . . . . .	59
A.1	Porovnání uživatelského rozhraní . . . . .	66

## Seznam tabulek

5.1	Popis jednotlivých kolekcí databáze . . . . .	54
6.1	Tabulka měření výkonu . . . . .	58

## Seznam zkratek

<b>DOM</b>	Document Object Model
<b>HTTP</b>	Hypertext Transfer Protocol
<b>HTTPS</b>	Hypertext Transfer Protocol Secure
<b>JSON</b>	JavaScript Object Notation
<b>PWA</b>	Progresivní Webová Aplikace
<b>UX</b>	User Experience
<b>UI</b>	User Interface
<b>URL</b>	Uniform Resource Locator

# 1 Úvod

Ačkoliv dnešní situace ve světě příliš zařízením využívající plyn nenahrává, jedná se stále o jeden z nejpoužívanějších typů zařízení pro vytápění. Dle českého statistického úřadu mají více než dvě třetiny bytů v České republice připojení na plyn. [1]

Plynová zařízení je nutné udržovat v provozuschopném a zejména bezpečném stavu, protože při opačném jednání může docházet k značným nárůstům nákladů na pozdějších opravách a nesprávně fungující plynového zařízení může být přímým ohrožením života a majetku osob. K tomu slouží revize, které jsou nezbytnou součástí každého provozu plynových zařízení a jsou ze zákona (Nařízení vlády č. 191/2022 Sb. [2]) povinná v případě výchozí revize a v případě provozní jsou povinná pouze pro organizace, nikoliv domácnosti. Revize plynových zařízení mají však svá pravidla, kterými se musí revizní technik řídit.

Revizemi se označuje celkové posouzení provozu schopnosti zařízení, při kterém se kontrolou, vyzkoušením a případně i měřením prošetřuje spolehlivost a zejména bezpečnost zařízení. Revize poté dělíme na výchozí a provozní, kterým se podrobně věnuje podkapitola 2.2.

Hlavním cílem této práce je vývoj webové aplikace pro revizní zprávy, jenž bude využívat prvků progresivních webových aplikací. Aplikace bude umožňovat záznam dat při provádění revizních kontrol a následné vytvoření revizní zprávy, která bude obsahovat všechny požadované náležitosti. Bude dále udržovat informace o současných a budoucích klientech, evidenci revizních zpráv a celkovou agendu potřebnou pro revizního technika. Zásadními požadavky na aplikaci jsou zjednodušení procesu vytváření revizních zpráv, celková úspora času a snížení spotřeby fyzických materiálů.

Práce se v první části věnuje analýze plynových zařízení a souhrnnému seznámení s problematikou vytváření revizních zpráv. Dále popisuje prvky a vlastnosti progresivních webových aplikací, včetně zhodnocení jejich použití.

V druhé části práce jsou definovány funkční a ostatní požadavky, které vychází z provedené analýzy. Následuje návrh řešení aplikace, který se zabývá návrhem User Experience a User Interface. Zde jsou popsány využití metodiky návrhu, doplněné o ilustrace. Součástí je i návrh nové podoby revizní zpráv - provozní, výchozí a zápis o tlakové zkoušce.

V další části se práce věnuje již implementaci navrženého řešení. Popisuje zvolené technologie pro vývoj a architekturu aplikace a seznamuje s výběrem funkčních prvků. Dále se zabývá implementací prvků PWA (instalací, notifikacemi a offline módem), datovým modelem aplikace a celkovým zabezpečením aplikace a jejich

dat.

Následující kapitola je zaměřena na testování. Testování probíhalo manuálně při průběhu vývoje a na konci vývoje v kooperaci s vybranými revizními techniky. Automatické testování se zaměřilo na prvky PWA, kde se testovala například doba načítání.

V poslední části práce jsou shrnuty výsledky z testování a celkové posouzení výsledné webové aplikace.

## 2 Analýza

Následující kapitola se nejprve věnuje analýze problematiky plynových zařízení a jejich revizí. Popisuje způsob získávání a zadávání dat při provádění revizí a tvorbě revizní zprávy. V další části pak analyzuje problematiku progresivních webových aplikací, kde jsou přiblíženy jednotlivé náležitosti pro vývoj PWA.

### 2.1 Plynová zařízení

Na základě stávající legislativy (Energetického zákona - 458/2000 Sb. [3]) jsou plynová zařízení definována jako všechna zařízení, která jsou připojena na plynárenská zařízení. Na začátku pomyslného seznamu plynových zařízení je hlavní uzávěr plynu a na konci konečné zařízení pro spotřebu plynu - kotel, bojler či varná deska. Plynárenskými zařízeními se rozumí zařízení výroby plynu, přepravní soustavy, distribuční soustavy, zásobníku plynu, těžebního plynovodu a přímého plynovodu. [3]

Pro plynová zařízení platí následující rozdělení do skupin, respektive podskupin:

#### (A) Výroba a úpravu plynů

- (A1) Zařízení pro výrobu a úpravu plyných paliv
- (A2) Zařízení pro výrobu a úpravu technických paliv

#### (B) Skladování plynů

#### (C) Plnění nádob plyny, tlakové stanice, zařízení pro plnění a čerpání nádrží vozidel plyny

- (C1) Zařízení pro plnění nádob plyny a tlakové stanice na plyná paliva
- (C2) Zařízení pro plnění nádob plyny a tlakové stanice na technické plyny

#### (D) Zkapalňování a odpařování plynů

#### (E) Zvyšování a snižování tlaku plynů

- (E1) Kompresní stanice plynu
- (E2) Regulační stanice plynu

## (F) Rozvod plynů

- (F1) Domovní rozvody na plynná paliva
- (F2) Průmyslové plynovody na plynná paliva kromě propanu, butanu a jejich směsí
- (F3) Nízkotlaké, středotlaké plynovody a přípojky pro veřejnou potřebu na zemní plyn
- (F4) Vysokotlaké plynovody a přípojky pro veřejnou potřebu na zemní plyn
- (F5) Nízkotlaké, středotlaké a vysokotlaké plynovody na propan, butan a jejich směsi
- (F6) Rozvody technických plynů

## (G) Spotřeba plynů spalováním včetně stabilních plynových motorů ve strojovnách nebo v kotelnách

- (G1) Spotřebiče s výkonem pod 50kW na plynná paliva
- (G2) Kotle s výkonem 50kW a více na plynná paliva
- (G3) Pece a průmyslová tepelná zařízení na plynná paliva

## (H) Vypouštění hasebních plynů

K provozu plynových zařízení se váží určité povinnosti. Provozovatel je povinen vést a uchovávat předepsanou průvodní technickou dokumentaci, provozní dokumentaci a evidenci plynového zařízení k zajištění bezpečnosti a ochrany zdraví při práci. Dále je povinen uchovávat zprávy o provedených kontrolách, servisu, údržbě a provozních revizích po dobu deseti let. Doklady o zkouškách provedených po celkové montáži a výchozí revize je povinen uchovávat po celou dobu provozu zařízení. Výše uvedené povinnosti plynou z § 14 Nařízení vlády č. 191/2022 Sb.

Povinnosti však vznikají i na straně revizního technika, respektive osoby, jenž je oprávněna k montáži, opravám, revizím nebo zkouškám plynového zařízení a to na základě § 9 Nařízení vlády č. 191/2022 Sb. Ten je povinen vést evidenci o provedených zkouškách a revizích, a to po dobu deseti let. Evidence musí obsahovat: evidenční číslo zprávy o revizi nebo zkoušce, datum vystavení dokumentů, označení skupiny a umístění posuzovaného plynového zařízení, kopie dokumentů souvisejících se zkouškou nebo revizí. [2]

## 2.2 Revize plynových zařízení

Všechna plynová zařízení lze provozovat až po provedení všech předepsaných revizí a zkoušek, které prokazují, že technický stav těchto zařízení je v souladu s právními a ostatními předpisy k zajištění bezpečnosti. Revize a zkoušky může provádět pouze osoba odborně způsobilá. Takovou osobou je osoba, která splňuje odbornou praxi a příslušné vzdělání a zároveň je držitelem osvědčení o odborné způsobilosti k montáži, opravám a revizím a absolvovala zaškolení v rozsahu stanoveném právními a ostatními předpisy. [2]

Od července tohoto roku (2022) dochází ke změně v legislativě a původní vyhláška č. 85/1978 Sb. o kontrolách, revizích a zkouškách plynových zařízení je zrušena na základě zákona č. 250/2021 Sb. o bezpečnosti práce v souvislosti s provozem vyhrazených technických zařízení a o změně souvisejících zákonů. Nově se tak revizní technik musí řídit nařízením č. 191/2022 Sb. o vyhrazených technických plynových zařízeních a požadavcích na zajištění jejich bezpečnosti. Tato skutečnost však nemění způsob provádění revizí ani zápis zpráv o ní. [4]

Momentální způsob získávání a zápisu dat z revizní kontroly se liší v závislosti na preferencích revizního technika. Na Obrázku 2.1 je možné vidět jednu z možností zápisu průběhu kontroly, která je následně převedena do formy revizní zprávy, které se práce věnuje níže.

Datum: 25.2.		Jméno: JENOLOVA ALELIA (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 <input checked="" type="checkbox"/> měď <input type="checkbox"/>	stoupačka: ocel <input checked="" type="checkbox"/> měď <input type="checkbox"/>
Z01	opravit protikorozní nátěr plynovodní instalace - drobná údržba		Z11	řádně ukotvit plynovodní potrubí u uzavěru před spotřebičem	
Z02	vodivé 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 vodivé		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 uzavěr před spotřebičem - ovládání, kontr., údr.	
Z10	řádně ukotvit plynovodní potrubí vedené v instal. šachtě		X	Z20	
Spotřebič: HORA 2489			uzavěr: <input checked="" type="checkbox"/>	umístění: <input checked="" type="checkbox"/>	
Z21	celý spotřebič vyčistit a seřídít		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 hořáku spotřebiče		Z28	vzhledem ke stavu a stáří spotřebiče doporučuji výměnu	
Zajistit:			X	Z29	
			Podpis:		

Obrázek 2.1: Zjednodušený zápis revize



## 2.2.1 Výchozí revize

Výchozí revize se provádí před samotným uvedením zařízení do provozu. Při tomto druhu revizí se prověřuje úplnost a správnost technické dokumentace zařízení, zda byly na zařízení provedeny povinné zkoušky a celkové posouzení provozuschopnosti.

### Úvodní strana

Na úvodní straně revizní zprávy se nachází jedinečný identifikátor revize, její evidenční číslo. To není nijak definované normou, je tedy čistě na revizním technikovi, v jakém formátu jej bude ukládat. Pro aplikaci to bude znamenat sjednocení formátu zápisu, tak aby se revizní zprávy lépe archivovaly.

Dále je možné si všimnout dvou různých dat, jedním se definuje den provedení revize a druhým pak, kdy byla revizní zpráva vypracována. Aplikace bude revizní zprávy vypracovávat automaticky, datum vypracování bude tedy shodný s dnem provedení revize.

Následně se zde nachází základní informace o zákazníkovi, respektive o objektu, ve kterém se revidované zařízení nachází. Společně s touto informací se zapisuje také zúčastněná osoba, která byla revizi přítomna. V mnohých případech je zákazníkem družstvo a poté je zúčastněnou osobou obyvatel bytové jednotky.

První část úvodní strany poté uzavírají údaje o revizním technikovi. Údaje, které jsou velmi důležité, neboť pokud se v nich přehledně jakákoliv chyba, může dojít k zneplatnění revizní zprávy. Je tím myšleno například neplatné evidenční číslo osvědčení technika, případně špatně zapsané evidenční číslo oprávnění.

V další části je zapsán druh revize, v tomto případě výchozí. Jak již bylo uvedeno výše, výchozí revize se provádí před samotným uvedením do provozu, zařízení nedisponuje závadami bránícími k bezpečnému používání a vypisují se případné nedostatky ohledně chybějících revizí elektro zásuvek a podobně. Dále se zapisuje označení revize, respektive předmět revize, kde je popsáno čím se revize zabývá. To je doplněno druhem zařízení a pomocí jakých norem je zařízení posuzováno, v tomto případě dle EN 1775, ČSN 386405, TPG G 704 01, G 800 03, G 934 01 a ostatních souvisejících předpisů.

Úvodní stranu zakončuje celkové posouzení stavu zařízení a zejména pak podpis a otisk razítka revizního technika. To právně stvrzuje bezpečné použití zařízení, v případě jeho odcizení, případně zneužití může revizního technika dostat do potíží.

### Další strany

Druhá a třetí strana se již věnuje technické stránce věci a je rozdělena do několika segmentů. Ty jsou označeny písmeny A - F, které označují jednotlivé úseky k prošetření.

Pod písmenem A se skrývá úsek, jenž nese technické hodnoty revidovaného zařízení. Popisuje se zde kompletní způsob zapojení zařízení, použité materiály, druhy závitů a umístění zmíněných komponent. Doplněné o informace kdo provedl montážní práce, včetně příslušných osvědčení zmíněných osob a firem. Součástí jsou

také údaje o revidovaném zařízení, jeho kompletní název, umístění, druh závitů a maximální spotřeba zásobníku plynu.

Segment B popisuje údaje o měření a zkouškách. Jsou v něm zapsány všechny zkoušky a měření, které byly na zařízení provedeny. Zapisuje se evidenční číslo zápisu o zkoušce, podrobněji v následujícím odstavci, a její výsledek.

Úsek označený písmenem C se věnuje zjištěnými závadami a nedostatky, případnými návrhy opatření a lhůtami k odstranění. U výchozích revizí zpravidla žádné závady nejsou, protože zařízení jsou zcela nové, nepoužité. Revizní technik může podotknout chybějící revizní zprávy k součástem zapojení, příkladem může být elektro zásuvka.

Segment D se též věnuje závadám, avšak těm z minulosti. Popisuje údaje o odstranění závad z předchozích revizí, případně kontrol. V tomto případě, kdy se jedná o revizi výchozí, logicky žádné předchozí závady nebudou existovat.

Segment E se zabývá záznamem o ostatních revizích provedených na zařízení. V této části jsou zapsány všechny doložené revizní zprávy, které se vztahují k zařízení. Jedná se například o zápis o tlakové zkoušce nebo zpráva o revizi spalinové cesty. Vše je doplněno o evidenční čísla revizí, datum jejich vykonání a samozřejmě o jméno revizního technika, který je vykonal.

Poslední písmeno F obsahuje segment doplňujících informací, může se jednat o soupis dalších dokumentů přiložených k revizní zprávě - protokoly o zkouškách, projektové dokumentace, osvědčení pracovníků a oprávnění montážní firmy nebo atesty použitých materiálů a systémů.

## **Zápis o zkoušce**

Nedílnou součástí výchozí revize plynových zařízení je také zápis o zkoušce, zejména pak zápis o pevnostní a tlakové zkoušce. Jedná se o samostatný dokument, jenž je připojen k revizní zprávě. Proto takový zápis má i své vlastní identifikační údaje. Obsahuje jak datum provedení zkoušky, tak i datum zápisu, jednoznačný identifikátor - evidenční číslo zkoušky, jméno a adresu zákazníka. Dále pak jméno technika, respektive název firmy, která provedla montážní práce, včetně čísla oprávnění. Základní informace uzavírá údaj o tom, kdo zkoušku provedl.

Následují technické hodnoty zařízení, kde jsou popsány materiály, jejich proporce a způsob zapojení. Poté se již provádí samotná zkouška, jejíž účelem je prokázat mechanickou pevnost a těsnost plynového zařízení před samotným uvedením do provozu. Jako první se provádí zkouška pevnosti, pokud se neprovádí obě zkoušky současně. Ta se provádí podle normy TPG 704 01, jenž vychází z pravidel ČSN EN 1775. Má za cíl prošetřit pevnost spojení u nově provedených instalací. K tomu je využit vzduch či inertní plyn (např. dusík), který je natlakován na zkušební tlak alespoň 1 bar a poté vpuštěn do ještě nevyčištěného vedení. Jakmile je dosaženo požadovaných hodnot tlaku, spustí se 10minutový aklimatizační cyklus. Poté již dochází na pevnostní zkoušku, kdy je plynovod vystaven zkušebnímu tlaku nejméně 15 minut a prošetřuje se, zda došlo k mechanickému poškození plynovodu či jeho částem a nedochází k úniku zkušebního média.

Pokud je zkouška pevnosti řádně ukončena je možné přejít ke zkoušce těsnosti.

Té musí být podrobena zařízení, jehož délka plynovodu přesáhne 3 m a může být zahájena jen pokud dojde k ustálení teploty zkušebního média. Jako zkušební médium lze použít vzduch nebo inertní plyn. Těsnící zkouška u plynovodů s přetlakem s 10 kPa se provádí zkušebním tlakem 1,5násobku provozního přetlaku, nejméně však 5 kPa. Plynovod je nejprve 15 minut převeden do aklimatizační fáze a poté již dochází ke zkoušce těsnosti a prošetřuje se, zda nedochází k poklesu zkušební tlaku.

Po provedení obou zkoušek je do zápisu zapsán výsledek obou zkoušek, včetně hodnot zkušební tlaku a délky zkoušky. Doplněny o název a výrobní číslo měřícího zařízení. [5]

## 2.2.2 Provozní revize

Provozní revize se provádí periodicky, nejméně jedenkrát za tři roky. Provádí se zejména po skončení zkušební doby, při úpravách na plynovém zařízení nebo při změně instalačního charakteru, které mohou mít vliv na bezpečnost a spolehlivost zařízení.

### Úvodní strana

Úvodní strana zprávy o provozní revizi se od výchozí jinak zásadně neliší. Obsahuje základní identifikační údaje o revizi - datum provedení, respektive vypracování a evidenční číslo. Následují informace o zákazníkovi, jeho jméno a adresa. Jméno revizního technika a jeho evidenční číslo oprávnění a osvědčení. Zásadním rozdílem oproti revizi výchozí je poté označení druhu revize a zařízení, kdy v rámci provozní revize je zařízení posuzováno dle norem EN 1775, TPG 704 01, G 934 01 a ostatních souvisejících předpisů. Úvodní stranu uzavírá údaj o zúčastněné osobě, celkové zhodnocení revize a zejména podpis a otisk razítka revizního technika, které celou revizní zprávu právně stvrzuje.

### Další strany

Další strany provozní revize se věnují technické zprávě. Obdobně jako u výchozí revize, je i zde rozdělena do jednotlivých segmentů označených písmeny A - F.

Segment A popisuje kompletní zapojení plynového zařízení, použité materiály a typy závitů. Následuje popis spotřebiče, jeho název, umístění a typ závitů.

Segment B se věnuje provedeným zkouškám a měřením. Narozdíl od výchozí revize zde probíhá více kontrol. Jmenovitě se jedná o kontrolu těsnosti přístupných rozebíratelných spojů, kontrolu funkce a úplnosti ovládacích a bezpečnostních armatur na zařízení, kontrolu spotřebičů (ovládací a bezpečnostní prvky) a kontrolu výskytu CO v místnosti. Vše doplněno o údaje příslušného měřícího přístroje - název a výrobní číslo. Výsledek těchto měření a zkoušek je zaznamenán v další segmentu pod písmenem C.

Segment C popisuje nalezené závady bránící a nebránící bezpečnému provozu. Ty jsou případně doplněny o nejzazší termín pro jejich odstranění.

V segmentu D revizní technik posuzuje odstranění nalezených závad z minulé revize. V návaznosti pak v segmentu E zapisuje, které revizní zprávy z minulosti, či

současnosti byly doloženy. Segment F je vhodný pro doplnění informací a doporučení týkající se provozní revize.

## 2.3 Progresivní webové aplikace

Progresivní webové aplikace, označované jako PWA, stojí pomyslně mezi nativními a webovými aplikacemi. Jedná se o webovou stránku, která disponuje určitými funkcemi, které byly dříve k vidění pouze u nativních aplikací. Není však pevně definováno, jakou sadu funkcionalit musí webová stránka obsahovat, aby mohla být označována za PWA. Nicméně mezi ty nejběžnější patří možnost instalace aplikace na plochu nebo její používání v offline režimu.

### 2.3.1 Pilíře

Progresivní webové aplikace je možné vnímat pomocí tří následujících pilířů.

#### Spolehlivost

Progresivní webová aplikace by měla fungovat v každém prohlížeči, i pokud nepodporuje PWA funkce. Zároveň by měla být responzivní a bezpečná.

#### Rychlost

PWA musí být rychlá, bez ohledu na stav sítě. Rychlost webu je zásadním faktorem. V případě, že se webová stránka načítá od 1 do 10 sekund, tak pravděpodobnost, že dojde k tzv. „user bounce“, se zvyšuje o 123 %. [6]

#### Přívětivost

PWA nabízí zážitek podobný nativním aplikacím, nezobrazuje ovládací prvky prohlížeče a navíc je možné využívat notifikace či další HW části zařízení. Důležitým aspektem je poté možnost instalace na plochu, která celý zážitek podtrhuje.

### 2.3.2 Manifest

Manifest webové aplikace je soubor typu JSON, který obsahuje informace, jak se má aplikace chovat v případě, že je nainstalována na počítač či mobilní zařízení.[7] Typicky manifest obsahuje data o názvu aplikace (*short\_name*, *name*), její ikony (*icons*), výchozí URL (*start\_url*), které má být otevřeno po spuštění aplikace a specifikaci zobrazení aplikace (*display*).

### 2.3.3 Service Workers

Service Workers našly řešení k dlouho nevyřešitelnému problému - jak správně ukládat data z webu do mezipaměti, aby byly přístupné i v offline režimu. Jedná se

v podstatě o specializovaný JavaScriptový soubor, který je možné označit jako virtuální proxy mezi prohlížečem a sítí. Jejich úkolem není pouze poskytovat přístup do aplikace v offline režimu, ale také i celkové zlepšení výkonnosti stávajících webů. To znamená, že pokud uživatel navštíví web podporující Service Workers v prohlížeči, který je nepodporuje, nedochází k žádným limitacím ve funkčnosti webu.

Na rozdíl od tradičního webového programování, Service Workers běží na samostatném vláknu od hlavního kódu aplikace a nemají tak přístup ke struktuře DOM. Jak už bylo uvedeno výše, Service Workers se nestarají pouze o offline režim, pomocí nichž je například umožněno využívat na webu notifikace, vypočítávat různé náročné operace na separátním vláknu. Jejich devízou je možnost převzít kontrolu nad síťovými požadavky, následně je upravovat nebo připravit vlastní odpovědi z mezipaměti.

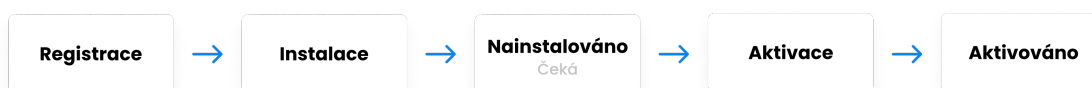
## Životní cyklus

Životní cyklus Service Workera je rozdělen do několika bodů (viz obr. 2.2), v rámci kterých je poté možné vykonávat určité akce. Ještě před samotnou registrací Service Workera je zapotřebí ověřit, zda ho prohlížeč podporuje. Nutno zmínit, že ten může reagovat pouze na akce pocházejících z definovaného *Scope*. Následně dochází k volání funkce `.register()` a pokud při tomto úkonu dojde k chybě, Service Worker je vyřazen.

V druhém bodě dochází k jeho instalaci, a to pomocí jeho první akce *install*, kterou získává pouze jednou za cyklus. V případě, že dojde ke změně, či úpravě Service Workera, prohlížeč na něj nahlíží jako na nového a provede instalaci znovu. V rámci této akce je možné uložit do mezipaměti všechna potřebná data a soubory pro offline režim.

Jakmile je Service Worker řádně nainstalován, schopen ovládat klienta a obsluhovat akce typu *push* a *sync*, je převeden do stavu, kdy čeká na aktivaci. Ta přichází pomocí volání akce *activate*. Ovšem pouze v případě, že aktuální Service Worker již neovládá žádného klienta. Pokud takový klient existuje, nový Service Worker zůstává ve fázi čekání. Díky tomu je možné ošetřit, že pouze jediná verze Service Workera je nainstalována. Existuje však způsob, jak převzít kontrolu nad všemi klienty a to pomocí volání funkce `client.claim()`.

Poslední fází životního cyklu Service Workera je jeho odstranění, ukončení. To se může stát skutečností v případě chyby při instalaci nebo aktivaci. Typicky ale dochází k jeho ukončení při instalaci nového Service Workera. [8]



Obrázek 2.2: Životní cyklus Service Workers

## 2.3.4 Cache

Nedílnou součástí Service Workera je rozhraní Cache, které umožňuje ukládání do mezipaměti, zcela oddělené od mezipaměti HTTP. Taková mezipaměť se nachází na klientské straně a není tedy nutné vysílat HTTP požadavky skrz síť. Je navíc rychlejší a dostupná i v offline režimu, což je jednou z klíčových vlastností PWA. Rozhraní Cache je programovatelné pomocí JavaScriptového kódu, na rozdíl od HTTP mezipaměti, která se řídí dle direktiv v HTTP hlavičce. To znamená, že je možné určit vlastní logiku, strategii ukládání požadavků do mezipaměti.

### Cache strategie

Strategií se rozumí přístup Service Workera k odpovědi na akci *Fetch*. Celkem existuje šest různých strategií, které jsou blíže popsány níže.

- **Pouze Cache**

Základní strategie, která říká, že pokud má Service Worker úplnou kontrolu nad stránkou, odpovídající požadavky budou chodit vždy pouze do mezipaměti. Všechna data musí být do mezipaměti uložena předem a nebudou nikdy aktualizována, aby taková strategie fungovala. Budou aktualizována pouze tehdy, kdy bude aktualizovaný i Service Worker.

- **Pouze síť**

Zcela opačná strategie od *Pouze Cache*, kde Service Worker předává požadavek síti, aniž by došlo k interakci požadavku s mezipamětí Service Workera. To je vhodné pro zajištění aktuálnosti obsahu, na druhou stranu však nutno zmínit, že taková strategie nikdy nebude fungovat s offline režimem.

- **Nejdříve Cache**

Při této strategii již dochází ke kooperaci mezi Cache a sítí. Nejprve je prohledán Cache a pokud obsahuje odpověď, poskytne ji. V opačném případě předá požadavek síti, který jej dokončí, přidá zpět do Cache a následně vrátí odpověď ze sítě.

- **Nejdříve síť**

Tato strategie nabízí opačný postup oproti *Nejdříve Cache*, kdy je nejprve požadavek zaslán do sítě a odpověď uložena do Cache. To poskytuje přístup k datům, pokud se uživatel nachází v offline režimu. Vhodné je to například pro HTML nebo API požadavky, které je potřeba mít nejaktuálnější v online režimu, ale také mít k dispozici i v offline.

- **Cache, poté síť**

Jedná se o složitější strategii, která kombinuje předchozí dvě strategie. Soustřeďuje se však rychlost přístupu ke zdroji, který zároveň udržuje aktuální. Funguje následovně: při prvním požadavku jej načte ze sítě, uloží do Cache a vrátí odpověď sítě. V případě následujících požadavků je využit Cache a na pozadí probíhá akce, kdy je požadavek vyžádán ze sítě a upraven v Cache.

- **Generický fallback**

Jedná se o rezervní strategii, kdy data nejsou dostupná z Cache ani ze sítě. Využívána jsou například u obrázků, kdy jsou v takovém případě nahrazeny šedým polem.[9]

## 2.3.5 Výhody PWA

### Rychlé načítání

Dnešní weby se dokáží načíst v milisekundách, často dokonce rychleji než nativní aplikace. Na rozdíl od klasického webu se díky Service Workeru PWA spustí okamžitě, bez ohledu na to, zda je uživatel právě připojen k 2G síti, či zcela bez připojení.

### Notifikace

Notifikace jsou jedním z nástrojů, jak znovu oslovit uživatele, připomenout jim návštěvu aplikace nebo doručit malé množství informací. Oznámení PWA mají nativní vzhled a jsou k nerozeznání od oznámení nativních aplikací. Notifikace nejsou ve skutečnosti jedna věc, ale skládají se ze dvou - *Push API* a *Notification API*.

*Push API* umožňuje uživatelům přihlásit se k odběru zpráv z aplikace a dovolit tak serveru kdykoliv zasílat prohlížeči uživatele zprávy. Tyto zprávy pak zpracovává Service Worker aplikace, který je může zpracovávat i ve chvíli, kdy uživatel aplikaci nepoužívá. Nejběžnější způsob zpracování je zobrazení oznámení, notifikace uživateli. S tím přichází poměrně velká dávka moci, kdy by bylo potencionálně možné zasílat uživateli nekonečné množství zpráv a tím jej obtěžovat. Aby se takovému zneužití předcházelo, Push zprávy nejprve prochází centrálním serverem pro zasílání zpráv. Tento centrální server je zajištěn dodavatelem prohlížeče a dohlíží na všechny odběry zpráv, které uživatel učinil. Má na starost, aby Push zprávy nebyly zneužity a uživatelé nebyli spamováni nevyžádanými zprávami. Jeho prací je také zajistit, že uživatel obdrží zprávu od aplikace i v případě, že v době odeslání nebyl k dispozici.

*Notification API* nabízí aplikaci nebo Service Workerovi vytvářet a spravovat zobrazení systémových oznámení na zařízení. Tyto oznámení jsou zobrazeny mimo uživatelské prostředí webového prohlížeče a jsou tedy mimo kompetenci okna či karty prohlížeče. Samozřejmě, aby bylo předcházeno tomu, že uživateli nebude každá navštívená stránka zasílat oznámení, musí se stránka, či aplikace nejprve zeptat, zda taková oznámení může posílat.

Kooperace výše dvou popsanych API vypadá následovně:

1. Aplikace si vyžádá povolení na zasílání notifikací, uživatel jej povolí
2. Aplikace kontaktuje centrální server a požádá o vytvoření nového odběru zpráv
3. Centrální server vrací detaily o odběru
4. Aplikace zasílá detaily o odběru na svůj server a server si jej ukládá
5. Aplikace využívá získaných detailů a posílá oznámení do centrálního serveru

6. Centrální server přeposílá oznámení do prohlížeče uživatele
7. Service Worker aplikace získává oznámení
8. Service Worker zobrazuje oznámení uživateli

Důležité uvést, že každý prohlížeč má nastavena odlišná pravidla chování k notifikacím. Například prohlížeč Mozilla Firefox blokuje všechna oznámení z webových stránek, která je zobrazují, ještě než požádají o povolení. Aktuálně notifikace nejsou dostupné pro systémy iOS od společnosti Apple, to by se však mělo změnit s nastupující verzí operačního systému, která má označení 16 a předpokládá se, že by měly být oznámení k dispozici v roce 2023. [10]

### **Možnost přidání na domovskou stránku**

Jednou z podstatných funkcí PWA je možnost jejího přidání na domovskou stránku mobilního zařízení či počítače. Tedy její ikonku, která pak umožňuje pohlížet na PWA jako na nativní aplikaci. Navíc to pro uživatele představuje jednodušší přístup do aplikace, než kdyby musel zadávat URL adresu do prohlížeče. K tomu aby bylo možné aplikaci přidat na domovskou stránku, je nutné zajistit následující:

- **Bezpečný původ**  
Aplikace by měla být dostupná v rámci HTTPS, aby bylo zajištěno, že obsah, který je předmětem požadavku nebyl mezi serverem a prohlížečem upraven.
- **Service Worker**  
Service Worker by měl být zaregistrovaný v rámci *start\_url*, který je definovaný v Manifestu.
- **Manifest**  
Manifest musí obsahovat základní informace, jako: *name*, *start\_url*, *icons* a *display*

Pokud jsou tyto požadavky splněny a prohlížeč podporuje Service Workera, je možné aplikaci nainstalovat na domovskou stránku zařízení.

Samotná instalace aplikace může být provedena dvěma způsoby. První možností je manuální instalace, kdy uživatel musí nalézt tlačítko k instalaci, které se nachází na obecně nedefinovaném místě. Každý prohlížeč k této problematice přistupuje po svém. Dalším ztížením může být fakt, že uživatel není seznámen s tím, že je webová aplikace progresivní a tudíž neví o možnosti instalace.

Druhou možností je o instalaci informovat uživatele předem. Prohlížeče nabízejí možnost, v případě detekce PWA, zobrazení informačního banneru s tlačítkem pro instalaci. Může být definováno, kdy k zobrazení dojde. Není nutné možnost instalace zobrazovat uživateli, který v aplikaci neprovádí žádné interakce. Banner pak může uživatele rušit. Naopak je vhodné instalaci nabídnout, pokud uživatel již několik interakcí provedl a může tak ocenit výhody instalace.



## **Nativní vzhled**

PWA otevřené přímo z domovské stránky mohou nabídnout vzhled nativní aplikace, např. včetně tzv. splash-screen. Mohou být spuštěny naprosto bez elementů uživatelského prostředí jak webového prohlížeče, tak i samotného zařízení. Případně je možné i zamknout orientaci displeje, to je využíváno zejména u her. [9]

### **2.3.6 Nevýhody PWA**

I přestože se PWA jeví jako vhodný nástroj pro vývoj mobilních aplikací, jsou i zde určitá omezení, která je nutné zohlednit.

#### **Distribuce**

PWA nejsou dostupné na běžných platformách pro stahování aplikací - Google Play a App Store. Uživatel musí aplikaci nejprve navštívit v rámci prohlížeče svého zařízení.

#### **Omezený přístup**

Na rozdíl od nativních aplikací, PWA mají omezený přístup k některým prostředkům zařízení. Nemohou využívat například kontakty, NFC, kalendář nebo Bluetooth.

#### **Stále ve vývoji**

Kompabilita PWA s (mobilními) prohlížeči a operačními systémy je stále ve vývoji. Zejména pak zařízení Apple byla do nedávna velmi limitována, nebylo možné využívat Face ID, Touch ID a nebo přijímat oznámení. Posledního jmenovaného ještě stále není možné využívat, nicméně jak již bylo uvedeno, v příštím roce by se toto mělo změnit.

#### **Výkon**

V návaznosti na vývoj je vhodné uvést výkonnost PWA, která dosahuje dobrých výsledků, ale ještě stále se nepřibližuje výkonnosti nativních aplikací.

## 3 Požadavky na aplikaci

V této kapitole se práce věnuje popisu požadavků na aplikaci, které vychází z provedené analýzy. Požadavky jsou rozděleny na funkční a ostatní.

### 3.1 Funkční požadavky

Funkční požadavky definují funkce, které má aplikace obsahovat v produkční podobě.

#### (1) Registrace / Přihlášení uživatele

- (1.1) Registrace uživatele pomocí emailové adresy, jména a příjmení, či účtů Apple nebo Google
- (1.2) Přihlášení uživatele pomocí emailové adresy a hesla, případně pomocí účtů Apple nebo Google
- (1.3) Možnost obnovy hesla

#### (2) Registrace a editace osobních údajů uživatele a měřících přístrojů

- (2.1) Uložení a editace osobních údajů (dosažené vzdělání, čísla osvědčení a oprávnění, razítko revizního technika, místo působení)
- (2.2) Registrace a editace měřících přístrojů (název přístroje, výrobní číslo)

#### (3) Tvorba revizních zpráv

- (3.1) Volba druhu revize - provozní a výchozí, případně načtení ze vzoru
- (3.2) Přidání osobních údajů o klientovi (jméno, adresa, telefonní číslo, email), případně jméno správce objektu a umístění bytové jednotky
- (3.3) Popis technických hodnot zařízení (evidenční číslo revize, název a umístění zařízení, typ závitu, způsob zapojení)
- (3.4) Informace o nejčastějších závadách na zařízení
- (3.5) Záznam provedených měření (kontrola těsnosti rozebíratelných spojů, kontrola funkce ovládacích a bezpečnostních armatur, kontrola spotřebičů, kontrola výskytu oxidu uhelnatého), včetně použitých měřících přístrojů a jejich údajů

- **(3.6)** V případě výchozí revize vytvoření záznamu o tlakové zkoušce (evidenční číslo, údaje o montáži, popis technických hodnot, hodnoty z provedených zkoušek a jejich výsledek, celkové posouzení) a záznam o maximální spotřebě zemního plynu
- **(3.7)** Popis nalezených závad na spotřebiči a zapojení celého systému, případně uložení lhůt k jejich odstranění
- **(3.8)** Záznam o odstranění závad z předchozích revizí
- **(3.9)** Záznam o ostatních revizích provedených na příslušném zařízení
- **(3.10)** Přidání doplňujícího popisu a případných doporučení
- **(3.11)** Uložení doporučeného termínu příští revizní kontroly na zařízení
- **(3.12)** Uložení celkového posouzení revizní kontroly
- **(3.13)** Přidání elektronického podpisu
- **(3.14)** Export revizní zprávy do formátu PDF
- **(3.15)** Uložení revizní zprávy, případně uložení jako vzor pro budoucí použití
- **(3.16)** Odeslání klientovi jako příloha v emailu

#### (4) Karta klienta

- **(4.1)** Vytvoření karty klienta na základě vytvořené revizní zprávy, obsahující základní údaje a adresu klienta a příslušné soubory
- **(4.2)** Správa karty klienta (editace, odstranění)
- **(4.3)** Správa souborů (úprava, uložení, tisk, odeslání jako příloha emailu, odstranění, použití jako vzor)
- **(4.4)** Využití mapových pluginů k zobrazení adresy na mapě

#### (5) Seznam klientů

- **(5.1)** Možnost vyhledávání v seznamu
- **(5.2)** Evidence provedených revizí a zkoušek (evidenční číslo, typ, datum provedení, jméno klienta)

#### (6) Kalendář

- **(6.1)** Přidání doporučených termínů příštích revizí
- **(6.2)** Vytvoření termínů schůzek (jméno klienta, datum, čas, místo)

#### (7) Notifikace

- **(7.1)** Blížící se expirace revizních zpráv

#### (8) Statistiky

- **(8.1)** Celkový počet revizních zpráv
- **(8.2)** Měsíčný vývoj počtu provedených revizních kontrol

## 3.2 Ostatní požadavky

Ostatní požadavky definují vlastnosti a podmínky fungování aplikace.

### (A) PWA

- Vývoj aplikace s využitím principů PWA, možnost instalace (přidání na plochu) na zařízení

### (B) Responzivní zobrazení

- Uživatelské prostředí přizpůsobené použití aplikace na počítači, mobilním zařízení či tabletu

## 4 Návrh řešení

Následující kapitola se zabývá návrhem aplikace v rámci User Experience a User Interface. Dále popisuje návrh podoby revizní zprávy, která je výstupem aplikace.

### 4.1 User Experience a User Interface

Návrh User Experience (UX) a User Interface (UI) je pro PWA specifickým. Je tomu tak zejména kvůli responzivnímu zobrazení a zohlednění využití PWA funkcí. Nicméně základní metody výzkumu UX se neliší.

#### 4.1.1 Kontextuální rozhovor

Kontextuální rozhovor slouží k pochopení činnosti uživatele a zjištění jeho potřeb. Při provádění kontextuálního rozhovoru je potenciální uživatel sledován v jeho přirozeném prostředí a nejsou po něm požadovány žádné úkoly. Vhodným doplněním je pak kladení otázek, které mohou přispět k úplnému porozumění problematice a získání potřebných dat pro další výzkum. [11]

V rámci této práce byl kontextuální rozhovor vykonán ve spolupráci s reprezentantem z řad revizních techniků, který působí ve městě Trutnov. S reprezentantem bylo možné se zúčastnit pracovního dne a seznámit se s metodami revizní kontroly a tvorby revizní zprávy.

Výstupem tohoto rozhovoru pak bylo zjištěno následující:

- V případě provádění revizí v panelovém domě, se značná část získaných dat opakuje
- Tlaková zkouška, jenž je součástí výchozí revize, se typicky provádí ve stejnou dobu jako revize
- Revizní technik zaměstnává kolegu, kterému zasílá oskenované protokoly z revizní kontroly (viz Obrázek 2.1), které jsou následně ručně převedeny do elektronické podoby
- Po vytisknutí a podepsání revizní zprávy je dopravena klientovi, případně si ji klient sám vyzvedne

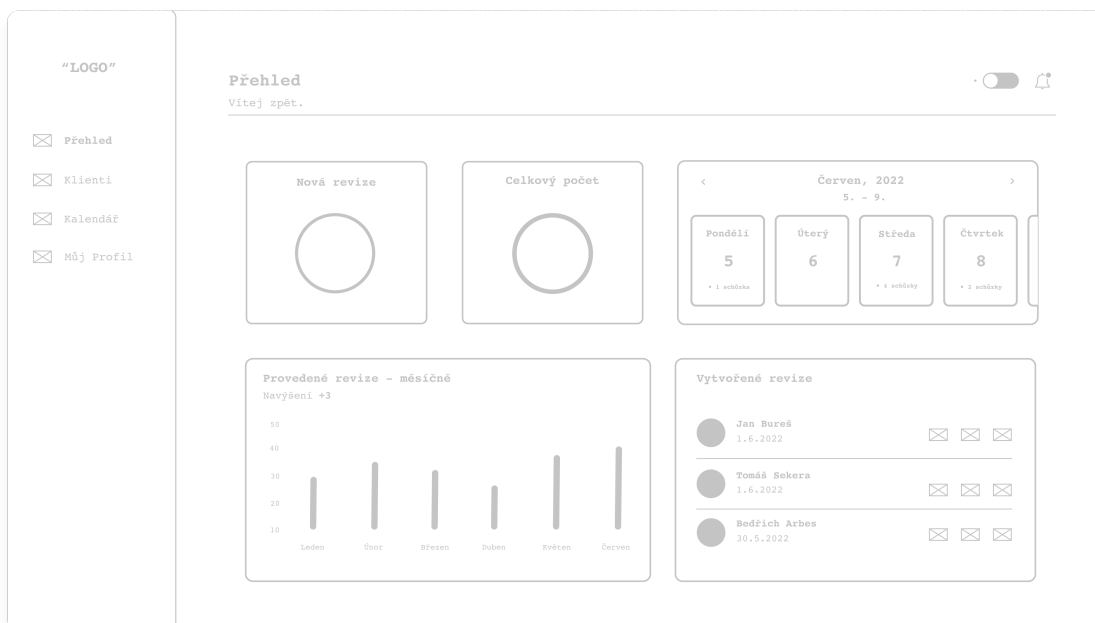
## 4.1.2 Prototyping

Prototyping je způsob, jak umožnit uživateli interakci s aplikací a to bez potřeby vývoje aplikace. Dochází tak k úspoře nejen času, ale finančních prostředků. To protože výsledné prototypy nemusí být vždy finálními a může docházet ke změnám v rozložení aplikace. Prototyp může být tzv. low-fidelity (papírové kresby) či high-fidelity (klikatelný prototyp). [12]

K návrhu UX a vytvoření prototypu byl využit software Figma, který je shodou okolností také PWA. Tento software nabízí možnost využití i svých nativních aplikací pro mobilní zařízení, kde je možné jednak pracovat s projekty, ale zároveň využít zařízení pro vizualizaci. Usnadňuje tak proces návrhu, protože designér má možnost vidět reálné zobrazení na požadovaném zařízení.

Jak již bylo uvedeno, návrh PWA je specifickým z důvodu responzivního zobrazení. Bylo tedy nutné zvolit, zda se nejprve bude navrhovat desktopové zobrazení a následně z něj bude odvozeno mobilní, či naopak. K návrhu byl využit high-fidelity model, z důvodu lepší možnosti vizualizace na zařízeních a podrobného testování. Předně bylo navrženo zobrazení pro desktop, konkrétně obrazovky pro přehled aplikace, která na pravé straně obsahuje boční panel s navigací po aplikaci, na opačné straně pak nabízí možnost přepínání mezi klasickým zobrazením a dark módem a oznamuje uživatele o nových notifikacích. V hlavní části se nachází tlačítko pro volbu vytvoření revizní zprávy, zjednodušené zobrazení kalendáře, jenž zachycuje pouze aktuální týden a informuje uživatele o naplánovaných schůzkách na jednotlivé dny. Dále obsahuje naposledy vytvořené revize, k nimž jsou připojeny rychlé volby pro manipulaci s nimi (zobrazení, úprava, odstranění). Přehledovou obrazovku uzavírají statistická okénka. První se věnuje celkovému počtu provedených revize, druhé pak vývoji provedených revizí v měsíčním horizontu.

Na Obrázku 4.1 je vyobrazeno porovnání přehledové obrazovky pro desktopové a mobilní zobrazení. Je možné si všimnout zejména přesunutí bočního panelu s navigací do spodní části, do tzv. Bottom Navigation Menu, které je typické pro nativní aplikace mobilních zařízení. Dále jsou jednotlivá okna seřazena za sebe do zúženého zobrazení a přechod mezi nimi je řešen pomocí scrollingu.



(a) Desktopové zobrazení



(b) Mobilní zobrazení

Obrázek 4.1: Porovnání desktopového a mobilního zobrazení

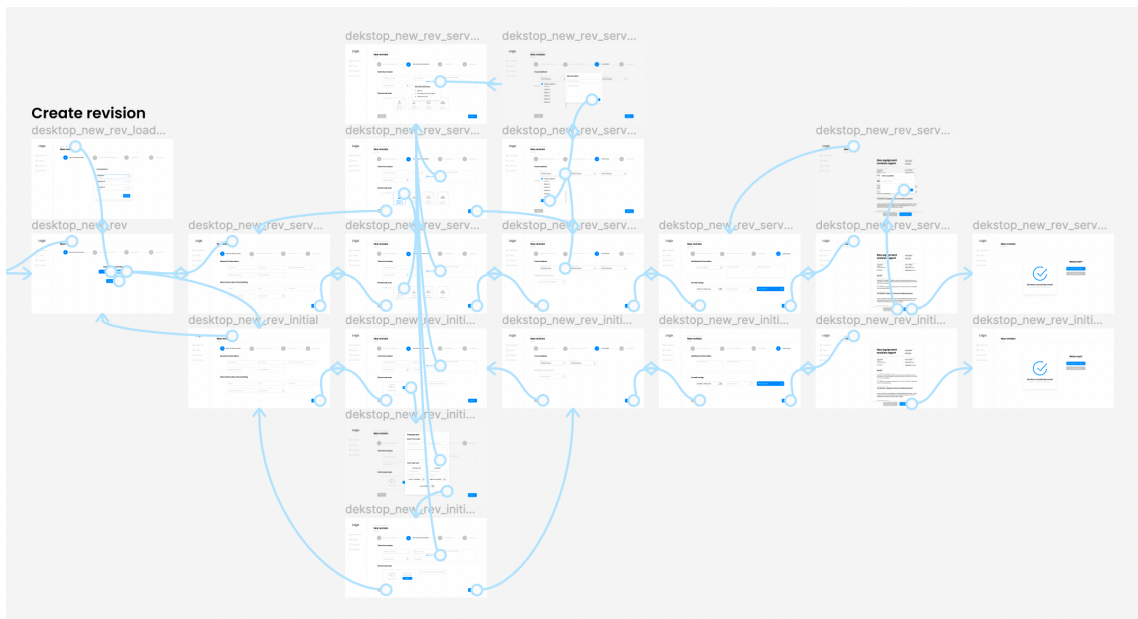
Dále byly navrhovány obrazovky pro vytvoření revizní zprávy, uživatele, klientů a kalendáře. Simultánně i pro mobilní zařízení. Po dokončení návrhu obrazovek bylo možné přejít k implementaci klikatelného prototypu.

Proces vzniku klikatelného prototypu je rozdělen do několika bodů:

- **Nastavení zařízení, na kterém bude prototyp prezentován**  
Je možné volit z již přednastavených, kde se nachází nejběžnější zařízení typu - iPhone, iPad, Apple Watch, Macbook a samozřejmě alternativy pro Android zařízení. Případně je možné definovat vlastní rozměr zobrazení.
- **Vytvoření vazby mezi obrazovkami**  
Propojení obrazovek se provádí pomocí nasměrování šipkami. Má své vlastnosti, které je možné upravovat. Je zvolen komponent obrazovky, který po stisku, přetažení či pohybu myši po něm, provádí přesun uživatele na následující obrazovku. Definuje se způsob přechodu na další obrazovku, je možné volit instantní přechod, přesunutí nebo například „smart animate“.
- **User Flows**  
User Flows slouží k modelování uživatelských scénářů. To je vhodné, pokud je požadováno rozdělení prototypu na několik dílčích scénářů. Například je možné vytvořit cestu pouze pro vytváření revizní zprávy a zamezit tak, aby tester měl možnost přístupu i do dalších částí prototypu. Flows se vytváří definováním počátečního bodu a následným propojením obrazovek.
- **Další možnosti**  
Dalšími možnostmi pro autentické zobrazení aplikace je možnost scrollování pro jednotlivé komponenty či celé obrazovky, a to horizontální nebo vertikální.

Pro ilustraci výstupu prototypingu je na Obrázku 4.2 vyobrazena část aplikace, která se soustředí na vytvoření revizní zprávy.





Obrázek 4.2: Klikatelný prototyp aplikace - vytvoření revizní zprávy

### 4.1.3 Testování použitelnosti

Posledním nutným bodem před samotnou implementací aplikace je její otestování v rámci použitelnosti. Typicky dostávají potencionální uživatelé úkoly, které musí v aplikaci splnit a designér je pozoruje a dělá si poznámky. Cílem je odhalit případné chyby v návrhu, či nesrovnalosti, které způsobují neintuitivnost a shromáždit data o spokojenosti uživatele.

Aby bylo docíleno úspěšného návrhu, je obecně doporučováno dodržovat základní hierarchii testování použitelnosti. Vstupní položkou testování je sestavení jeho plánu, jehož účelem je zdokumentovat, co bude vykonáno, jak to bude vykonáno, kým to bude vykonáno, jaké metriky budou zachyceny a jaké scénáře budou využity. Dále již dochází k procesu testování, který probíhá za definovaných podmínek.

Výstupem testování je analýza a sepsání reportu o výsledcích testování. V závislosti na určených metrikách jsou shromážděny získaná data, zejména pak kvantitativní data, jako míra chybovosti nebo úspěšnost splnění daného úkolu. Součástí jsou také kvalitativní data, která popisují jakými cestami se respondenti vydali, jakým problémům čelili nebo jak dané úkoly okomentovali.

Poté je provedena analýza, k níž je nápomocný výše uvedený plán testování, aby bylo i zpětně jednoznačně určeno, co bylo testováno a na jaké otázky bylo odpovídáno.

V příloze A.1 je přiložen plán testu použitelnosti této práce, kde jsou popsány základní informace o aplikaci - její pracovní název a zejména pak záměr testování, tedy ověření, že je aplikace intuitivní a že vyhovuje potřebám revizního technika. Sekci zakončuje informace o rozsahu testování, kdy byl určen kompletní návrh aplikace. Dále je popsán způsob testování, kdy byl určeno místo a čas konání. Z důvodu pracovní vytíženosti revizního technika bylo možné testování provést pouze ve večer-

ních hodinách a z důvodu vzdálenosti mezi autorem práce a revizním technikem bylo testování provedeno online. Software Figma nabízí sdílení odkazu na projekt, který je u příjemce plně funkční. Byla definována potřebná vybavení, tak aby respondent byl na testování dostatečně připraven, a orientační délka celkového testování.

Následují informace o respondentovi a naplánované scénáře. Celkem bylo naplánované pět scénářů, kdy se konkrétně jednalo o úkoly na vytvoření výchozí revize, vytvoření provozní revize, otevření karty klienta, stažení revizní zprávy a přidání schůzky s klientem.

Plán pro test použitelnosti uzavírají zvolené metriky - subjektivní a kvantitativní. Subjektivními metrikami se zjišťovalo celkový vjem aplikace na respondenta u konkrétních scénářů. Pro první scénář, kdy měl respondent za úkol vytvořit provozní revizi, byl dotazován jak hodnotí celkový průběh revize, jak hodnotí definování závad u spotřebičů a zda by něco na provádění revize změnil. Pro druhý scénář, kdy měl respondent za úkol vytvořit výchozí revizi, byl dotazován, jak hodnotí zápis tlakové zkoušky. U třetího scénáře, jenž se zabýval manipulací s kartou klienta, bylo dotazováno, zda je stránka přehledná a zda nechybí nějaké rychlé volby u dokumentů. Kvantitativními metrikami byly zvoleny základní metriky, tedy míra chybovosti při provádění úkolů a úspěšnost splnění úkolů.

V příloze A.2 této práce je přiložen zápis (report) o testu použitelnosti, kde je zaznamenán průběh testování - jednotlivé scénáře, jejich splnění, případné potíže a poznámky. Respondent splnil všechny definované úkoly a neměl s nimi žádné potíže. Pouze u tvorby revizní zprávy se pozastavil nad umístěním pole pro vyplnění evidenčního čísla revize. Spíše by uvítal umístění pole v popředí technických hodnot. Dále se report věnuje připraveným otázkám a odpovědím na ně respondentem. Respondent odpověděl na otázky spíše kladně s přihlédnutím na výše uvedenou výtku. Report uzavírá celkové hodnocení a doporučení, kde bylo shledáno, že testování bylo úspěšné a bylo doporučeno upravit návrh tvorby revizní zprávy, konkrétně umístění pro pole evidenčního čísla.

Na základě provedeného testování o použitelnosti, respektive na základě jeho výsledků, bylo nutné provést změny v návrhu aplikace. Na Obrázku 4.3 je vyobrazeno porovnání původního návrhu zadávání dat pro technické hodnoty zařízení a nově upraveného tak, aby korespondoval s požadavky z reportu testování.

#### Technické hodnoty

Způsob zapojení	Evidenční číslo	Název spotřebiče
	Umístění spotřebiče	<input checked="" type="checkbox"/> Nejčastější závady spotřebiče
		Typ závitů

(a) Návrh před úpravou

#### Technické hodnoty

Evidenční číslo	Název spotřebiče	Způsob zapojení
	<input checked="" type="checkbox"/> Nejčastější závady spotřebiče	
Umístění spotřebiče	Typ závitů	

(b) Návrh po úpravě

Obrázek 4.3: Porovnání návrhu před a po testu použitelnosti

### 4.1.4 Vizuální identita

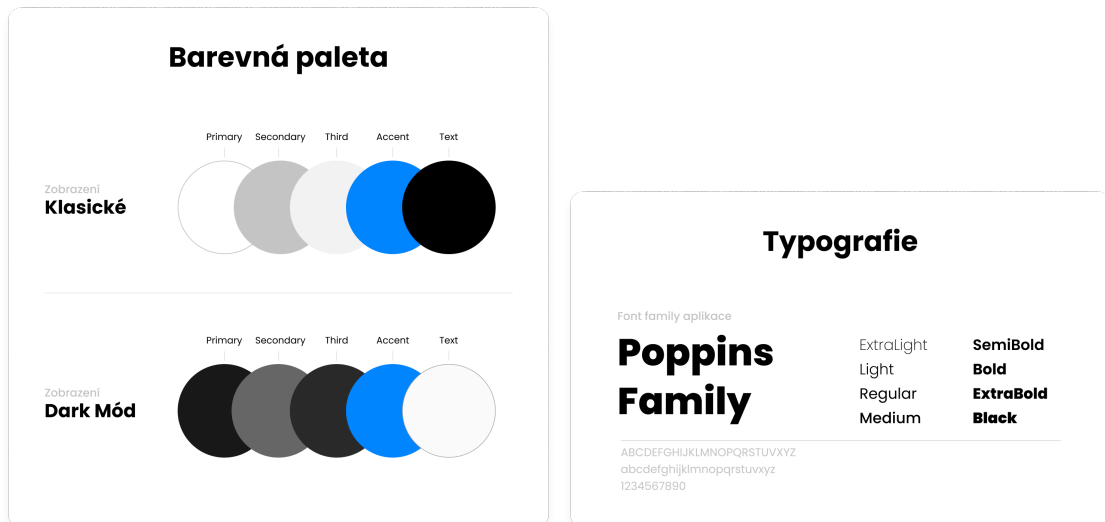
Po dokončeném návrhu UX aplikace bylo možné přejít k estetické stránce aplikace. Bylo nutné nejprve definovat barevnou paletu aplikace, typografii a samozřejmě název a logo aplikace. Z již provedené analýzy a získaných informací z UX testování bylo vyzkoušeno, že na aplikaci je kladen spíše funkční důraz a není tedy nutné vymýšlet závrtné designové prvky. Ty by mohly být mnohdy na škodu a potenciálně uživatele mást. Cílem bylo vydat se směrem minimalistického designu, který bude působit moderně a čistě. A zejména pak bude předávat pozornost na důležité prvky aplikace, jimiž jsou vstupy a výstupy aplikace. Z toho důvodu bylo nutné definovat barevnou paletu aplikace i s ohledem na tuto skutečnost.

Barevná paleta, jenž je vyobrazena na Obrázku 4.4a, v sobě ve skutečnosti ukrývá celkem dvě různé palety. Aplikace totiž nabízí i tzv. „Dark mód“, který je v dnešní době hojně používán a tedy i vyžadován. Každý uživatel si tak může, dle svých preferencí, zvolit způsob zobrazení aplikace. Klasické zobrazení nabízí základní bílou barvu, doplněnou o stupně šedi. Jako akcentní barva byla zvolena světle modrá, tak aby působila kontrastně a upoutala pozornost uživatele - je využívána zejména pro tlačítka a interaktivní elementy aplikace. Dark mód zobrazení má jako základní barvu černou, nikoliv však černou s označením #000000, ale světlejší, tak aby mohly být využívány prvky stínování. Doplněna je o stupně šedi, narozdíl od klasického zobrazení jsou tyto výrazně tmavší. Akcentní barva je totožná s klasickým zobrazením. V dark módu není obecně doporučováno pro text volit bílou barvu (#FFFFFF), protože působí až příliš kontrastně a způsobuje uživatelům potíže při čtení. Z toho důvodu byla zvolena tmavší odstín bílé, konkrétně #FAFA.

Aplikace využívá rodinu písma Poppins, které působí moderně a doplňuje minimalistický styl designu (viz Obrázek 4.4b). Název aplikace vychází z celkové myšlenky a přínosu aplikace, která má pomoci uživatelům s revizními zprávami. Vymyšlen

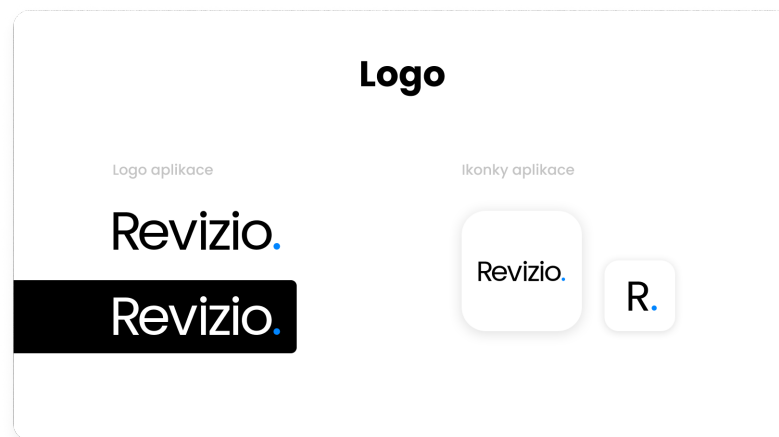
byl název „Revizio“, kde první část názvu „Reviz“ označuje revizi, druhá část „io“ poté označuje zkratku pro vstupy a výstupy, tedy inputs/outputs. Kdy vstupem jsou myšleny získaná data z revizní kontroly a výstupem je revizní zpráva.

Logo aplikace se zaměřuje na název aplikace, jedná se tedy spíše o logotyp, protože neobsahuje žádný piktogram. Vychází ze zvoleného písma Poppins, kde poté byly upraveny jednotlivá písmena, rozestupy apod. Dále je ještě doplněn o tečku v akcentní barvě. Pro tmavý podklad došlo pouze k inverzi primární barvy. Ikonka aplikace obsahuje kompletní logo (logotyp), které je umístěno na bílém podkladu. Logo a ikonka jsou ilustrovány na Obrázku 4.4c



(a) Barevná paleta aplikace

(b) Typografie aplikace



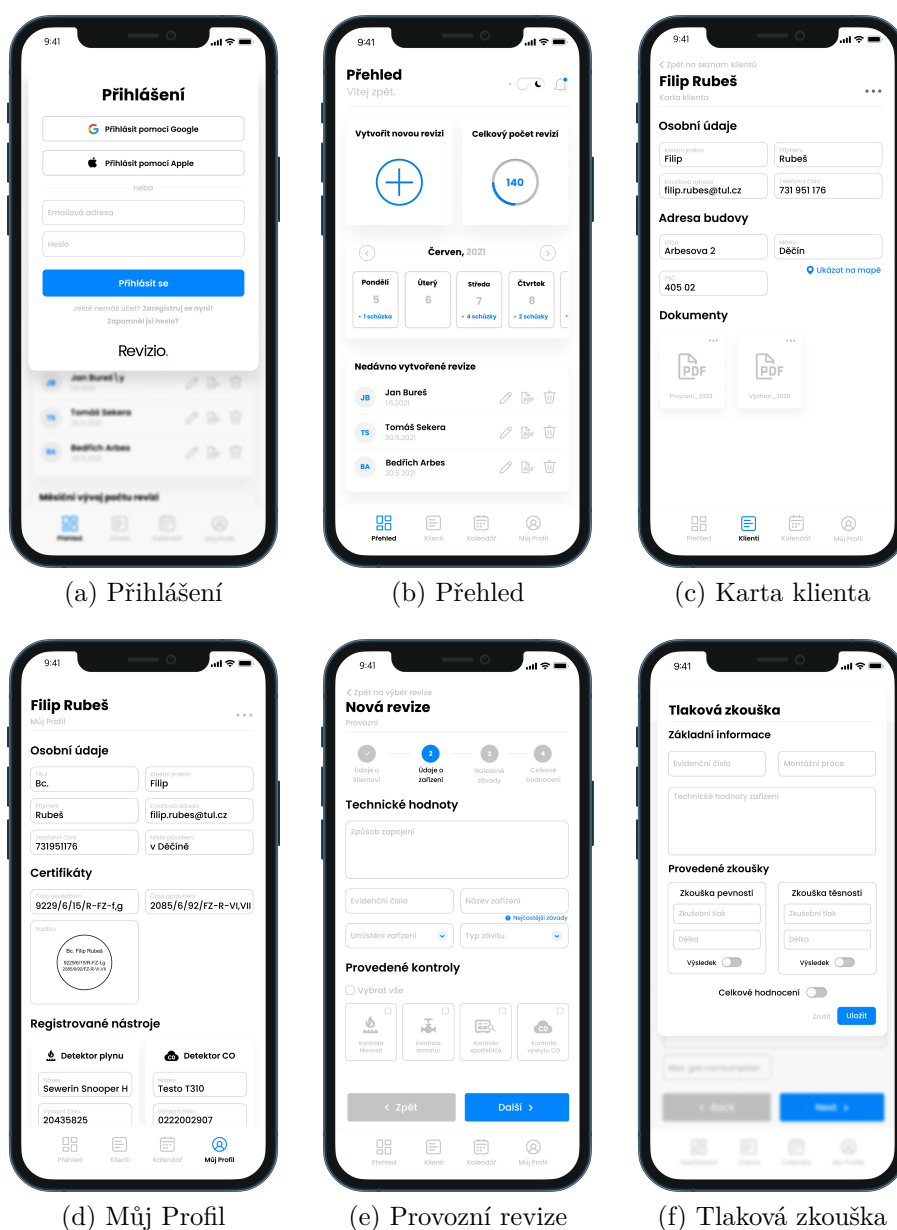
(c) Logo aplikace

Obrázek 4.4: Vizuální identita aplikace

## 4.1.5 Grafické uživatelské rozhraní

Na základě výše definovaných prvků vizuální identity bylo přistoupeno k návrhu grafického rozhraní. Jde v podstatě o přidání barev, stylů, efektů do UX návrhu. Na Obrázku A.1, který je v příloze této práce, je ilustrován rozdíl mezi návrhem pouze UX a finálním návrhem aplikace, jež nese prvky UI designu.

Proces vzniku uživatelského rozhraní nebyl v řádech dnů, či týdnů. Jednotlivé elementy UI se upravovaly během celého vývoje aplikace a finální UI design je vyobrazen na následujících obrázcích (Obr. 4.5), které zobrazují výběr obrazovek aplikace pro mobilní zařízení.



Obrázek 4.5: Grafické uživatelské rozhraní na mobilním zařízení

## 4.2 Revizní zpráva

Velmi důležitým faktorem celé aplikace je revizní zpráva. Po analýze aktuálního formátu zprávy (viz kapitola 2.2) bylo usouzeno, že by bylo vhodné celou zprávu rozložit a poskládat znovu, moderněji, přehledněji. Samozřejmě by bylo možné vzít aktuální a pouze měnit, respektive doplňovat získaná data. Přesto však bylo rozhodnuto pro zcela nový vzor revizní zprávy.

### 4.2.1 Provozní revize

Zpráva o provozní revizi plynového zařízení byla vytvořena tak, aby korespondovala s vizuální identitou aplikace. Obsahuje rodinu písma Poppins a využívá definované palety barev.

#### Úvodní strana

Úvodní strana zprávy obsahuje nejprve evidenční číslo revize a datum jejího provedení. Dále jméno klienta, jeho adresu a jméno osoby, jenž se zúčastnila revizní kontroly. V případě, že objekt má ještě svého správce, je tato skutečnost doplněna. Následují informace o revizním technikovi - jméno, číslo osvědčení a číslo oprávnění. V dalších částech strany je zaznamenán druh revize, označení zařízení, které je předmětem revize a druh zařízení, kde je specifikováno, dle jakých norem bylo zařízení posuzováno. Ve spodní části strany se nachází celkové posouzení revize, doplněno o rozdělovník a otisk razítka s podpisem (elektronickým), kterými je stvrzena právní účinnost. Na Obrázku 4.6a, respektive 4.6d je vyobrazeno porovnání původního a zaktualizovaného vzhledu úvodní strany revizní zprávy.

#### Druhá strana

Druhá strana se věnuje technické zprávě, kterou otevírá seznam segmentů. Následuje segment A s popisem technických hodnot zařízení, názvem, umístěním a typem závitu spotřebiče. Dále je popisován segment B, který zaznamenává provedené kontroly a měření a jakými nástroji bylo měřeno, včetně výrobních čísel nástrojů. Segment C zaznamenává zjištěné závady na zařízení a spotřebičích a případně oznamuje o lhůtě k odstranění závad. Porovnání původní verze druhé strany je vyobrazena na Obrázku 4.6b a nová verze na Obrázku 4.6e.

#### Třetí strana

Třetí strana obsahuje dodatečné informace k revizní kontrole - informaci, zda byly odstraněny závady z předchozí revize, informaci, jaké další revize byly na zařízení provedeny a následně doporučení, či případné další poznámky. Stranu uzavírá doporučený termín příští revize.

## Zpráva o revizi plynového zařízení

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

Jméno a adresa majitele:	Bytové družstvo Sedláčků b. p. Fizez 611 – přízemí vlevo Papírnická 590 Tutinov
Jméno a příjmení revizního technika:	ev. č. ověřování Martin Chlum ev. č. oprávnění

Druh revize: PROVOZNI

Označení zařízení (předmět revize): Zkoušení provozuschopnosti a bezpečnosti provozovaného domovního NTL rozvodu ZP od HUP, po sčítáč 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 účastní: p. Fizez

Celkové hodnocení: Zařízení je schopno bezpečného a spolehlivého provozu.

Upozorňuji na povinnost odstranit závady uvedené v bodě „C“ v navrhovaných termínech a dodržet podmínky uvedené v bodě „F a)“ této revizní zprávy. V opačném případě počítávám s tím, že platnost.

Zpráva obsahuje 3 strany.

V Tutinově dne: 22.1.2020

1 x uživatel  
1 x BD  
1 x revizní technik

podpis a otisk razítka

(a) Původní úvodní strana

A. Technické hodnoty revidovaného zařízení	D. Údaje o odstranění závad
C. Zvláštní závady a nedostatky, návrh opatření a říší k odstranění.	E. Záznam o ostatních revizích provedených na zařízení.
D. Údaje o odstranění závad z předchozích revizí, příloha: listovní.	F. Ostatní

A. Odběrná plynová zařízení je napájeno ze stávajícího provozovaného stoupačného NTL plynomodu odbočkou pr. 28, která je provedena v instalatérské sáčce na WC bytové jednotky. Tato odbočka je rovněž v sáčce na WC uložena kulovým kohoutem DN 25, který slouží jako HV odbočného plynového zařízení. Za nítu jsou provedeny vývody pro plynometr. Od plynometru je rozvod veden instalatérskou sáčkou, odkud poté prochází do kuchyně, kde je potřeba uloženo uzavřením, ze kterého je odpojen spotřebič. Kuchyně je přímo vzdušná a průtoková výměňovač. Kuchyně je přímo vzdušná a průtoková výměňovač. Rozvod je proveden z trubek měděných spojovaných lisováním. Celý rozvod je uláčen v kanálcích a třmenech.

Spotřebiče:  
- 1x Plynová deska Electrolux umístěna v kuchyni  
- kulový kohout DN 15

B. - Kontrola těsnosti přístupných rozbočovacích spojů revidovaného zařízení byla provedena detektorem plynu Sewerin Snoopar H mini - v.č. 0221002907  
- Kontrola funkce a úplnosti ovládacích a bezpečnostních armatur na zařízení  
- Kontrola spotřebičů (odbočka a bezpečnostní prvky, funkce)  
- Kontrola výskytu CO v místnosti byla provedena detektorem přístrojem Testo 1310 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:  
- výměna výměny hadice odpojovací spotřebič - překročena životnost 10 let 20.6.2020  
Plynová deska Electrolux bez závadových závad

D. Závady z předchozí revize byly částečně odstraněny.

E. Ostatní doložení revize:  
- zápis z předchozí provázení revize provedené na zařízení z roku 2017 - RT Martin Chlum

Strana 1 z 3  
F. a) pro zajištění správné a bezpečné funkce spotřebičů je nutné udržovat dostatečný přívod spotřebiče vzduchu do místnosti. Toto je zajištěno v technické zprávě popisovaným opatřením - udržovat tyto cesty průchod - odpovídá majitel (uživatel).

b) K revizi bylo doloženo:  
- zápis z předchozí provázení revize provedené na zařízení z roku 2017 - RT Martin Chlum

K revizi nebylo doloženo:  
- zápis o výtahové revizi a sezónní údržbě v provozem spotřebičů dle návodu výrobce - dohledat  
- zápis o výtahové revizi plynového zařízení - dohledat  
- dohledat 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

(b) Původní druhá strana

(c) Původní třetí strana

### Zpráva o revizi plynového zařízení

350/01/2022  
25.1.2022

Revizní technik: **JEFF WHITE**  
Číslo ověřování: **9229/9/16/N-P2-I-G**  
Číslo oprávnění: **2085/62/02/PZ-R-VI, VII**

**PROVOZNI**

Zkoušení provozuschopnosti a bezpečnosti nového rozvodu ZP od HUP instalace, přes regulátor a plynometr, po uzávěť před spotřebičem.

**Zařízení je schopno bezpečného a spolehlivého provozu.**

Upozorňuji na povinnost odstranit závady uvedené v bodě „C“ v navrhovaných termínech a dodržet podmínky uvedené v bodě „F a)“ této revizní zprávy. V opačném případě počítávám s tím, že platnost.

Zpráva obsahuje 3 strany.  
V Tutinově, dne 25.1.2022

1 x majitel  
1 x revizní technik

**JEFF WHITE**  
60296/1548/27/21  
280602/PZ

(d) Nová úvodní strana

### Technická zpráva

350/01/2022

A. Technické hodnoty revidovaného zařízení

B. Údaje o odstranění závad z předchozích revizí, příloha: listovní.

C. Záznam o ostatních revizích provedených na zařízení.

D. Údaje o odstranění závad z předchozích revizí, příloha: listovní.

E. Záznam o ostatních revizích provedených na zařízení.

F. Ostatní

a) Pro zajištění správné a bezpečné funkce spotřebičů je nutné udržovat dostatečný přívod spotřebiče vzduchu do místnosti. Toto je zajištěno v technické zprávě popisovaným opatřením - udržovat tyto cesty průchod - odpovídá majitel (uživatel).

b) K revizi bylo doloženo:  
- zápis z předchozí provázení revize provedené na zařízení z roku 2017 - RT Martin Chlum

K revizi nebylo doloženo:  
- zápis o výtahové revizi a sezónní údržbě v provozem spotřebičů dle návodu výrobce - dohledat  
- zápis o výtahové revizi plynového zařízení - dohledat  
- dohledat o servisní prohlídce spotřebiče - dohledat  
- projektová dokumentace - dohledat

Doporučený termín příští revize: 15.1.2022

Martin Chlum  
Revizní technik

(e) Nová druhá strana

(f) Nová třetí strana

Obrázek 4.6: Porovnání stran revizní zprávy

## 4.2.2 Výchozí revize

Struktura zprávy výchozí revize se od provozní revize nijak zásadně neliší. Jediný rozdíl je v segmentu A technické zprávy, kde u údajů spotřebiče je zaznamenána ještě maximální spotřeba zemního plynu. Důležité však zdůraznit, že součástí revize je také zápis o tlakové zkoušce, viz další podkapitola.

## 4.2.3 Zápis o tlakové zkoušce

Zápis o tlakové zkoušce obsahuje vlastní evidenční číslo a datum provedení, který je shodný s datem provedení výchozí revize. Dále obsahuje údaje o klientovi a revizním technikovi, který zkoušky vykonal. V další části je popsán předmět zkoušky, kým byla provedena montáž zařízení a jeho číslo oprávnění. Následují technické hodnoty zařízení a údaje o měření - typ zkoušky, zkušební tlak, délka zkoušky, její výsledek a název a výrobní číslo měřicího nástroje. Zápis zakončuje celkový výsledek zkoušky a krátké shrnutí.

**Zápis**  
o pevnostní a tlakové zkoušce odběrného plynového zařízení

ze dne: 30.05.2022 ev. č. 18/2022

**Jméno a adresa majitele:** Bešorna Miroslav  
Kiesewetrova p.p.č. 2656/564  
Trutnov

- Byl zkontrolován a vyzkoušen nový domovní rozvod ZP.

**Montážní práce provedla:** J.A. Kostas Ioanidis - č. opr. ITI 4509/6/13-PZ-M-fg

**Tlakovou zkoušku provedl:** Martin Chlum

**Technické hodnoty zařízení:** Jedná se o PE Dn 40 a měděné potrubí pr. 28 v celkové délce cca 16 m, spojované lisováním.

- Trubka PE Dn 40 - 13 m  
- Trubka měď pr. 28 - 3 m

**Použitý měřicí přístroj Esders DruckTest 2000 HMG 2 - v.č. 26D20382.**  
**Zkouška pevnosti** - zkušební tlak 195 kPa - délka 15 minut - výsledek vyhovující.  
**Zkouška těsnosti** - zkušební tlak 15 kPa - délka 15 minut - výsledek vyhovující.

**Výsledek zkoušek – vyhovující**

Prohlídkou odběrného plynového zařízení bylo zjištěno, že zařízení odpovídá požadavkům EN 1775, ČSN 386405, TPG G 704 01, G 800 03, G 93401 a ostatním souvisejícím předpisům. Zařízení vyhovělo předepsanému zkušebnímu přetlaku a souhlasím s uvedením do provozu.

**Poznámky a doplňky:**  
V Trutnově dne: 01.06.2022

Název organizace,  
která zkoušku provedla

Jméno a příjmení pracovníka,  
který zkoušku provedl

## Zápis

o pevnostní a tlakové zkoušce  
odběrného plynového zařízení

Evidenční číslo  
18/2022  
Datum provedení  
30.5.2022

Jméno a příjmení revizního technika  
Jeff White  
Číslo osvědčení  
9229/6/16/R-PZ-1.g  
Číslo oprávnění  
2085/6/92/PZ-R-VI,VII

Byl zkontrolován a vyzkoušen nový domovní rozvod ZP.

Montážní práce provedl(a)

fa. Montážní firma - číslo oprávnění ITI 4509/6/13-PZ-M-fg

Technické hodnoty zařízení

Jedná se o PE Dn 40 a měděné potrubí pr. 28 v celkové délce cca 16 m, spojované lisováním.

- Trubka PE Dn 40 - 13 m  
- Trubka měď pr. 28 - 3 m

Provedená zkouška	Zkušební tlak	Délka	Výsledek
Zkouška pevnosti	195 kPa	15 minut	Vyhovující
Zkouška těsnosti	15 kPa	15 minut	Vyhovující

Použitý měřicí přístroj  
Esders DruckTest 2000 HMG 2 - v.č. 26D20382.

Celkové hodnocení zkoušek

**Vyhovující.**

Prohlídkou odběrného plynového zařízení bylo zjištěno, že zařízení odpovídá požadavkům EN 1775, ČSN 386405, TPG G 704 01, G 800 03, G 93401 a ostatním souvisejícím předpisům. Zařízení vyhovělo předepsanému zkušebnímu přetlaku a souhlasím s uvedením do provozu.

V Trutnově, dne 30.5.2022

(a) Původní zápis o zkoušce

(b) Nový zápis o zkoušce

Obrázek 4.7: Porovnání zápisů o zkoušce



## 5 Implementace

Tato kapitola se již věnuje implementaci navrženého řešení. Popisuje zvolenou architekturu pro vývoj, jednotlivé technologie a funkční prvky aplikace. Zdrojový kód aplikace je k dispozici na odkaze <https://github.com/rbstr/Revizio>.

### 5.1 Použité technologie

Technologie byly zvoleny na základě požadavků na aplikaci a také osobních preferencích. Aplikace byla implementována jako multiplatformní webová aplikace s využitím principů progresivních webových aplikací. Z důvodu snížení serverové zátěže byla značná část aplikační logiky přesunuta do klientské části. Zvolené technologie jsou níže stručně popsány.

#### 5.1.1 Frontendová část

##### HTML5

HTML5 je nejnovější verze jazyka HTML, jenž slouží k určení logické struktury stránky. V moderních webových aplikacích se přenáší logická struktura aplikace na klientskou část a minimalizuje se tak serverová část. Stejný princip využívají i JavaScriptové frameworky, které přenášejí část generování HTML souborů ze strany serveru na stranu klienta, čímž dochází ke snížení zátěže u serveru. Mimo jiné je také možné takto výrazně snížit potřebný tok dat a načítat pouze ta data, která mají být následně klientem zpracována, až ve chvíli kdy si je klient vyžádá, bez nutnosti načítání celé stránky

##### JavaScript

JavaScript je základní technologie pro vývoj tzv. frontendu webových aplikací, která se nachází na straně klienta. V případě využití JavaScriptového frameworku se jedná o komplexní a funkční webovou aplikaci, která generuje HTML stránky využívající vlastní správu stavů, směrování i databáze.

##### React JS

React JS je JavaScriptová knihovna vyvíjená společností Meta, která umožňuje vývoj single-page aplikací. Je postavena na konceptu virtuálního DOMu, komponent

a jednosměrného toku dat.

v React existuje pro každý DOM objekt virtuální DOM objekt, který má stejné vlastnosti jako reálný DOM. Manipulace s virtuálním DOM je rychlejší, protože změny v něm se nemusí projevit hned a React tak může vyhodnotit, co je přesně potřeba změnit.

Komponenty jsou samostatné, nezávislé entity, které opisují část uživatelského rozhraní. Uživatelské rozhraní může být rozděleno na menší komponenty, kde má každá svůj vlastní kód, strukturu či API.

Data jsou směrované jednosměrně pomocí *State* a *Props*. To znamená, že ve vícekomponentové architektuře společný rodičovský komponent udržuje *State* a odevzdává ho prostřednictvím *Props* svým potomkům. [13]

## Redux

Redux je knihovna, která spravuje stav JavaScriptové webové aplikace. Původně byl vyvinut pro kooperaci s frameworkem React, nicméně dnes je možné jej používat s jakýmkoliv frameworkem či knihovnou. Stav (*State*) aplikace je uchovávan v úložišti (*Store*) a každá komponenta z tohoto úložiště má přístup k libovolnému stavu, se kterým potřebuje pracovat. [14]

## Bootstrap

Bootstrap je CSS framework pro vývoj mobile-first frontendové části webu. Obsahuje šablony pro typografii, textová pole, tlačítka a další prvky UI. Bootstrap poskytuje jednotný styl prvků skrze celý web, zároveň je však možné tyto styly přizpůsobovat tak, aby odpovídaly dalšímu obsahu webu. [15]

## Material UI

Material UI je knihovna React komponent, které implementují Google Material Design. Obsahují předpřipravené komponenty, které je možné ihned využívat. Poskytuje i možnost úprav, pomocí kterých je možné implementovat vlastní design systém. [16]

## Sass

Sass je preprocesorový skriptovací jazyk, který je interpretován nebo kompilován do CSS. Umožňuje používat proměnné, vnořená pravidla nebo funkce. Sass pomáhá při organizaci a přehledňuje tak celkový kód. [17]

## Yup

Yup slouží k validaci a analýze dat, které jsou za běhu vkládány uživatelem. [18]

## **PDF-Lib**

PDF-Lib je JavaScriptová knihovna pro manipulaci a zejména modifikaci PDF dokumentů. Umožňuje vytvářet PDF dokumenty od základu, či upravovat již vytvořené pomocí vkládání textů, obrázků nebo křivek. Zároveň nabízí možnost přidávat, odstraňovat stránky či spojovat více PDF dokumentů do jednoho. [19]

## **Chart.js**

Chart.js je JavaScriptová knihovna, jenž poskytuje sadu grafů, pluginů a možnosti přizpůsobení. Nabízí také vytváření vlastních anotací a interakcí s grafy. [20]

## **SendGrid**

SendGrid slouží k zasílání emailových zpráv. Tato platforma je vhodná pro zasílání téměř všech druhů emailů. Nabízí monitoring zasílání, tudíž uživatel může vidět, zda jeho emaily byly doručeny, otevřeny či byly označeny jako spam. [21]

### **5.1.2 Backendová část**

Pro backendovou část byl použit výhradně ekosystém Firebase od společnosti Google. Ten byl zvolen z důvodu široké nabídky požadovaných funkcionalit na webovou aplikaci a možnosti mít vše na jednom místě.

#### **Firestore**

Firestore je NoSQL databáze k ukládání a synchronizaci dat. Databáze je flexibilní a rozšiřitelná pro použití na mobilní, webový či serverový vývoj. V reálném čase udržuje synchronizaci dat napříč klientskými uživateli a nabízí podporu offline režimu.

#### **Storage**

Storage nabízí ukládání a přístup k všem různým druhům dokumentů, jako například fotografiím, videím nebo PDF dokumentům.

#### **Cloud Functions**

Cloud Functions je serverless (z pohledu uživatele) framework sloužící k automatickému spouštění funkcí v backendu, které jsou spouštěny pomocí HTTPS požadavků, nebo reagují na určitou akci vyvolanou událostmi v ekosystému Firebase.

#### **Cloud Messaging**

Cloud Messaging umožňuje zasílání zpráv/oznámení napříč různými platformami.

## **Firebase Authentication**

Firebase Authentication slouží k ověřování uživatelů v rámci backend funkcí a hotových UI knihoven. Ověřování je možné pomocí emailových adres, telefonních čísel nebo celkových účtů u Google, Apple nebo Facebook.

## **Firebase Hosting**

Firebase Hosting poskytuje zabezpečený hosting pro webové aplikace. Nabízí využití Google Analytics a Firebase Performance Monitoring k analýze webových aplikací. Výhodou je možnost použití vlastní domény. [22]

## 5.2 Architektura aplikace

Architektura aplikace je založena na využití *State* managementu Redux (5.1.1), jenž vyháází z architektury zvané Flux. [23] Jejím hlavním cílem je řídit aplikaci s globálním stavem (*State*) a usnadnit sdílení dat mezi komponentami. Globální stav se nachází v tzv. *Store* a pokud uživatel zaktivuje určitou funkci, prochází *Reducerem*, který přebírá předchozí globální *State* a funkci a vrací nový globální stav. Ve chvíli, kdy je *Store* aktualizován pomocí *Reduce* funkce je provedeno opětovné vykreslení *DOM*, jinými slovy dojde k aktualizaci UI. Architektura Redux zajišťuje, že všechny změny ve *Store* probíhají jedna po druhé, v sekvenčním pořadí. Vhodným doplňkem při využívání Reduxu je Redux DevTools, což je rozšíření pro webové prohlížeče Google Chrome a ostatní, jenž jsou postaveny na tomto prohlížeči, např. prohlížeč Brave. [24]



Obrázek 5.1: Architektura Redux

### 5.2.1 Frontend

Frontend aplikace využívá, z důvodu komplexnějšího řešení, vícenásobný Reducer, který byl vytvořen pomocí funkce *combineReducers*, díky čemuž je umožněna separátní správa stavů. Aplikace konkrétně obsahuje šest dílčích reducerů: *Auth*, *Client*, *Defect*, *Pattern*, *Revision* a *Meeting*. Ty jsou pojmenovány podle tzv. *Slice*, které spravují.

*Slice* jsou vytvořeny pomocí zjednodušené funkce *createSlice*, jenž je součástí knihovny Redux-Toolkit. *Slice* definuje jakým způsobem se bude aktualizovat stav a to pomocí definovaných funkcí. Jednotlivé *Slice* a jejich funkce jsou zaznamenány v následujícím přehledu.

- **Auth** (práce s uživateli) - *setUser*, *logOut*
- **Client** (práce s klienty) - *resetClientFields*
- **Defect** (práce se závadami) - *setType*
- **Pattern** (práce se vzory) - *setPatternIdToEdit*
- **Revision** (práce s revizemi) - *resetRevisionForm*, *setPdfRevisionData*, *setBasicInformation*, *setPerformedTests*, *setTechInformation*, *setFoundDefects*, *setAdditionalInformation*, *setRevisionData*, *setEditRevision*
- **Meeting** (práce se schůzkami) - *setMeetingData*

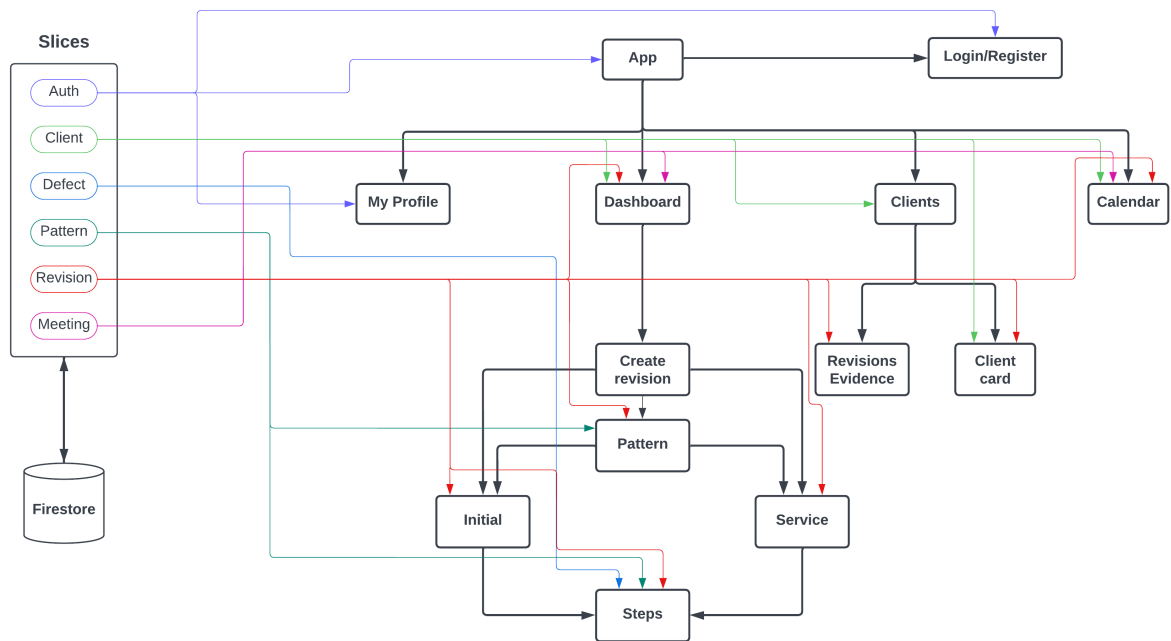
Například Slice *Auth* obsahuje dvě funkce reduceru: *setUser* a *logOut*. První uvedená zavádí do stavu údaje o přihlášeném uživateli a druhá naopak tyto údaje ze stavu odstraňuje.

Slice zároveň definuje funkce s využitím middlewaru *AsyncThunk*, jehož úkolem je provádět zpožděné, asynchronní funkce ještě před samotným odesláním výsledku do reduceru. Asynchronní funkce jsou vytvořeny pomocí funkce *createAsyncThunk* a ve většině případů jsou využity ke komunikaci se serverovou částí, databází Firebase. Jednotlivé asynchronní funkce příslušných Slice jsou zaznamenány níže.

- **Auth** - *register, loginWithGoogle, login, logout, getProfile, getSettings*
- **Client** - *createClient, getClient, getClients, DeleteClient, getClientByEmail*
- **Defect** - *createDefect, getDefect, deleteDefect, getDefects, getCommonDefects*
- **Pattern** - *createPattern, getPattern, getPatterns, renamePattern, deletePattern*
- **Revision** - *createRevision, downloadRevisionPdfFile, dashboard, getRevision, getRevisionsById, getRevisions, deleteRevision*
- **Meeting** - *createMeeting, getMeeting, getMeetings, getMeetingsForCount, deleteMeeting*

Například funkce *getCommonDefects* slouží k získání nejčastějších závad k zařízením/spotřebiči. Ta vyhledává v databázi zadaný název a porovnává jej s předešle vytvořenými revizemi všech zaregistrovaných uživatelů. Následně ze získaných dat vybírá tři nejčastěji se vyskytující a předává jej příslušné komponentě.

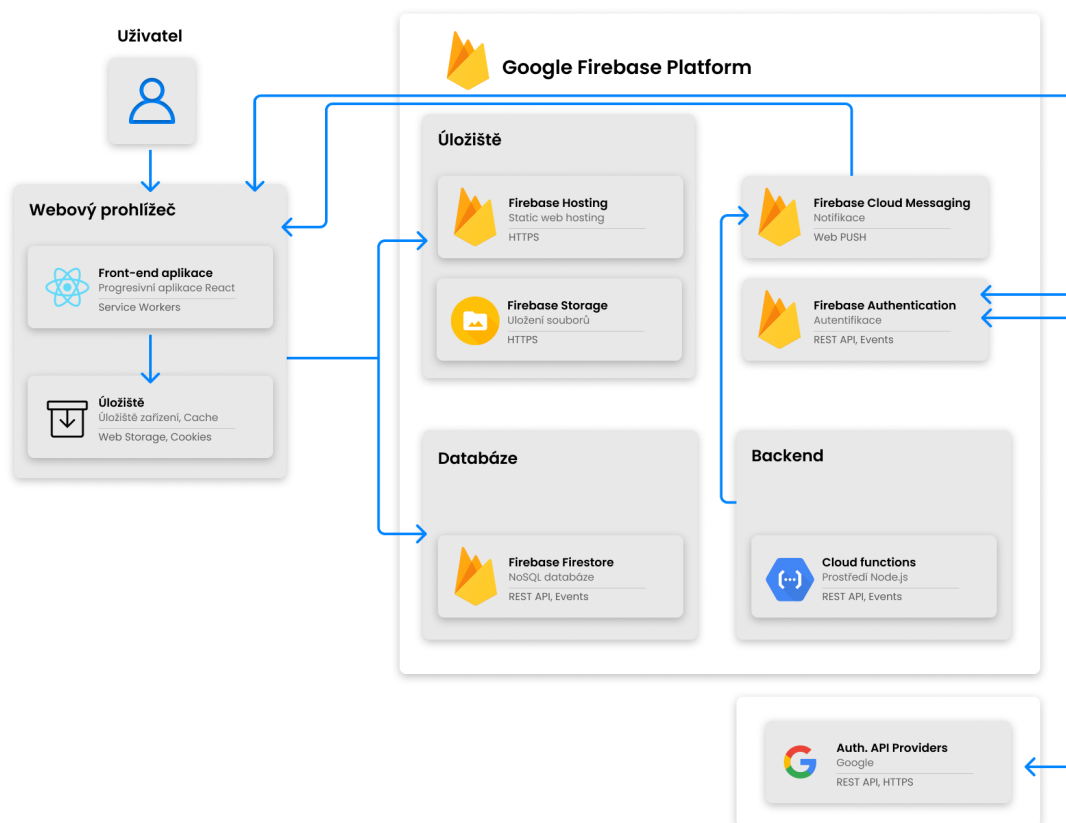
Aplikace obsahuje 4 základní stránky/komponenty (*Dashboard, Clients, Calendar* a *My Profile*), ve kterých jsou zahrnuty další dílčí stránky, respektive komponenty. Komponenty je možné rozdělit na hloupé a chytré. Hloupé pouze vykreslují hodnoty předané z parametru a jedná se o klasické React komponenty. Chytré komponenty naopak využívají data ze stavu. To je vhodné v případě znovupoužitelnosti komponent bez nutnosti dalších úprav. Příkladem chytré komponenty může být komponenta *Table*, která se vyskytuje na stránce *Clients*, respektive *Revisions evidence*. Ta vykresluje data na základě poskytnutého názvu middlewaru, tedy *getClientsAsyncThunk* (seznam klientů) nebo *getRevisionsAsyncThunk* (seznam revizí). Název middlewaru je získán ze *Store*, což je objekt, který v sobě uchovává uložená data aplikace. Na Obrázku 5.2 je vyobrazeno schéma frontendu aplikace - jednotlivých komponent a příslušných Slice.



Obrázek 5.2: Schéma frontendové části

## 5.2.2 Backend

Jak již bylo uvedeno výše, serverová část je kompletně pod záštitou ekosystému Google Firebase. Nicméně konkrétně Cloud Functions bylo nutné implementovat manuálně. Aplikace využívá celkem čtyři takové funkce, které mají za úkol vytvořit notifikace pro expiraci revizních kontrol (*createNotification*) a vytvořených schůzek pro aktuální den (*createMeetingNotification*), odeslání těchto notifikací uživateli (*notify*) a odeslání emailových zpráv klientům (*sendEmail*). Pro lepší ilustraci komunikace uvnitř Firebase a napojení na aplikaci bylo vytvořeno schéma aplikace, které je vyobrazeno na Obrázku 5.3.



Obrázek 5.3: Schéma backendové části

## 5.3 Funkční prvky

V této podkapitole je popsán výběr funkčních prvků, jenž byly definovány v požadavcích na aplikaci.

### 5.3.1 Autentizace uživatelů

Aplikace nabízí standardní způsob autentizace uživatele. Uživatel může využít klasické registrace pomocí emailové adresy a hesla nebo využít přihlášení pomocí účtu Google. Původním záměrem bylo implementovat přihlášení i pomocí účtu Apple, jelikož však je k této funkci nutné mít potřebná oprávnění a certifikaci vývojáře produktů Apple, bylo od této myšlenky oproštěno.



### 5.3.2 Nejčastější závady na zařízení

Uživatel má možnost si zjistit, ještě před samotnou diagnostikou zařízení, jaké závady se na konkrétním zařízení vyskytují nejčastěji. K tomu slouží tlačítko umístěné pod textovým polem názvu zařízení. Po vyplnění názvu zařízení je tlačítko zaktivované a po stisku nabídne tři nejčastěji se vyskytující závady. V případě, že na zařízení ještě nebyl proveden dostatek revizních kontrol, či zařízení není poruchové, je o této skutečnosti uživatel obeznámen. Data potřebná k provozu této funkce se získávají napříč celou aplikací. Zhotovené revizní zprávy jednoho revizního technika tak mohou pomoci v práci dalších revizních techniků.

### 5.3.3 Definování závad

Při vytváření revize, uživatel na třetí straně zapisuje nalezené závady na zařízení a celkovém zapojení. Závady jsou rozděleny na tři typy: závady bránící bezpečnému provozu, závady nebránící bezpečnému provozu a závady na zařízení. Uživatel může vybírat z již přednastavených závad, které vychází z aktuálního seznamu závad. Má však i možnost si do seznamu uložit nové závady, které se v seznamu aktuálně nenachází. Z důvodu zpřehlednění jsou v seznamu data uložena pomocí objektů, které obsahují dvě hodnoty: zkrácený název a úplný popis. Proto při ukládání nové závady je uživatel vyzván k vyplnění zkráceného názvu, který bude zobrazen v seznamu závad, a úplného popisu, který bude zobrazen v revizní zprávě.

### 5.3.4 Elektronický podpis a razítko

Revizní zpráva je posuzována jako právní dokument a jako takový musí obsahovat potřebné náležitosti. Těmi jsou myšleny zejména čísla oprávnění a osvědčení, razítko a podpis revizního technika, který revizní kontrolu prováděl. Čísla oprávnění a osvědčení a razítko se ukládají na stránce uživatele, které jsou při vytváření revize následně vloženy do zprávy. Na poslední straně vytváření revize má poté uživatel možnost vložit elektronický podpis. Podpis byl naimplementován pomocí knihovny *react-signature-canvas*[25], která nabízí tzv. pad, který zaznamenává pohyb po vymezené oblasti a po uložení převádí podpis do formátu SVG.

### 5.3.5 Export revizní zprávy

Jakmile uživatel vyplní všechna požadovaná data, jsou tato data předána k zápisu do předpřipravených šablon revizních zpráv. Šablony jsou rozlišeny na základě typu revize, tedy provozní a výchozí. Dalším rozlišujícím faktorem je přítomnost správce budovy. Ten se vyskytuje zejména u panelových domů, kde figuruje například jako družstvo. Aplikace také obsahuje šablonu pro vyplnění Zápisu o tlakové zkoušce, která je nedílnou součástí výchozí revize. K vytvoření PDF dokumentu je využita knihovna *PDF-Lib* (5.1.1), kdy je nejprve načtena vyhovující šablona, jsou definována písma a poté již pomocí funkce *drawText* nebo *drawImage* jsou do dokumentu zapsána požadovaná data, v rámci definovaných vlastností.

### 5.3.6 Uložení a načtení vzoru zpráv

Po úspěšném dokončení revizní zprávy je nabídnuto uživateli uložení průběhu revize jako vzor. To znamená, že jsou uložena všechna vstupní data a naměřené hodnoty a je možné je využít kdykoliv v budoucnu. Tato funkce může revizním technikům usnadnit práci v případě, že se průběh revize odehrává v panelovém domě, kde se značná část dat shoduje. Při ukládání vzoru revize je uživatel povinen uvést název vzoru, tak aby při pozdějším použití byly další vzory od sebe jednoznačně identifikovatelné. Použití těchto vzorů je možné dvěma způsoby: uživatel může zvolit načtení ze vzoru na úvodní straně vytváření revizní zprávy, kde se nachází rozevírací seznam pro provozní revizi, výchozí revizi a uvedené načtení ze vzoru, druhým způsobem je pak na stránce klienta, kdy po kliknutí na možnost u dokumentu je možné zvolit volbu „Použít jako vzor“.

### 5.3.7 Odesílání revizních zpráv pomocí emailu

Uživatel má možnost využít integrovaného systému na odesílání emailů. Může tak učinit na poslední stránce při vytváření revize nebo na kartě klienta, kde po otevření možností u dokumentu klienta vybere možnost pro odeslání emailu. V obou případech se uživateli zobrazí tzv. *modal*, ve kterém jsou předdefinované hodnoty, jako email klienta a předmět emailu, který obsahuje datum provedené revize. Dále je předdefinované oslovení a pozdrav na konci emailové zprávy, doplněný o údaje uživatele (jméno, email, telefonní číslo). Samotný text emailu je však plně v kompetenci uživatele. Pro implementaci této funkce byla využita služba *SendGrid* (5.1.1) v součinnosti s *Firebase Cloud Functions*.

### 5.3.8 Evidence revizních zpráv

Každý revizní technik je povinen vést si evidenci provedených revizí. Proto bylo nutné v aplikaci vytvořit také samostatnou stránku, kde se nachází všechny provedené revize uživatelem aplikace. Ta obsahuje provozní a výchozí revize, ale také zápisy o tlakové zkoušce, jež je nutné vést také samostatně, nikoliv jen jako součást výchozí revize.

### 5.3.9 Využití map

Aplikace využívá *Google maps* k zobrazení adres budov na mapě. Uživatel může tuto funkci využít u karty klienta, kde je zaznamenána kompletní adresa. Napomáhá reviznímu technikovi k lepší identifikaci budovy a celkové orientaci. K provozuschopnění *Google maps* bylo nezbytné vytvoření API klíče na platformě *Google Cloud*.

### 5.3.10 Kalendář

Aplikace disponuje vestavným kalendářem, který umožňuje plánování schůzek a revizních kontrol. Kalendář je rozdělen na dvě části. Kompletní kalendář, který figuruje

na stránce určené výhradně jemu, je naimplementován pomocí knihovny *react-big-calendar*[26] a uzpůsoben tak, aby korespondoval s vizuální stránkou aplikace. Zde jsou umožněny standardní úkony, jako přepínání mezi měsíci a dny, zobrazení aktuálního dne s kompletní agendou a tlačítko pro vytvoření nové schůzky. Při deklaraci nové schůzky uživatel definuje jméno a příjmení klienta, dále datum a odhadovaný časový rozptyl. Nepovinným údajem je adresa schůzky, obvykle adresa budovy klienta. Uživatel má zároveň výše uvedené informace možnost vyplnit automaticky v případě, že se jedná o schůzku s již uloženým klientem. Druhá část kalendáře je zobrazena na přehledové stránce aplikace, ta nabízí zjednodušený pohled na nejbližší dny a informuje o nadcházejících schůzkách příslušných dnů, kde je zároveň možné zobrazit místo schůzky na mapě. Základním stavebním kamenem této komponenty je knihovna *Swiper*, která slouží k pohybu mezi jednotlivými dny pomocí prstu, či myši.

### 5.3.11 Statistiky

Přehledová stránka aplikace nabízí hned dvě statistická okénka. Jedno okénko zaznamenává celkový počet provedených revizí uživatelem a pro zpestření je zde vytvořen i ukazatel aktuální úrovně uživatele. Další okénko poté porovnává vývoj provedených revizí v jednotlivých měsících. Graf v této komponentě byl naimplementován pomocí knihovny *Chart.js* (5.1.1).

## 5.4 PWA prvky

Stežejní vlastností aplikace byla implementace prvků progresivních webových aplikací, které byly zaznamenány v rámci ostatních požadavků na aplikaci a blíže definovány v kapitole 2.3.

### 5.4.1 Instalace aplikace

Aplikace plně splňuje požadavky, aby jí bylo možné nainstalovat na zařízení. Existují dva způsoby instalace. Uživatel je při návštěvě stránky dotázán webovým prohlížečem, zda si přeje aplikaci nainstalovat na své zařízení. Druhý způsob je pomocí připraveného tlačítka, které je umístěné na přehledové stránce. Zde je potřebné, aby se tlačítko zobrazovalo pouze tehdy pokud ještě aplikace není nainstalována. K tomu slouží událost *beforeinstallprompt*, která tuto skutečnost sleduje.

Důležité zmínit, že výše uvedené způsoby zcela nefungují v prohlížeči Safari. Na zařízeních s operačním systémem MacOS není instalace možná vůbec a nepodporují tak ani událost *beforeinstallprompt*. Na zařízeních s operačním systémem iOS tato událost také není podporována, nicméně aplikaci lze nainstalovat ručním způsobem pomocí tlačítka „sdílet“ a následné volby „přidat na plochu“. Bylo tedy nutné, ještě před samotným zobrazením tlačítka pro instalaci, zjistit, v jakém webovém prohlížeči je aplikace otevřena.

### 5.4.2 Notifikace

Notifikace aplikace lze rozdělit na dva stupně. U prvního stupně jsou oznámení povoleny pouze v rámci aplikace a uživatel může být upozorněn o nových okolnostech skrze vytvořené notifikační centrum. U druhého stupně již dochází k notifikacím přímo na zařízeních, tak jak je to běžné u nativních aplikací. Tato činnost je však podložena nutností povolení oznámení od uživatele.

První stupeň využívá k zaslání notifikací Firebase Cloud Functions, kde byly naimplementovány funkce nejprve pro vytvoření notifikace a posléze k odeslání notifikace. Při vytvoření nové revizní zprávy uživatel definuje doporučený termín příští revize, a toto datum je použito k vytvoření plánu k odeslání oznámení. Bylo nastaveno, aby se oznámení uživateli zobrazilo jeden týden před expirací revizní kontroly. Jakmile je této podmínky dosaženo, notifikace je odeslána příslušnému uživateli do notifikačního centra. Dále aplikace upozorňuje uživatele o naplánovaných schůzkách aktuálního dne.

Druhý stupeň je postaven na službě Firebase Cloud Messaging (FCM) a balíčku *firebase/messaging*. Při spuštění aplikace je uživateli nabídnuto povolení oznámení, kdy při potvrzení je pro příslušného uživatele vytvořen token. Ten je uložen v databázi společně s jedinečným ID, protože uživatel může mít více zařízení. Vytvoření tohoto tokenu, dotázání na oznámení a nebo uložení tokenu v FCM má na starost funkce *getToken()*. Pokud uživatel při dotazování oznámení odmítne, může je později povolit v nastavení svého profilu. Protože aktuálně nejsou webové notifikace

dostupné na zařízeních iOS, bylo nutné ošetřit, aby se uživateli nezobrazovala bílá obrazovka v případě, že by chtěl oznámení povolit. Bílá obrazovka se zobrazuje v návaznosti na chybějící rozhraní *Notification*. Nejprve tedy bylo nutné zjistit, zda se uživatel nenachází na zařízení využívající operační systém iOS. V případě, že ano, je povolení oznámení zablokováno a je zobrazena chybová hlášení.

### 5.4.3 Offline mód

Díky Service Workeru jsou uloženy do *cacheStorage* všechny statické soubory typu JavaScript, HTML nebo CSS. Dále pak všechny využívané fonty z rodiny Poppins a loga aplikace. Zároveň jsou ukládány do paměti všechny soubory z Firebase Storage, které byly navštíveny v rámci pohybu po aplikaci - revizní zprávy a razítko.

Samotná data uživatelů a klientů jsou k dispozici pomocí nastavení databáze Firestore a použití funkce *enableIndexedDbPersistence()*.

## 5.5 Datový model

Pro databázi aplikace bylo možné volit mezi Firebase Realtime nebo Firestore databází. Zásadním rozdílem mezi těmito databázemi je jejich struktura a dostupnost v offline režimu. Realtime nabízí JSON stromovou strukturu a Firestore uložení dat v podobě kolekcí. Firestore je dostupný v offline režimu pro všechny webové klienty (iOS, Android, Web), ale Realtime je dostupný pouze pro první dva uvedené. Z toho důvodu byla pro tuto práci zvolena Firestore databáze.

Jak již bylo uvedeno, data jsou ukládána do dílčích kolekcí a jejich dokumentů, kde jsou v podobě klíč-hodnota. Dokumenty jsou tak v podstatě typu JSON. Aplikace obsahuje celkem devět kolekcí: *client*, *dashboard*, *defects*, *meetings*, *notifications*, *performedTests*, *profiles*, *revisions* a *settings*. Popis jednotlivých kolekcí je uveden v Tabulce 5.1.

Tabulka 5.1: Popis jednotlivých kolekcí databáze

Název	Obsah
Clients	Údaje o klientovi
Dashboard	Analytické podklady
Defects	Nově definované závady
Meetings	Údaje o schůzkách
Notifications	Údaje o upozorněních
PerformedTests	Vytvořené zápisy o zkouškách
Profiles	Údaje o uživatelích
Revisions	Vytvořené revize
Settings	Nastavení uživatelů

## 5.6 Zabezpečení

Vzhledem k tomu, že progresivní webové aplikace jsou prosté webové stránky, je nutné se zaměřit i na jejich zabezpečení. Podobně jako webové stránky se mohou stát terčem různých metodik napadnutí, jako například Cross-Site Scripting (XSS) nebo Cross-Site Request Forgery (CSRF).

### 5.6.1 Frontendová část

V případě, že se provádí na stránce určitý proces, spustí se JavaScript, který získá určité údaje z vstupních polí, v tomto případě například jméno klienta, které následně zpracuje. Útočník však může pomocí tohoto pole spustit nějaký další kód, skript, který může být pro aplikaci závadný. Jak již bylo uvedeno klientská část, neboli frontend, je vyvinuta pomocí knihovny React. Tento framework, s použitím syntaktického dodatku JSX, v sobě obsahuje tzv. sanitizační nástroje, které obsah každé proměnné, kterou se chystá React vykreslit, nejprve převede do typu String.

Tedy tak, aby výsledkem nebyl spustitelný kód. Použije-li útočník k vyplnění jména klienta text `<script>alert('ahoj');</script>` a pokusí se jej předat JavaScriptu, výsledkem bude uložení klienta se jménem `<script>alert('ahoj');</script>` a kód útočníka tak nebude zpracován.

Samozřejmě existují případy XSS útoků, které React neošetřuje a je nutné jejich zabezpečení. Jedná se například o případ, kdy v aplikaci uživatel zadává URL stránky, ze které následně aplikace stahuje nějaký skript. Zde je nutné využít atribut `dangerouslySetInnerHTML`, který sanitizační nástroje React obchází. To ovšem vyžaduje od programátora, aby tuto funkci zabezpečil manuálně. Nicméně v této práci tento případ nebylo nutné implementovat. [27]

## 5.6.2 Backendová část

Backendová část aplikace se plně nachází na platformě Google Firebase, z toho důvodu nebylo nutné provádět žádná zásadní bezpečnostní opatření samostatně. Nicméně Firebase doporučuje několik bodů, které je záhodno v aplikaci zvážit.

- **Monitoring a oznámení pro backend služby**

V práci byl nastaveno monitorování a oznámení pro Cloud Functions, Firebase Firestore, Cloud Storage a Hosting. Tímto je možné detekovat zneužívající provoz, jako jsou například útoky Denial-Of-Service (DOS).

- **Konfigurace Cloud Functions**

Cloud Functions jsou automaticky nastaveny, aby splňovaly požadavky aplikace. To však v případě napadení může znamenat vysoké poplatky za služby. Z toho důvodu bylo nutné nastavit limity pro souběžné operace na základě běžného používání.

- **Nastavení pravidel**

Pravidla slouží jako most mezi uživateli a daty. Využívají svůj vlastní jazyk pro definování podmínek použití, ale nejedná se o nic složitého, protože tento jazyk je založen na syntaxi JavaScriptu a struktuře JSON. V rámci aplikace bylo nastaveno, že uživatel má přístup pouze ke svým dokumentům a datům, se kterými může manipulovat. Samozřejmě tento uživatel musí být řádně provednou autentizací.

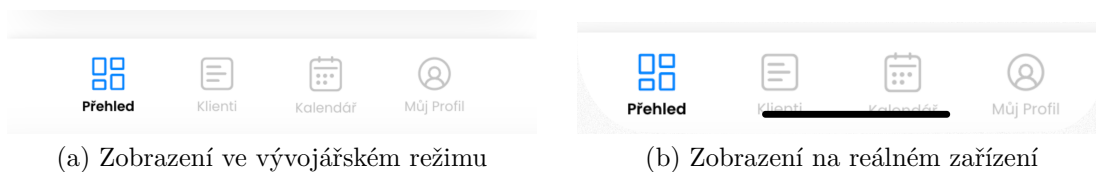
## 6 Testování

Tato kapitola popisuje metodiky testování prováděné na aplikaci, jak v průběhu vývoje, tak i v jeho finální fázi.

### 6.1 Testování v průběhu vývoje

Aplikace byla v průběhu vývoje testována nespočetněkrát, kdy testování probíhalo na několika částech aplikace. Testování bylo zaměřené na převod UI z programu Figma do webové aplikace, responzivitu, funkčnost definovaných funkčních prvků, správné fungování využívaných API a samozřejmě také na platformu Google Firebase. Na testování během vývoje byl kladen velký důraz, aby finální testování určené uživateli nevykazovalo žádné triviální chyby, které by mohly zamezit dalšímu testování důležitých funkcí.

Testování probíhalo s využitím vývojářských nástrojů v prohlížeči, kde je možné si nasimulovat zobrazení na různých zařízeních, ale také i pomocí reálných zařízení. Tento způsob ukázal několik nedostatků, které mohly být přehlédnuty. Zejména na Obr. 6.1 je vidět rozdíl zobrazení mobilního menu při využití vývojářského režimu a reálného zařízení. Zde je možné si všimnout, že na reálném zařízení, kterým bylo zařízení Apple iPhone XS, tzv. *Home Bar* narušuje text u jednotlivých ikon. Bylo tedy nutné celkové menu posunout o několik pixelů výše. Dalším nalezeným nedostatkem bylo například vykreslování obrázků ve formátu SVG v prohlížeči Safari. To bylo způsobeno tím, že Safari neumí správně vykreslovat tag *img* s SVG formátem pro displeje retina.



Obrázek 6.1: Porovnání zobrazení menu



## 6.2 Uživatelské testování

V rámci uživatelského testování bylo možné díky předchozímu testování použitelnosti (viz kapitola 4.1.3) vynechat testování týkající se designové stránky aplikace (zejména UX) a plně se soustředit na testování funkčních prvků.

Uživatelského testování se zúčastnili celkem dva vybraní revizní technici, kteří dostali k dispozici již finální, plně odladěnou verzi aplikace, tak aby bylo možné případně najít chyby i na těch nejméně exponovaných místech. Testování bylo rozdělo na dvě části, přičemž první část se věnovala definovaným scénářům a celkovému seznámení respondentů s aplikací. Druhou částí bylo vyzkoušení aplikace při reálném použití se zákazníkem.

### 6.2.1 Průběh testování

Oba respondenti se při první části testování řídili následujícími scénáři:

- **Scénář 01:**  
Zaregistrujte se v aplikaci a uložte své osobní údaje.
- **Scénář 02:**  
Vytvořte provozní revizi, následně použijte tuto revizi k vytvoření další revize.
- **Scénář 03:**  
Vytvořenou provozní revizi upravte a odešlete klientovi pomocí emailové zprávy.
- **Scénář 04:**  
Vytvořte výchozí revizi, uložte ji jako vzor a následně ji uložte do svého zařízení.
- **Scénář 05:**  
Přidejte do kalendáře nový termín schůzky s klientem.
- **Scénář 06:**  
Přidejte si aplikaci na plochu svého zařízení.

Po úspěšném seznámení respondentů s aplikací bylo možné aplikaci vyzkoušet v reálném prostředí u zákazníka. Předmětem testování byla bytová jednotka panelového domu ve městě Trutnov. Jednalo se tedy o nejčastější případ provádění revizní kontroly. Výsledek testování je blíže popsán v následující kapitole.

### 6.2.2 Výsledek testování

Uživatelské testování přineslo víceméně pozitivní ohlasy respondentů, kteří hodnotili zejména funkční prvky aplikace. Shodli se, že výsledné protokoly zcela splňují požadavky pro využití v reálném použití. Ocenili rychlost a intuitivnost aplikace.

Nicméně byl nalezen i jeden nedostatek. Jednalo se o případ, který nebyl zohledněn při provádění analýzy a následném návrhu aplikace. Konkrétně bylo nutné vyřešit případ, kdy klient má více adres, respektive budov. Takový případ se objevuje zřídka, nicméně aplikace by měla tuto skutečnost brát v potaz. V případě, že je vytvořena nová revize, jenž obsahuje identické osobní údaje již existujícího klienta a rozdílné údaje u budovy klienta, je tato budova přidělena ke stávajícímu klientovi. Z pohledu uživatele se poté mění zobrazení stránky klienta, kdy je k údajům o budově přiděleno tlačítko pro iteraci mezi budovami (viz Obr. 6.2).

**Adresa budovy** [Další budova >](#)

Ulice  
**Arbesova 2**

Město  
**Děčín**

[Ukázat na mapě](#)

PSČ  
**40502**

Obrázek 6.2: Klient s více adresami

## 6.3 PWA testování

PWA testování probíhalo ve smyslu výkonnosti, což by měla být jedna z devíz PWA. Testování bylo posuzováno dle doby načtení stránky a jejího obsahu. PWA disponují jedinečným kešovacím mechanismem, z důvodu použití Service Workeru.

Aby nedocházelo k různorodým testovacím prostředím, bylo nastaveno konzistentní prostředí, kdy testování probíhalo v prohlížeči Brave s definovanou rychlostí internetového připojení - 5G. Úvodní test simuloval příchod nového uživatele do aplikace, tedy došlo k vymazání všech dosavadních dat aplikace uložených v prohlížeči (Service Worker, lokální úložiště, Cookies apod.). Následující test simuloval návrat stálého uživatele do aplikace, s již uloženými daty z předchozích návštěv. Obě měření byla nejprve provedena bez použití Service Workera a následně s jeho použitím. K eliminaci případných anomálií byla jednotlivá měření provedena čtyřikrát a poté byly časy zprůměrovány. Průběh měření je zaznamenán v Tabulce 6.1.

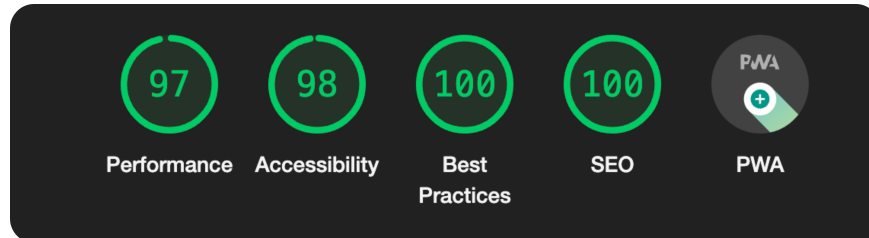
Tabulka 6.1: Tabulka měření výkonu

	1. načtení	2. načtení	Zlepšení
<b>Bez Service Workeru</b>	2,17 s	1,12 s	48 %
<b>Se Service Workerem</b>	1,84 s	1,02 s	44 %

Měření ukázalo, poměrně překvapivě, že výkon aplikace se zásadně nemění bez použití Service Workeru. Zlepšení při druhém načtení aplikace je také téměř totožné.

### 6.3.1 Lighthouse

Dalším způsobem testování PWA a webových aplikací obecně, je Lighthouse. Toto rozšíření pro prohlížeče umožňuje automatické testování v rámci výkonnosti, přístupnosti, SEO (Search Engine Optimization) a zabezpečení aplikace. Výsledek testování je zobrazen na následujícím Obrázku 6.3.



Obrázek 6.3: Hodnocení Lighthouse

## 7 Závěr

Tato diplomová práce měla za hlavní cíl navrhnout a vytvořit progresivní webovou aplikaci pro revizní zprávy. Vývoj aplikace byl založen na provedené analýze problematiky revizních zpráv a také progresivních webových aplikací. K ucelenému seznámení s průběhem revizních kontrol bylo spolupracováno s respondenty z řad revizních techniků.

Na základě získaných poznatků byly sepsány funkční a ostatní požadavky na aplikaci, které následně formovaly návrh aplikace. Ten byl vytvořen v programu Figma, jenž umožňuje vytvářet klikatelné prototypy a zvizualizovat tak podobu aplikace, bez nutnosti programování. Posledním bodem návrhu bylo testování použitelnosti, které poukázalo na nedostatky v návrhu, jenž bylo nutné opravit. Následně byla pro aplikaci vytvořena vizuální identita, bylo definováno logo, typografie a barevná paleta aplikace.

Progresivní webová aplikace byla vytvořena za pomoci frameworku React Js, na základech architektury Redux. Komponenty aplikace vychází z knihovny Material UI a CSS frameworku Bootstrap, který napomáhá k nativnímu vzhledu aplikace v mobilním zobrazení. Frontend aplikace využívá externí knihovny, jako např. Google maps, PDF-lib, či SendGrid, pomoci kterých jsou naimplementovány požadované funkcionality. Aplikaci je možné nainstalovat na zařízení a je možno ji používat v offline režimu. Zároveň aplikace disponuje notifikacemi, které uživatele upozorňují na blížící se expirace revizí, či domluvené schůzky.

Serverová část aplikace je vyvinuta v rámci ekosystému Google Firebase, kde je využívána databáze Firestore, úložiště Storage a nebo například Cloud Functions. Ty slouží k vytváření a zasílání notifikací a odesílání emailových zpráv klientům.

Aplikace byla testována konstantně během vývoje a zejména pak na konci vývoje, kdy proběhlo uživatelské testování s revizními techniky. Ti během testování nejprve vykonali připravené testovací scénáře a následně aplikaci otestovali během reálného použití u klientů. Výsledkem testování je konstatování, že aplikace je vhodná k vytváření revizních zpráv a zefektivňuje práci revizních techniků. Bylo však upozorněno na chybějící ošetření případu, kdy klient má ve vlastnictví více budov. Tento nedostatek byl následně do aplikace doplněn.

Výsledná aplikace tak splňuje definované funkční požadavky, využívá principů PWA a je plně responzivní pro využití na různých zařízeních. Aplikace je v tuto chvíli již využívána reálnými uživateli a je k dispozici na následujícím odkaze: <https://revizio-9d3ba.web.app/>.

## Použitá literatura

- [1] *Třetina bytů v Česku je napojena na dálkové vytápění, více než dvě třetiny má připojení na plyn* [online]. Český statistický úřad, 2022 [cit. 2022-11-24]. Dostupné z: <https://www.czso.cz/csu/czso/tretina-bytu-v-cesku-je-napojena-na-dalkove-vytapeni-vice-nez-dve-tretiny-ma-pripojeni-na-plyn>.
- [2] *Nařízení vlády č. 191/2022 Sb. Nařízení vlády o vyhrazených technických plynových zařízeních a požadavcích na zajištění jejich bezpečnosti*. Sbírka zákonů, 2022.
- [3] *Zákon č. 458/2000 Sb. Zákon o podmínkách podnikání a o výkonu státní správy v energetických odvětvích a o změně některých zákonů (energetický zákon)*. Sbírka zákonů, 2000.
- [4] *Nový zákon o vyhrazených technických zařízeních* [online]. Odborový svaz UNI-OS, 2022 [cit. 2022-08-31]. Dostupné z: [https://www.osunios.cz/info.php?id\\_info=2021000024&event\\_akce=info\\_detail](https://www.osunios.cz/info.php?id_info=2021000024&event_akce=info_detail).
- [5] *Zkoušení a uvádění plynovodů do provozu – TPG 704 01 – 1. část* [online]. Tzbinfo, 2013 [cit. 2022-08-11]. Dostupné z: <https://vytapeni.tzb-info.cz/vytapime-plynem/10275-zkouseni-a-uvadeni-plynovodu-do-provozu-tpg-704-01-1-cast>.
- [6] AN, Daniel. *Find out how you stack up to new industry benchmarks for mobile page speed* [online]. 2018. [cit. 2022-08-11]. Dostupné z: <https://www.thinkwithgoogle.com/marketing-strategies/app-and-mobile/mobile-page-speed-new-industry-benchmarks/>.
- [7] CÁCERES, Marcos et al. *Web Application Manifest* [online]. [cit. 2022-12-01]. Dostupné z: <https://w3c.github.io/manifest/>.
- [8] ARCHIBALD, Jake. *The service worker lifecycle* [online]. 2018. [cit. 2022-08-20]. Dostupné z: <https://web.dev/service-worker-lifecycle/>.
- [9] ATER, Tal. *Building Progressive Web Apps: Bringing the Power of Native to the Browser*. Beijing: O'Reilly, 2017. ISBN 978-1491961650.
- [10] *Nové funkce v iOS 16*. [online]. Apple, 2022 [cit. 2022-11-28]. Dostupné z: <https://www.apple.com/cz/ios/ios-16/features/>.
- [11] *Contextual Interview* [online]. Usability.gov, 2022 [cit. 2022-08-20]. Dostupné z: <https://www.usability.gov/how-to-and-tools/methods/contextual-interview.html>.

- [12] *Prototyping* [online]. Usability.gov, 2022 [cit. 2022-08-20]. Dostupné z: <https://www.usability.gov/how-to-and-tools/methods/prototyping.html>.
- [13] *React: The Virtual DOM* [online]. Codecademy, 2022 [cit. 2022-11-16]. Dostupné z: <https://www.codecademy.com/article/react-virtual-dom>.
- [14] *Understanding Redux: A tutorial with examples* [online]. LogRocket, 2022 [cit. 2022-11-17]. Dostupné z: <https://blog.logrocket.com/understanding-redux-tutorial-examples>.
- [15] *Bootstrap* [online]. [cit. 2022-11-17]. Dostupné z: <https://getbootstrap.com>.
- [16] *Material UI* [online]. [cit. 2022-11-17]. Dostupné z: <https://mui.com>.
- [17] *Sass* [online]. [cit. 2022-11-17]. Dostupné z: <https://sass-lang.com>.
- [18] QUENSE, Jason. *Yup* [online]. [cit. 2022-11-17]. Dostupné z: <https://github.com/jquense/yup>.
- [19] DILLON, Andrew. *PDF-Lib* [online]. [cit. 2022-11-17]. Dostupné z: [pdf-lib.js.org](https://pdf-lib.js.org).
- [20] *Chart.js* [online]. [cit. 2022-11-17]. Dostupné z: <https://www.chartjs.org>.
- [21] *SendGrid* [online]. [cit. 2022-11-19]. Dostupné z: <https://sendgrid.com>.
- [22] GOOGLE. *Firebase* [online]. [cit. 2022-11-19]. Dostupné z: <https://firebase.google.com/>.
- [23] *Flux* [online]. Meta, 2022 [cit. 2023-01-06]. Dostupné z: <https://facebook.github.io/flux/docs/in-depth-overview>.
- [24] LEE, James, Tao WEI a Suresh Kumar MUKHIYA. *Redux Quick Start Guide: A beginner's guide to managing app state with Redux*. Packt Publishing, 2019. ISBN 978-1789610086.
- [25] GILGUR, Anton. *React-Signature-Canvas* [online]. [cit. 2022-11-19]. Dostupné z: <https://github.com/agilgur5/react-signature-canvas>.
- [26] QUENSE, Jason. *React-Big-Calendar* [online]. [cit. 2022-11-17]. Dostupné z: <https://jquense.github.io/react-big-calendar/examples/index.html?path=/story/about-big-calendar--page>.
- [27] *React XSS Guide: Examples and Prevention* [online]. Stackhawks, 2021 [cit. 2022-11-19]. Dostupné z: <https://www.stackhawk.com/blog/react-xss-guide-examples-and-prevention/>.

## A Přílohy

## A.1 Plán testu použitelnosti

# Plán Test použitelnosti

Vypracoval  
Filip Rubeš

### O aplikaci

Název  
**Aplikace pro revizní zprávy**

Záměr testování  
**Je aplikace intuitivní? Dokáže vyhovět potřebám revizního technika?**

Rozsah testování  
**Kompletní návrh aplikace**

### Způsob testování

Místo a čas  
**24.6.2022 (19:00)**  
**Online - Figma, videohovor FaceTime**

Potřebné vybavení  
**Notebook, mobilní zařízení (telefon, tablet)**

Délka testování  
**60 minut + případná diskuze**

### Respondent

Jméno  
**Martin Chlum**

Povolání  
**Revizní technik**

Doplňující informace  
**46 let, v oboru podniká více jak 20 let**

### Scénáře

Počet  
**5**

- Název
- 1. Vytvoř provozní revizi**
  - 2. Vytvoř výchozí revizi**
  - 3. Otevři kartu klienta**
  - 4. Stáhni revizní zprávu**
  - 5. Přidej schůzku s klientem**

### Metriky

Subjektivní metrika

- Scénář  
**Jak hodnotíte celkový průběh revize? Jak hodnotíte definování závad? Co byste změnil?**
- Scénář  
**Jak hodnotíte zápis tlakové zkoušky?**
- Scénář  
**Jak hodnotíte přehlednost stránky? Chybí vám nějaké rychlé volby u dokumentů?**

Kvantitativní metrika

Všechny scénáře  
**Bude měřena míra chybovosti a úspěšnost splnění úkolů. Délka provedení úkolu nebude měřena.**



## A.2 Zápis o testu použitelnosti

# Zápis o testu použitelnosti

Vypracoval  
**Filip Rubeš**  
Respondent  
**Martin Chlum**  
24.6.2022

### Průběh testování

Scénář	Splnění úkolu	Potíže	Poznámky
1.	OK	-	Respondent nemá žádné potíže s vytvořením revize, poukazuje však na pole evidenčního čísla, které by spíše uvítal hned na začátku.
2.	OK	-	Respondent vytvořil revizní zprávu bez potíží, ocenil zjednodušení tvorby tlakové zkoušky.
3.	OK	-	Respondent našel kartu klienta bez potíží.
4.	OK	-	Respondent stáhl revizní zpráv bez problémů, využil již známé cesty.
5.	OK	-	Respondent vytvořil schůzi bez problémů.

### Zodpovězení otázek

Jak hodnotíte celkový průběh revize? Jak hodnotíte definování závad? Co byste změnil?

Průběh hodnotím kladně, napodobuje reálný průběh kontroly a je tak možné vytvářet zprávu simultánně. Osobně bych pouze změnil umístění evidenčního čísla revize. U definování závad se mi líbí, že je výchozí hodnotou "bez závad". Na vytvoření revize bych jinak nic neměnil.

Jak hodnotíte zápis tlakové zkoušky?

Zápis se mi velmi líbil, jde o velmi zjednodušenou formu zápisu, která ale plně splňuje požadavky.

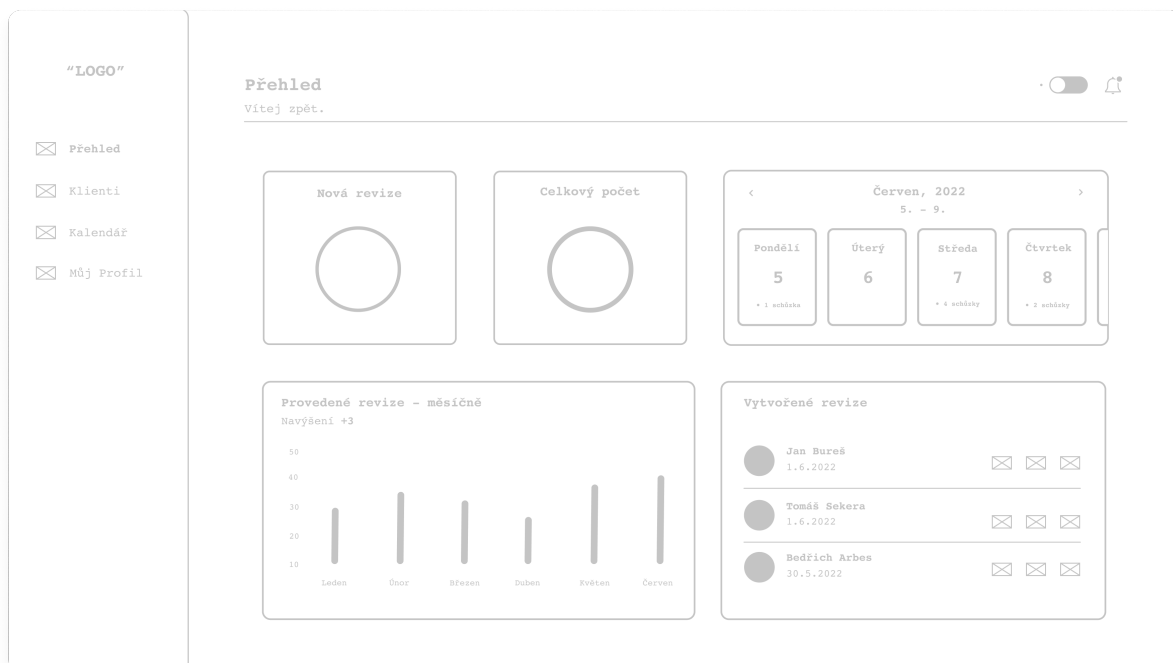
Jak hodnotíte přehlednost stránky? Chybí vám nějaké rychlé volby u dokumentů?

S přehledností jsem žádný problém neměl, vše co potřebuji na stránce nacházím. Co se týče rychlých voleb, tak mě nenapadá nic dalšího, co by mohlo reálně chybět.

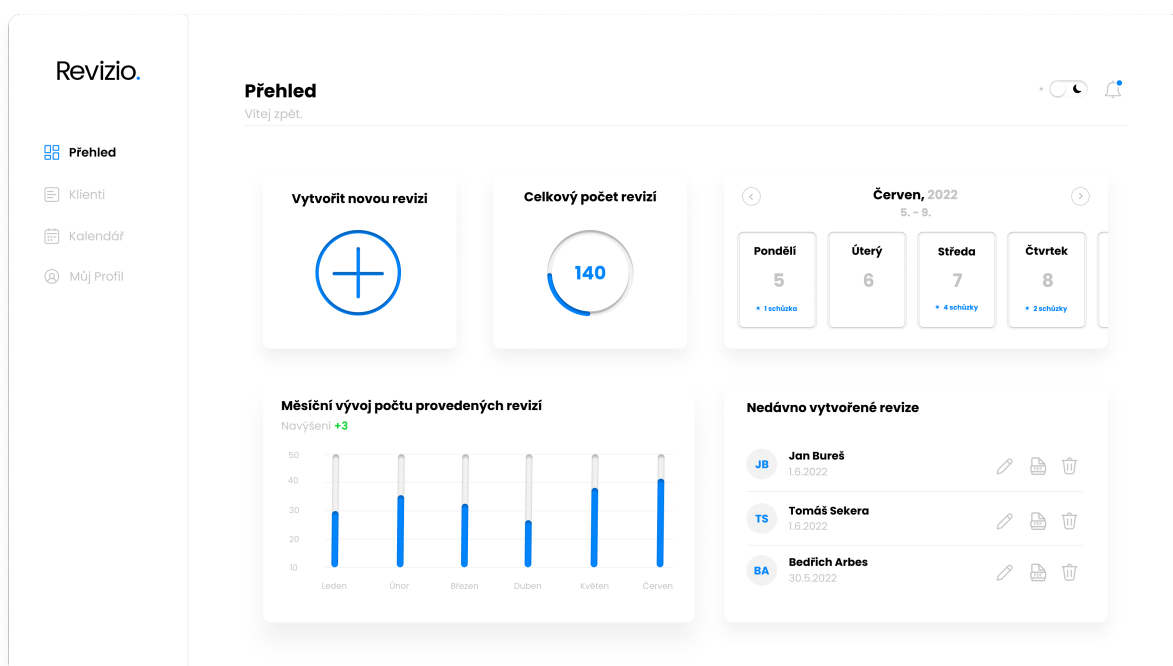
### Hodnocení a doporučení

Celkově bylo testování úspěšné. Respondent plně spolupracoval a objektivně posoudil návrh aplikace. Nalezl nekritickou chybu při návrhu, která musí být opravena (přesunutí pole pro evidenční číslo revize).

## A.3 Porovnání uživatelského rozhraní



(a) Rozhraní před



(b) Rohraní po

Obrázek A.1: Porovnání uživatelského rozhraní