



Zadání diplomové práce

Název:	Systém pro správu revizí tlakových nádob a techniky
Student:	Bc. Pavel Jordán
Vedoucí:	Ing. Jan Matoušek
Studijní program:	Informatika
Obor / specializace:	Softwarové inženýrství
Katedra:	Katedra softwarového inženýrství
Platnost zadání:	do konce letního semestru 2024/2025

Pokyny pro vypracování

Cílem této práce je navrhnout a vytvořit systém pro správu revizí tlakových nádob (a další techniky), který by nahradil starší systém používaný ve větším průmyslovém podniku.

Pokyny k vypracování:

- 1) Analyzujte problematiku informačních systémů pro správu revizí techniky.
- 2) Analyzujte současný stav používané aplikace a aktuální potřeby podniku.
- 3) Zvolte vhodnou metodiku vývoje a na základě analýzy navrhnete nové řešení, které zjednoduší práci a odstraní případné nalezené nedostatky.
- 4) Navržené řešení implementujte a řádně zdokumentujte.
- 5) Navrhnete a provedte vhodné testy aplikace a ověřte její použitelnost.
- 6) Připravte aplikaci k nasazení v podniku, včetně migrace dat ze staré aplikace.
- 7) Dosažené výsledky a získané zkušenosti shrňte.

Diplomová práce

SYSTÉM PRO SPRÁVU REVIZÍ TLAKOVÝCH NÁDOB A TECHNIKY

Bc. Pavel Jordán

Fakulta informačních technologií
Katedra softwarového inženýrství
Vedoucí: Ing. Jan Matoušek
11. ledna 2024

České vysoké učení technické v Praze
Fakulta informačních technologií

© 2024 Bc. Pavel Jordán. Všechna práva vyhrazena.

Tato práce vznikla jako školní dílo na Českém vysokém učení technickém v Praze, Fakultě informačních technologií. Práce je chráněna právními předpisy a mezinárodními úmluvami o právu autorském a právech souvisejících s právem autorským. K jejímu užití, s výjimkou bezúplatných zákonných licencí a nad rámec oprávnění uvedených v Prohlášení, je nezbytný souhlas autora.

Odkaz na tuto práci: Jordán Pavel. *Systém pro správu revizí tlakových nádob a techniky*. Diplomová práce. České vysoké učení technické v Praze, Fakulta informačních technologií, 2024.

Obsah

Poděkování	viii
Prohlášení	ix
Abstrakt	x
Seznam zkratek	xi
Úvod	1
1 Teorie	3
1.1 Úvod do domény revizí	3
1.1.1 Vyhrazená technická zařízení	3
1.1.2 Revize a pojmy s ní spojené	4
1.1.3 Proces revize	4
1.2 Metodiky vývoje	6
1.2.1 Vodopád	7
1.3 Funkční a nefunkční požadavky	8
1.3.1 Metody pro zjištění požadavků	8
1.3.2 Způsoby kategorizace	10
1.4 Uživatelské rozhraní	12
2 Rešerše	13
2.1 Proč zvolit nové řešení?	13
2.2 Řešení pro malé podniky	14
2.2.1 Fyzická správa	14
2.2.2 Lokální soubory	15
2.2.3 Online nástroje	15
2.3 Specializované systémy	16
2.3.1 Facman	16
2.3.2 Evis Pro	18
2.3.3 Online Revize	19
2.3.4 Civop	19
2.3.5 EcoSun	19
2.3.6 Eviduj si Revizi	20
2.3.7 Insio	21
2.3.8 Q-LanYs	22
2.4 Vlastní systém	22
2.4.1 Důvody pro volbu vlastního řešení v konkrétní firmě	23
2.5 Shrnutí existujících řešení	24

3	Analýza	27
3.1	Analýza současného systému	27
3.1.1	Kategorie	27
3.1.2	Aktuální struktura databáze	30
3.1.3	Doménový model	31
3.1.4	Požadavky vycházející ze současné aplikace	31
3.1.5	Současné uživatelské rozhraní	32
3.2	Analýza požadavků pro novou aplikaci	33
3.2.1	Uživatelé	33
3.2.2	Funkční požadavky	33
3.2.3	Nefunkční požadavky	37
4	Návrh	39
4.1	Uživatelské role	39
4.2	Případy užití	39
4.2.1	Správa zařízení	39
4.2.2	Tlakové lahve	40
4.2.3	Správa firem	40
4.2.4	Správa středisek	40
4.2.5	Správa nedostatků	41
4.2.6	Ostatní	41
4.2.7	Pokrytí Funkčních požadavků	42
4.3	Uživatelské rozhraní	42
4.3.1	Druhy prototypů	42
4.4	Návrh prototypu	43
4.4.1	Úvodní stránka	43
4.4.2	Stránka kategorií	43
4.4.3	Stránka souhrnu stroje	45
4.4.4	Tlakové nádoby	46
4.4.5	Nedostatky a kontakty na externí firmy	48
4.4.6	Kontakty na střediska	48
4.4.7	Log provedených změn	48
4.4.8	Kalendář	48
4.5	Databáze	49
4.6	Shrnutí	49
5	Realizace	51
5.1	Nette	51
5.1.1	Architektura Nette	51
5.1.2	Další technologie	51
5.1.3	Databáze	52
5.1.4	Použité knihovny	53
5.2	Vývoj	53
5.2.1	Vývojové prostředí	54
5.2.2	Struktura aplikace	54
5.2.3	Názvy tříd - URL	54
5.2.4	Modely	54
5.2.5	Komponenty	55
5.2.6	Presentery	56
5.2.7	Šablony	57
5.3	Shrnutí	57

6 Testování	59
6.1 Uživatelské testování	59
6.1.1 Respondenti	59
6.1.2 Počet účastníků	60
6.1.3 Scénáře	60
6.1.4 Dotazník	61
6.2 Průběh	62
6.2.1 Respondent 1	62
6.2.2 Respondent 2	63
6.2.3 Respondent 3	63
6.2.4 Respondent 4	63
6.2.5 Respondent 5	64
6.2.6 Respondent 6	64
6.2.7 Respondent 7	64
6.2.8 Respondent 8	65
6.3 Shrnutí	65
7 Nasazení	67
7.1 Migrace dat	67
7.2 Docker	67
7.3 Přidání nové kategorie	68
7.4 Další vývoj	69
8 Závěr	71
A Původní řešení - Access	73
B Návrh prototypu	77
C Testování - scénáře, dotazníky	81
Obsah přiloženého média	103

Seznam obrázků

1.1	Proces probíhající u revize	5
1.2	Vodopádový model [11]	7
2.1	Facman: ukázka revidovaného objektu [19]	17
2.2	Evis Pro: ukázka [20]	18
2.3	Eviduj si Revizi: ukázka seznamu zařízení [24]	20
2.4	INSIO: ukázka [25]	21
2.5	Q-LanYs: ukázka [26]	22
3.1	Úvodní stránka současné aplikace	28
3.2	Formulář pro správu tlakových lahví	29
3.3	Databázové tabulky staré aplikace	29
3.4	Doménový model aplikace	31
4.1	Seznamy vlastností jednotlivých kategorií	44
4.2	Návrh formuláře pro zadávání zařízení v otevřené kategorii	44
4.3	Ukázka stránky se seznamem nadcházejících revizí v otevřené kategorii	45
4.4	Souhrn revizí pro stroj ve staré aplikaci	46
4.5	Návrh stránky pro souhrn revizí na stroji.	46
4.6	Export pro tlakové lahve. Nahoře: stará aplikace. Dole: nová šablona	47
4.7	Návrh seznamu pro provozní deníky tlakových lahví.	48
5.1	Architektura MVC [30]	52
5.2	Ukázka výsledného vzhledu dlaždic z úvodní stránky aplikace	56
5.3	Stránka stroje z výsledné aplikace	56
5.4	Ukázka formuláře z výsledné aplikace	58
6.1	Procento nalezených chyb v závislosti na počtu respondentů. [40]	60
7.1	Graf plánovaných termo revizí na jednotlivá místa	69
8.1	Čas strávený prací na diplomové práci, rozdělený podle aktivity	72
A.1	Seznam prováděných revizí na vybavení	73
A.2	Databázové tabulky staré aplikace	74
A.3	Vytváření záznamu o nalezeném nedostatku	75
A.4	Formulář pro kategorii SHZ	75
A.5	Formulář pro kategorii freony	76
A.6	Formulář pro kategorii DRAEGER	76
B.1	Návrh: Úvodní stránka	77
B.2	Návrh: vyskakovací formulář pro zařízení	78
B.3	Návrh: stránka formuláře pro tlakové nádoby	79
C.1	Testování: Dotazník a instrukce	81

C.2 Testování: Scénáře	82
----------------------------------	----

Seznam tabulek

2.1 Souhrn prodáváných systémů 1	25
2.2 Souhrn prodáváných systémů 2	25
4.1 Pokrytí funkčních požadavků případy užití - část 1	42
4.2 Pokrytí funkčních požadavků případy užití - část 2	42

Seznam výpisů kódu

7.1 Ukázka docker kódu pro instalaci ODBC a SQL Server ovladačů	68
---	----

Chtěl bych poděkovat především svému vedoucímu, panu Ing. Janu Matouškovi, za vedení této práce a také za veškeré rady související s obsahem textu a návrhy do aplikace. Dále bych chtěl poděkovat Janu Fillovi, který hrál roli zadavatele v zástupu firmy, za jeho pomoc v pochopení tématu revizí a cenné rady ohledně programování v Nette. Nakonec bych rád poděkoval své rodině za jejich podporu během studia.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací. Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisů, zejména skutečnost, že České vysoké učení technické v Praze má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 citovaného zákona.

V Praze dne 11. ledna 2024

Abstrakt

Tato diplomová práce se zabývá návrhem a implementací nového systému pro správu revizí tlakových nádob a techniky, vytvářeného na míru pro firmu, kde pracuji. Začínám popisem domény revizí a potřebnou teorií. Procházím existující řešení na trhu a následně provádím kompletní analýzu funkcí systému jak současného řešení, tak i nových funkcí. Na základě této analýzy je pak vytvořen návrh aplikace s důrazem na jednoduchost a snadné použití. Další část této práce se věnuje implementaci systému, následuje jeho testování a ověření použitelnosti. Práce je zakončena nasazením systému pro použití.

Klíčová slova Nette, php, revizní systém, tlakové lahve, vývoj nové aplikace

Abstract

This diploma thesis deals with the design and implementation of a new system for the review management of pressure vessels and other technology. I create a tailor-made system for the company where I work at. I begin with a description of the revision domain and the necessary theory. I go through existing solutions on the market and then perform complete analysis of the functions of both the current solution and new functions. Based on this analysis, an application design is created with an emphasis on simplicity and ease of use. The next part of this work is devoted to the implementation of the system, followed by its testing and verification of use. The work is completed by deploying the system for use.

Keywords Nette, php, revision system, pressure vessels, new application development

Seznam zkratk

AD	Active Directory
BOZP	Bezpečnost a Ochrana Zdraví při Práci
GUI	Graphical User Interface
PO	Požární Ochrana
VTZ	Vyhrazené Technické Zařízení
ŽP	Životní Prostředí

Úvod

Provádění revizí techniky je důležitou součástí provozu firmy. Ve firmě, kde pracuji, jsou výnosy závislé na plynulém chodu strojů. Revize jsou klíčovým prvkem naší společnosti. Obzvláště, když se jedná o důležité stroje, bez kterých, v případě jejich výpadku, nelze dále vyrábět prodávané zboží. Zároveň vyhrazená technická zařízení mají přímo nařízené plnění revizí státem, ve kterém se firma nachází. Jejich nesplněním se ohrožuje celková bezpečnost, navíc jsou za nedodržení vysoké sankce.

Pro evidenci revizí, jejich plánování a ukládání záznamů o provedených revizích se používá aplikace vytvořená v Microsoft Access. Aplikace slouží pro své účely dostatečně, ale přidávání nových funkcionalit, nebo dělání změn je problematické. Z tohoto důvodu se rozhodlo o vytvoření nové aplikace, založené na jiné technologii, která umožní nové funkce a změny provádět jednoduše. Rozhodlo se, že se bude jednat o webovou aplikaci. Aplikace se zároveň v prvním nasazení nebude moc lišit od způsobu použití té stávající.

Jedním z hlavních cílů je centralizace veškerých dat do databáze. Tímto krokem zajistíme nejen jednoduchou správu dat, ale také umožníme data využívat i v dalších aplikacích. Těmi může být například vizualizace v grafech pomocí nástroje Microsoft Power BI nebo jiná analýza. Dále aplikace zjednoduší provádění změn a oprav, které bude několikanásobně rychlejší, jelikož změna v Access aplikaci vyžaduje několik paralelních kroků. To zároveň zvyšuje i flexibilitu a zjednoduší údržbu.

V dalších částech této práce postupně projdu cestou standardního vývoje aplikace. Začínám analýzou existujících řešení, současného systému a požadavků na novou aplikaci. Pokračuji návrhem prototypu a jeho ohodnocením. Následuje samotná implementace reálné aplikace, ve které popíši některé klíčové části implementace. V předposlední kapitole se věnuji uživatelskému testování aplikace a jeho výstupu, kterým jsou požadavky na změny a úpravy. Nakonec se věnuji nasazení aplikace pro reálné použití.

Výsledná aplikace není řešení, které by šlo jednoduše nasadit pro všechny podniky. Jedná se o aplikaci vytvořenou na míru pro firmu, ve které pracuji. Oblasti všech kategorií jsou uzpůsobeny firemním zvyklostem. Jiné firmy mohou mít jiné způsoby a jiné vlastnosti, které si chtějí v systému uchovat. Do určité míry aplikace umožňuje jednoduchou editaci, jelikož i pro zadavatele to bude výhodou. Například při přidávání nových kategorií, které nemusí být vedené stejným způsobem jako ty analyzované v rámci této práce.

Kapitola 1

Teorie

V této kapitole upřesním použité pojmy související s tématem revizí a vývojem aplikace spolu s rešerší spojenou s vývojovým procesem software.

1.1 Úvod do domény revizí

Představím klíčové pojmy spojené s revizemi a prostředím, ve kterém budu vyvíjet nový systém správy revizí.

Revize zařízení jsou klíčové pro zajištění bezpečného pracovního prostředí a prevenci nehod. Pravidelné revize umožňují identifikovat potenciální rizika, zhodnotit stav zařízení a zajistit jeho soulad s platnými normami a předpisy. Bezpečnostní revize jsou nezbytné pro minimalizaci rizika pracovních úrazů a zajištění souladu s právními předpisy.

Důležitým aspektem revizí je udržování přesných a kompletních záznamů. To zahrnuje dokumentaci provedených revizí, výsledky měření, a akce, které byly provedeny. Správná dokumentace je klíčová pro sledování stavu zařízení, plánování budoucích revizí a zajištění souladu s předpisy.

Bez platné revize a kompletní dokumentace není dovoleno zařízení provozovat. Tato opatření nejsou pouze formální, ale mají reálné důsledky. Provoz nezpůsobilého stroje bez platné revize nejenže nese riziko právních důsledků, ale také představuje ohrožení bezpečnosti pracovního prostředí. Takovýto provoz stroje může být postižen vysokými pokutami a sankcemi. Pravidelné revize nejen zajišťují správnou bezpečnost, ale také minimalizují riziko poruch zařízení, která snižují produktivitu práce nebo ztráty výdělků z důvodu odstavení stroje.

Z tohoto důvodu je striktní dodržování revizí a udržování kompletní dokumentace nejen zákonem vyžadováno, ale také představuje klíčový krok k zajištění bezpečného, efektivního a nepřetržitého provozu.

1.1.1 Vyhrazená technická zařízení

Jsou to zařízení se zvýšenou mírou ohrožení zdraví a bezpečnosti osob a majetku, podléhající dozoru podle zákona. Jsou ustanovena dle ustanovení § 6b odst. 1 zákona č. 174/1968 Sb., o státním odborném dozoru nad bezpečností práce. Podle stupně nebezpečí se zařazují do skupin nebo tříd. Ty určují způsob prověřování odborné způsobilosti organizací nebo fyzickými osobami.

[1]

Jedná se o zařízení:

- tlaková - parní a kapalinové kotle, tlakové nádoby,
- zdvihací - jeřáby s nosností nad 5 tun, výtahy,

- elektrická - zařízení pro výrobu, přeměnu, přenos, rozvod a odběr elektřiny,
- vyhrazená plynová - zařízení pro výrobu, úpravu, skladování a přepravu plynů.

Veškeré revize vyhrazených technických zařízení podle zákona č. 174/1968 Sb. a uvedených vyhlášek mohou provádět pouze revizní technici s platným osvědčením. Revizní technici musí mít odbornou kvalifikaci a absolvovat odborné školení. Osvědčení odborné způsobilosti je udělováno po splnění přísných kritérií, což zajišťuje, že revize jsou prováděny kvalifikovanými profesionály. [1]

1.1.2 Revize a pojmy s ní spojené

Vyhrazené technické zařízení (VTZ) - zařízení podléhající dozoru podle zákona.

Revize - posouzení bezpečnosti VTZ uváděného do provozu, nebo již provozovaného, kdy se prohlídkou nebo měřením ověřuje, zda zařízení odpovídá právním a dalším předpisům k zajištění bezpečnosti a ochrany zdraví při práci.

Revizní technik - odborně způsobilá osoba oprávněná provádět revize, která má osvědčení o odborné způsobilosti vydané podle zákona.

Montáže - činnost, při které jsou jednotlivé dílčí části spojovány v technologický celek; montáží se rozumí i demontáž a zpětná montáž.

Oprava - zásah do provozovaného VTZ, kterým se odstraňuje porucha nebo opotřebení. Může dojít k výměně, demontáži a zpětné montáži funkčních částí za účelem obnovit použitelný stav bez změny technických nebo bezpečnostních parametrů zařízení.

Údržba - činnost prováděná na VTZ nebo jeho částech za účelem zajištění bezpečného a provozuschopného stavu, bez opravy nebo montáže.

Průvodní dokumentace - soubor dokumentů dodaných výrobcem nebo dodavatelem VTZ.

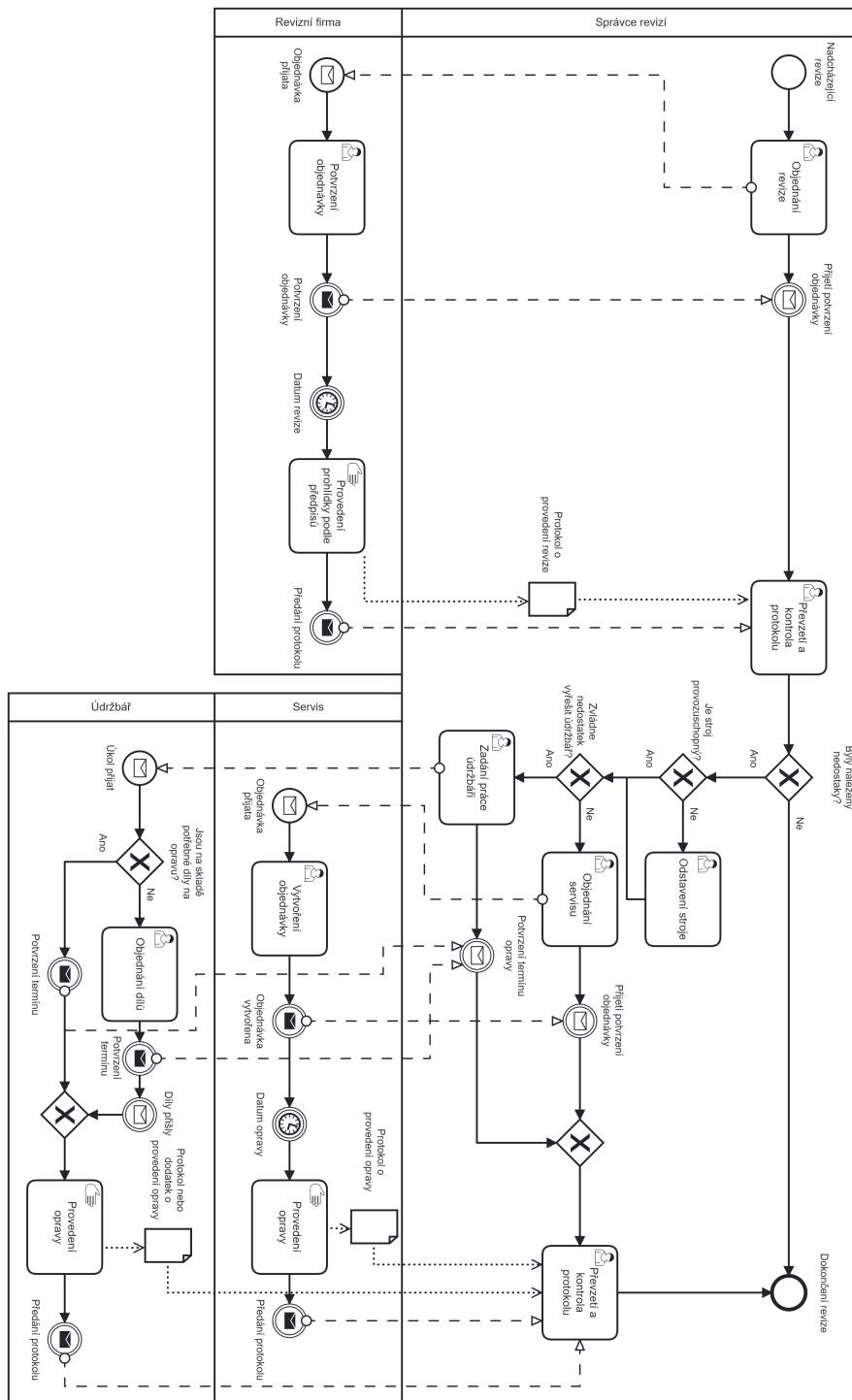
Provozní dokumentace - soubor dokumentů obsahující záznamy o kontrolách, zkouškách a revizích, místní provozní řád, provozní deník, doklady o kvalifikaci obsluhy, záznamy o opravách a údržbě, vznikající při provozu daného VTZ v rozsahu s předpisy.

Rekonstrukce - nahrazení stávající nevyhovující části provozovaného VTZ novou nebo modernější částí zařízení, kdy dojde ke změně technických nebo bezpečnostních parametrů zařízení [2]

1.1.3 Proces revize

Proces revize zahrnuje pravidelné a systematické posuzování stavu vyhrazených technických zařízení. Frekvence revizí se odvíjí od typu zařízení a příslušných předpisů. Kroky revize zahrnují důkladnou prohlídku, měření a testování klíčových parametrů zařízení, a následnou aktualizaci dokumentace.

V diagramu 1.1 můžeme vidět jakým způsobem revize probíhá v reálném světě. Stroj může být i s nalezenými nedostatky provozuschopný, záleží na závažnosti závady. Je klíčové, aby záznamy o odstranění problému obsahovaly popis opravy a byly pečlivě podepsané, což posiluje transparentnost a odpovědnost v rámci revizního procesu.



■ Obrázek 1.1 Proces probíhající u revize

1.2 Metodiky vývoje

Metodologie vývoje softwaru je soubor pravidel, postupů a nástrojů používaných pro návrh, plánování a řízení vývoje softwaru. Metodika vývoje je jeden konkrétní používaný postup, nebo se metodikou rozumí použití určitého frameworku při vývoji. První formální metodický framework byl vytvořen již v roce 1960 pod jménem Systems Development Life Cycle (SDLC). Jeho hlavní myšlenkou bylo „nastavit vývoj informačních systémů velmi účelným, strukturovaným a metodickým způsobem, což vyžadovalo, aby všechny fáze vývoje od nápadu až po dodání finálního systému byly provedeny postupně a vždy v nezměněné formě“.[3]

V průběhu let bylo vyvinuté velké množství takových frameworků, kdy každý měl své silné i slabé stránky. Každý framework je vhodný pro jiné typy projektů s ohledem na jejich potřeby. Metodické frameworky jsou často rozvíjené, podporované a propagované nějakou organizací a jsou definované pomocí formální dokumentace. Časem vzniklo mnoho vývojových postupů, které metodické frameworky předepisují.

Nejvýznamnější přístupy vývoje:

- Vodopádový přístup - lineární typ.
- Prototypový přístup - iterativní typ.
- Inkrementální přístup - kombinace lineárního a iterativního typu.
- Spirální přístup - kombinace lineárního a iterativního typu.
- RAD (Rapid Application Development) - iterativní typ.

Nejznámější metodiky:

Vodopádový model (Waterfall) Vodopádová metodika je tradiční a sekvenční přístup k vývoji software, kde každá fáze následuje po předchozí. Požadavky jsou definovány na začátku a každá fáze musí být dokončena před tím, než začne další. Tento model je snadno pochopitelný a má jasně definované milníky, ale může být méně flexibilní pro změny požadavků. [4]

Agilní metodiky Metodiky, jako je Scrum, Extreme Programming (XP) atd., se zaměřují na iterativní a inkrementální vývoj software. Pracuje se s krátkými vývojovými cykly nazývanými iterace nebo sprinty, kde se každý cyklus soustředí na dodání funkčního a ověřitelného kusu softwaru. Agilní metodiky jsou vhodné pro projekty, které vyžadují vyšší flexibilitu a schopnost reagovat na změny požadavků. [5]

V-model V-model je variantou vodopádového modelu, který klade větší důraz na testování. Každá fáze vývoje má příslušnou testovací fázi, což zvyšuje spolehlivost softwaru. Může být vhodný pro projekty s vysokými nároky na kvalitu.[6]

Spirálový model Spirálový model kombinuje prvky vodopádového modelu s agilními aspekty. Projekt prochází iterativními cykly, přičemž každý cyklus zahrnuje plánování, analýzu rizik, implementaci a hodnocení. Tento přístup umožňuje pružnost při řešení změn a rizik. [7]

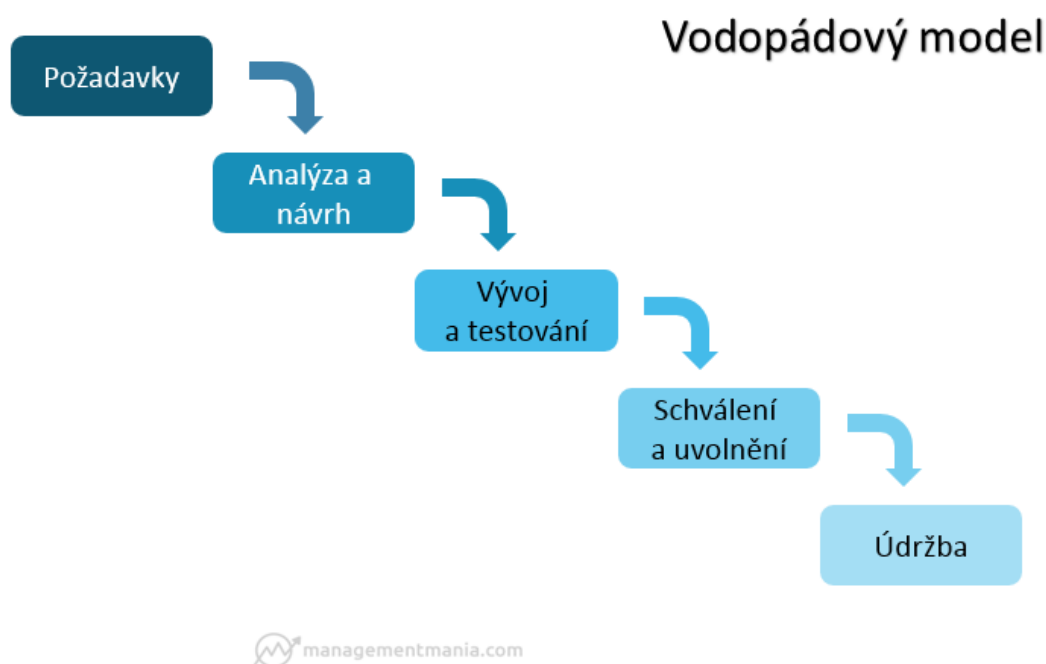
Lean software development Metodika vychází z principů lean managementu a zaměřuje se na minimalizaci plýtvání a zvyšování hodnoty pro zákazníka. Klade důraz na rychlou dodávku funkčního softwaru a zapojení zákazníka do celého vývojového procesu. [8]

RUP (Rational Unified Process): RUP je strukturovaná metodika vyvinutá společností IBM, která klade důraz na iterativní a inkrementální vývoj software. Má dobře definované role a pracovní postupy a věnuje se důkladnému řízení rizik. [9]

DevOps DevOps je metodika, která spojuje vývoj (Development) a provoz (Operations) softwaru. Cílem je dosáhnout lepší spolupráce mezi vývojáři a IT provozem, čímž se snižuje čas a riziko spojené se zaváděním nových verzí softwaru. [10]

1.2.1 Vodopád

Přestože se v dnešním prostředí častěji upřednostňují agilní metodiky, jako je Scrum, jsem zvolil metodiku Vodopád. Je to stabilní a osvědčená metodika a navíc sedí pro můj případ vývoje. Vodopádový model nabízí pevnou strukturu, která je ideální pro projekty s jasně definovanými a neměnnými požadavky. Má jasně definované fáze, které mi usnadní integrovat vývoj s procesem psaní diplomové práce.



■ **Obrázek 1.2** Vodopádový model [11]

Na vývoji se budu podílet sám, mám jasně definovaný rozsah práce, kterou budu vykonávat. Vývoj bude postupovat postupně podle jednotlivých fází. Od detailní analýzy, přes návrh, implementaci, testování, až po výsledné nasazení. Navíc při vývoji píši diplomovou práci, která slouží mimo jiné i jako důkladná dokumentace spojená s vývojem.

Fáze vodopádu:

Analýza V této fázi se shromažďují a definují požadavky na projekt od zákazníka nebo stakeholderů. Cílem je jasně stanovit, co má být vytvořeno. Výsledkem této fáze je dokumentace požadavků, která slouží jako základ pro celý projekt.

Návrh Na základě definovaných požadavků se vytváří podrobný návrh systému. Tato fáze zahrnuje navržení architektury, struktury dat, uživatelského rozhraní a dalších technických aspektů. Výsledkem této fáze je návrhová dokumentace.

Implementace V této fázi se provádí skutečný vývoj softwaru na základě návrhu. Programátoři píšou kód a integrují jednotlivé komponenty. Výsledkem této fáze je funkční software, který odpovídá specifikacím z návrhu.

Testování Po implementaci následuje fáze testování, kde se provádí testování vytvořeného softwaru, aby se zjistilo, zda splňuje stanovené požadavky a pracuje správně. Tato fáze zahrnuje testování funkcionality, výkonu, bezpečnosti a dalších aspektů. Po úspěšném testování interně může následovat testování zákazníkem nebo uživateli. Toto testování se může brát jako samostatná fáze, která se nazývá odběrné testování a slouží k ověření, zda výsledný software splňuje očekávání uživatelů.

Zavedení a údržba Po schválení všech testovacích fází se software zavádí do ostrého provozu. Poté následuje fáze údržby, kde se řeší chyby, aktualizace a případné rozšíření softwaru. Při dalším rozšiřování se může zvolit jiná metodika, nebo se proces vodopádu opakuje znovu pro nové požadavky. [4]

1.3 Funkční a nefunkční požadavky

Požadavky z pohledu softwarového inženýrství jsou specifikací toho, co by mělo být implementováno. Popisují veškeré chování systému nebo jeho části. Měly by jasně a jednoznačně definovat konkrétní omezení, hranice, kde by měl softwarový proces skončit. Nestanovení jasné hranice může způsobit problémy se zákazníkem, který nebude chtít platit za něco, co dle něj již bylo zapláceno. Z tohoto důvodu se doporučuje jasně a jednoznačně vymezit to, co se udělat má a co se implementovat nebude.

Funkční požadavek by měl obsahovat:

- Název,
- Zkratku - ta usnadňuje odkazování na daný požadavek,
- Popis - nejdůležitější část požadavku (měl by být jednoznačný s konkrétní činností, mít vymezené podmínky a omezení, které se na požadavek kladou).
- Prioritu,
- Složitost. [12]

Softwarové požadavky můžeme dělit dle následujících typů:

- Funkční - požadavky popisující funkce systému a jeho chování.
- Nefunkční - požadavky zabývající se obecnými vlastnostmi systému.
Typicky: *přenositelnost, bezpečnost, udržovatelnost, spolehlivost, výkon, flexibilita.*
- Požadavky na doménu - liší se v závislosti na konkrétní doméně. [13]

1.3.1 Metody pro zjištění požadavků

Při sepisování funkčních a nefunkčních požadavků pro vývoj software existuje několik metodik a technik, které lze použít pro zjištění požadavků. Výběr správných metodik závisí na specifickém projektu, velikosti týmu a preferencích. Existuje i možnost použít více metod nebo je kombinovat. Ve výsledku je důležité zajistit, aby požadavky byly přesné, jednoznačné a dohodnuté všemi členy týmu a zákazníkem.

Často používané metody:

Brainstorming Tato technika spočívá v organizované skupinové diskuzi, kdy se členové týmu podílejí na generování nápadů a požadavků pro projekt. Brainstorming pomáhá shromáždit různorodé perspektivy a zkušenosti. Často se používá při zahájení projektu nebo při identifikaci nových funkcionalit v rané fázi vývoje.

- Fáze generování: Každý člen týmu přináší své nápady.
- Fáze diskuse: Společné hodnocení a diskuse nad nápady.
- Výběr klíčových požadavků: Identifikace klíčových prvků.

Rozhovor Setkání s uživateli a zákazníky nebo dalšími stranami umožní získat přímou zpětnou vazbu a lépe pochopit jejich požadavky a potřeby. Je vhodný pro důkladné porozumění potřebám konkrétních uživatelů nebo zainteresovaných stran.

- Příprava: Definice cílů a otázek.
- Provedení: Setkání s uživateli.
- Analýza: Vyhodnocení získaných informací.

Prototypování Vytvoření jednoduchého prototypu nebo náčrtku aplikace může pomoci lépe pochopit funkční požadavky a zjistit, jak bude produkt fungovat v reálném světě. Používá se při potřebě vizualizace a lepšího pochopení, jak by měly jednotlivé funkce vypadat.

- Navrhování prototypu: Vytvoření zjednodušené verze produktu.
- Testování: Získání zpětné vazby od uživatelů.
- Iterace: Opravy a vylepšení prototypu.

Use Case (Případy užití) Use case je způsob modelování scénářů, ve kterých bude software používán. Pomáhá identifikovat jednotlivé akce, které musí software provádět, a určuje, jakým způsobem má interagovat s uživateli a systémem. Velmi často využívaný pro identifikaci konkrétních akcí a interakcí uživatele se systémem.

- Identifikace aktérů: Kdo interaguje se systémem.
- Popis scénářů: Jak aktéři používají systém.
- Identifikace požadavků: Získání klíčových požadavků.

Uživatelské příběhy Uživatelské příběhy jsou krátké, jednoduché a snadno pochopitelné výpisy požadavků z pohledu uživatelů. Každý uživatelský příběh popisuje specifickou funkcionalitu nebo vlastnost produktu. Je vhodný pro jednoduché a srozumitelné popisy požadavků z pohledu uživatelů.

- Identifikace uživatelských rolí: Kdo jsou klíčoví uživatelé.
- Popis funkcionalit: Jednotlivé vlastnosti z pohledu uživatelů.
- Prioritizace: Ohodnocení důležitosti každého příběhu.

Scrum board V agilních metodikách lze použít Scrum board k vizualizaci a správě požadavků. Jedná se o několik vizuálních tabulí, na kterých lze sledovat například co je potřeba udělat, co se aktuálně dělá a co už je hotovo. Používá se během agilního vývoje.

- Vytvoření boardu: Rozdělení na kategorie (To Do, In Progress, Done).
- Přidělování úkolů: Každý týmový člen si bere úkoly.
- Sledování pokroku: Vizuální sledování stavu úkolů.

SWOT analýza Analýza silných stránek, slabých stránek, příležitostí a hrozeb. Používá se k hodnocení výhod a nevýhod, které mohou ovlivnit projekt. Pomáhá identifikovat klíčové oblasti, které je třeba zvážit při stanovování požadavků.

- Identifikace silných a slabých stránek: Vnitřní faktory.
- Hodnocení příležitostí a hrozeb: Externí faktory.
- Stanovení priorit: Určení klíčových oblastí pro zvážení požadavků.

Benchmarking Benchmarking porovnává výkon a funkce produktu s konkurenčními nebo srovnatelnými produkty. Tato technika umožňuje identifikovat silné a slabé stránky projektu a inspiruje se osvědčenými postupy odjinud.

- Identifikace konkurentů: Určení srovnatelných produktů nebo služeb.
- Sběr dat: Shromáždění informací o výkonnosti konkurence.
- Analýza výsledků: Porovnání výkonu a funkcí s výsledky benchmarkingu.

[14], [15]

1.3.2 Způsoby kategorizace

Způsobů, jak kategorizovat, určit priority, složitost a další metriky jednotlivých požadavků je také několik. Často se metody liší v pojmenování kategorií a podrobnostech, ale v zásadě slouží k tomu, aby pomohly identifikovat a rozdělit požadavky podle jejich důležitosti. Každá z metod má své výhody a může být použita v závislosti na specifických potřebách projektu a preferencích týmu. Důležité je, aby byla zvolená metoda transparentní a všichni zainteresovaní byli zapojeni do procesu prioritizace požadavků. Metody je opět možné kombinovat nebo používat pro různé fáze projektu.

1.3.2.1 MoSCoW

Pomáhá určit, jaké požadavky nebo úkoly jsou nejvyššího významu a musí být splněny jako první. Často se používá v agilních metodologiích. Používá pojmy „Must have“, „Should have“, „Could have“, „Won't have (this time)“. [16]

Příklad výstupu metody MoSCoW:

- **Must have** - Implementace bezpečnostního systému pro ukládání citlivých informací.
- **Should have** - Možnost exportu dat do formátu CSV pro snadné sdílení.
- **Could have** - Integrace s externími službami sociálních médií pro sdílení úspěchů uživatelů.
- **Won't have (this time)** - Implementace rozsáhlého systému doporučení produktů, což by bylo mimo rozsah aktuálního projektu.

1.3.2.2 Kano model

Kano model rozděluje požadavky do několika kategorií podle toho, jak ovlivňují spokojenost uživatelů. Kategorie zahrnují základní, očekávané, překvapivé, neutrální a negativní požadavky. Tato metoda umožňuje lépe porozumět vztahu mezi funkcemi a spokojeností uživatelů. [16]

Příkladem může být:

- **Základní požadavek** - Funkční a spolehlivý proces registrace uživatelů.
- **Očekávaný požadavek** - Rychlá a responsivní uživatelská rozhraní pro efektivní používání aplikace.

- **Překvapivý požadavek** - Personalizované zprávy o novinkách a aktualizacích.
- **Neutrální požadavek** - Implementace funkcí, které jsou průměrně hodnoceny a mají střední dopad na spokojenost uživatele.
- **Negativní požadavek** - Přidání složitých funkcí, které by zpomalily celkový výkon aplikace.

1.3.2.3 100-point metoda

Tato metoda přiděluje každému požadavku určitý počet bodů na základě jeho důležitosti. Projektový tým následně může věnovat pozornost požadavkům s nejvyšším počtem bodů. [16]

Příklad rozdělení:

- **Vysoká priorita (80-100 bodů)** - Vytvoření bezpečného přihlašování s ověřením dvou faktorů.
- **Střední priorita (50-79 bodů)** - Zavedení uživatelských rolí pro rozlišení oprávnění.
- **Nízká priorita (20-49 bodů)** - Přidání nástroje pro personalizaci uživatelského rozhraní.
- **Extrémně nízká priorita (0-19 bodů)** - Implementace virtuální reality pro rozšíření zážitku uživatele.

1.3.2.4 Value vs. Complexity matrix

Tato technika rozděluje požadavky do matice. Na jedné ose je hodnota pro uživatele (uživatelská hodnota) a na druhé ose složitost nebo náklady pro vývoj (technická složitost). To umožňuje identifikovat požadavky, které mají vysokou hodnotu pro uživatele a jsou relativně snadné na implementaci. [16]

Příkladem lze uvést tyto případy:

- **Vysoká hodnota, Nízká složitost** - Implementace jednoduchého uživatelského rozhraní pro rychlý přístup k často používaným funkcím.
- **Vysoká hodnota, Vysoká složitost** - Integrovaný systém umožňující automatické generování podrobných analytických zpráv na základě složitých datových analýz.
- **Nízká hodnota, Nízká složitost** - Přidání jednoduchého tlačítka pro změnu barvy pozadí aplikace, což by zvýšilo estetickou přitažlivost aplikace.
- **Nízká hodnota, Vysoká složitost** - Implementace složité a pokročilé vizualizace dat, která by byla málo využívána koncovými uživateli.

1.3.2.5 Binary prioritization

Tato metoda zahrnuje rozdělení požadavků do dvou kategorií - "musí být splněno" a "může být odloženo". To umožňuje zaměřit se na základní funkcionality a zároveň zvážit další funkce pro budoucí iterace. [16]

Jednoduchý příklad:

- **Musí být splněno** - Zabezpečení citlivých dat a údajů.
- **Může být odloženo** - Implementace tematických emotikonů pro chatování mezi uživateli.

1.4 Uživatelské rozhraní

Uživatelské rozhraní je souhrn způsobů, jakými uživatel ovlivňuje chování systému. Hlavními aspekty pro hodnocení uživatelského rozhraní je jednoduchost a přehlednost.

Je to prostředník, který zprostředkovává komunikaci mezi člověkem a počítačem nebo informačním systémem. Jeho vzhled, funkčnost, přehlednost nebo chaos ovlivňují, zda se uživatel příště vrátí, nebo ne. Návrh, struktura a následná realizace jsou důležitým faktorem vzniku jakéhokoli informačního systému.

Uživatelské rozhraní tvoří následující 3 složky:

- Fyzická složka: vstupní, výstupní zařízení a nástroje zpětné vazby.
- Konceptuální složka: mechanismy použity při komunikaci mezi člověkem a počítačem. Např. dotazovací jazyky.
- Perceptuální složka: vnímání objektů na obrazovce, jejich uspořádání a použití barev. [17]

Dobře navržené uživatelské rozhraní by mělo být:

- Jednoznačné,
- stručné,
- responzivní,
- konzistentní,
- efektivní,
- pohodlné,
- lehce napravitelné. [18]

Jednou z hlavních vlastností uživatelského rozhraní je použitelnost. Použitelnost souvisí s ovládáním, srozumitelností a přehledností. Zda se uživatel orientuje snadno, rychle a bez námahy.

Některé z principů použitelnosti:

- Pravidlo 10 minut - uživatel by měl být schopen osvojit si základy práce se systémem do 10minut.
- Pravidlo 3 kliknutí - pokud se nachází hledaná informace na stránce, měl by jí uživatel najít do 3 kliknutí.
- Pravidlo 2 sekund - doba odezvy systému by měla být v rozmezí 2 sekund. [17]

Kapitola 2

Rešerše

Před tvorbou návrhů nového uživatelského rozhraní je vhodné analyzovat existující řešení a prodávané systémy případné konkurence. To pomůže k rozšíření perspektivy nad danou doménou problému. Je možné se inspirovat různými prvky existujících systémů a zvážit jejich přidání do vyvíjeného systému, nebo naopak zjistit proč dané prvky vynechat. Například kalendář nadcházejících revizí by se mohl zdát jako potřebná a užitečná funkce. Po důkladnější analýze přímo na funkčních procesech ve firmě se ale ukázal jako zbytečný a nepotřebný.

Existuje několik způsobů, kterými lze vést si informace o zařízeních a jejich revizích.

- fyzická správa,
- lokální soubory (Excel),
- online nástroje,
- specializované systémy,
- vlastní systém.

Každý způsob řízení má své výhody a nevýhody. Výběr nejlepšího řešení záleží na velikosti podniku, počtu zařízení vedených v podniku a frekvenci používání k různým účelům (vyhledávání v historii, ukládání dokumentů, plánování revizí). Pro podnik jsou navíc důležité náklady spojené s tvorbou nebo zakoupením řešení. Ty mohou zahrnovat například licence, školení zaměstnanců a další. Dále je nutné zohlednit náklady na pravidelnou údržbu a aktualizace systému.

2.1 Proč zvolit nové řešení?

Důvodů pro přechod na nové řešení je několik. V lepším případě se pro přechod rozhodne včas, jelikož je plánované rozšiřování firmy, nakupování nových zařízení atd. a už se ví dopředu, že bude potřeba provádět revize, ukládat dokumenty o provedení, plánovat revize a další věci kolem revizí. Zavedení nového a pořádného řešení, které zvládne vše co je a bude potřeba při rozvoji, je pak důležité. Z dlouhodobého hlediska se sníží náklady, které by byly potřeba při přesunu na jiné řešení nebo zvyšující se náklady spojené s provozem pomalu stárnoucího stávajícího řešení nebo náklady na jeho vývoj.

Tím se přesouváme k dalšímu důvodu pro přesun na nové řešení, kterým je zastaralý systém co nezvládá vše to, co je od něj vyžadováno, nebo není schopen zpracovat stále se zvyšující objem dat a náklady na jeho správu a vývoj přesáhly přijatelnou mez. V takovém případě je možné detailní analýzou zjistit, že přechod na nové řešení bude pro firmu výhodnější, i přes všechny náklady spojené s přesunem dat.

Faktorů pro volbu řešení je také několik. Bude záležet například na rozsahu firmy a počtu revidovaných zařízení. Nové nebo malé firmy mají často flexibilitu v zavedení nových systémů bez komplexních migračních procesů. Nemají moc dat, která jsou potřeba přesunout do nového systému a zaměstnanci si zatím pořádně nezvykli používat řešení používané před změnou. Přejít na moderní systém může tyto firmy vybavit efektivnějším nástrojem pro správu revizí a usnadnit dodržování předpisů.

U velkých firem je to naopak často dost obtížné, někdy i nemožné. Dat potřebných převést do nového systému je moc, mnohdy je potřeba více lidí jen k tomu, aby se zjistilo, kde všechny data jsou a co vše je potřebné převést. Samotný přesun dat ani nemusí být jednoduše možný, například z důvodu zastaralých technologií, které neumožňují správný export dat. V takovém případě pak nároky jenom na samotný přesun dat několikanásobně vzrostou. Další problém, který je potřeba zvážit, je ten, že si zaměstnanci už za dlouhou dobu používání na stávající systém zvykli, a bude těžké si zvyknout na nový. Nové řešení tedy musí být atraktivní pro uživatele, musí jim ulehčovat práci natolik, aby se chtěli v systému naučit pracovat.

Hlavním faktorem, který bude ve volbě dominovat velkou většinou jsou určitě náklady spojené s případným přechodem na nové řešení, nebo správou toho aktuálního. Dále to, jakým způsobem dané řešení plní všechny naše požadavky, jestli bude potřeba přizpůsobit procesy ve firmě systému, nebo zaplatit víc a systém upravit pro zaběhlé procesy.

2.2 Řešení pro malé podniky

Pro malé podniky představují základní řešení, jako je fyzická správa dokumentů, ukládání do lokálních souborů nebo využití online nástrojů, nízko-nákladovou možnost řízení revizí. Tyto přístupy poskytují jednoduchý a přehledný systém, což je ideální pro společnosti s omezenými zdroji. Často malé rozvíjející se firmy začínají těmito řešeními a v případě úspěchů a růstu firmy se setkají s výzvami spojenými s přechodem na pokročilejší systémy správy revizí.

Při zohlednění specifických potřeb malých podniků jsou tato jednoduchá řešení snadno implementovatelná a přizpůsobí se organizační struktuře menších firem. Výhody jako bezplatné nebo nákladově efektivní řešení, zabezpečení přístupu a jednoduchá údržba činí tato řešení ideálními pro malé firmy, které hledají spolehlivý způsob správy revizí ve svých raných fázích rozvoje. Pokud je firma ambiciózní a předpovídá velký růst, doporučil bych rovnou používat nástroje přímo na revize specializované. Může se totiž stát, že výdaje spojené s přechodem na jiný systém převyšují částku, kterou by firma zaplatila, kdyby systém platila rovnou od začátku pro omezený počet zařízení.

2.2.1 Fyzická správa

Základní způsob, kterým lze řešit správu revizí je samozřejmě samotné vedení důležitých dokumentů v papírové podobě někde ve skříni v šanonu. Toto řešení je téměř bez nákladové a pro malé podniky může být vyhovující. Avšak existují určité nevýhody. Například zabezpečení dokumentů, ochrana proti ztrátě a vyhledávání. Nemusí se jednat o žádnou katastrofu, stačí nechtěné vylití čaje na dokument a jeho nenávratné zničení nebo založení na špatné místo. Jeho hledání je pak místo otázky 5 minut hledání například na hodinu.

Výhody

- bezplatné řešení,
- není potřeba žádná dodatečná znalost.

Nevýhody

- vhodné pouze pro velmi omezený počet zařízení a revizí,

- zabezpečení a rychlost přístupu k informacím,
- není možnost přehledného zobrazení aktuálního stavu,
- zdoluhavé hledání v historii,
- potřeba pamatovat si termíny nadcházejících revizí.

2.2.2 Lokální soubory

Dalším způsobem, který je také téměř bezplatný, je možnost ukládat si důležité dokumenty někde na disku v počítači a pro přehlednost si vytvořit soubor, který uchovává informace o místě uložení jednotlivých dokumentů vázajících se k danému stroji, který zároveň bude zobrazovat aktuální revizi a plánovaný datum další revize. K tomu lze použít například Microsoft Excel, který umožňuje takovýto soubor jednoduše připravit spolu i s vyhledáváním, řazením a dalšími operacemi pro zvýšení přehlednosti. Například barevného zvýraznění záznamů podle kritérií, jako čas do další revize, označení revizí, kde byli nalezeny nedostatky atd.

Přístup do složky a k daným souborům lze nastavit, čímž se zamezí neoprávněným přístupům. Stále existuje možnost ztráty dat poruchou hardwaru nebo živelní pohromou, z tohoto důvodu je potřeba provádět zálohy.

Výhody

- téměř bezplatné řešení,
- stačí základní znalost práce s počítačem,
- zabezpečení přístupu heslem,
- zvýšená přehlednost.

Nevýhody

- vhodné pouze pro malý podnik,
- potřeba provádět pravidelné zálohy na jiné zařízení,
- ukládání dlouhodobé historie není ideální.

2.2.3 Online nástroje

Online služby jako Microsoft Sharepoint a Excel Online nebo Google disk v kombinaci s Google Sheets atd. Data jsou zálohovaná přímo poskytovatelem služby, tedy data máme zabezpečená a není potřeba řešit scénáře, ve kterých ztratíme data selháním hardwarových prostředků. Oproti ukládání lokálních souborů je potřeba zaplatit si cloudové úložiště, pokud nestačí bezplatná velikost. Zaměstnanci musí umět používat dané online nástroje, ale výhodou je, že může více zaměstnanců upravovat soubor zároveň.

Výhody

- potřeba jen základní znalosti práce s kancelářskými nástroji,
- automatické zálohy,
- zabezpečení přístupu heslem,
- možnost přístupu více uživatelů zároveň.

Nevýhody

- vhodné pouze pro malý podnik,
- placení za cloudové úložiště,
- ukládání dlouhodobé historie není ideální.

2.3 Specializované systémy

Specializované systémy, vytvořené přímo pro správu revizí techniky jsou určitě ve velké většině případů výborným řešením. Takové systémy jsou často především zaměřeny na velké firmy, obsahují tedy velkou většinu potřebných nástrojů a funkcionalit. V některých firmách se může zdát, že funkcionalita je příliš mnoho, a část z nich se vůbec nevyužije, v jiných firmách však může některá funkcionalita chybět. Zakoupené specializované systémy jsou často drahé, jak na pořizování, tak na údržbu, ale není potřeba mít vlastní zaměstnance, kteří by se o správu systému starali.

Nasazení a používání specializovaného systému je nejjednodušší hned od začátku založení firmy, nebo jeho zavedení velmi brzo, než se firma rozroste. Uživatelé si takto rychle zvyknou na systém a nevzniknou velké náklady, především časové, potřebné pro zavedení všech informací ze starého řešení do nového systému.

Výhody

- přehlednost,
- veškerá funkcionalita je připravená,
- častá podpora aktualizací systému z důvodu státních nařízení.

Nevýhody

- drahé,
- potřeba naučit se používat nový systém,
- nemožnost přidání funkcionality, nebo drahé přidání funkcionality,
- velmi drahý přesun na nový systém v rozsáhlé firmě.

Aktuálně je na trhu několik dostupných revizních systémů. Jedná se o placené systémy, kde většina z nich nabízí vyzkoušení na 30 dní zdarma. Následně se ceníky pohybují od 360 Kč až po 3 000 Kč měsíčně, podle různých parametrů, jako jsou počet uživatelů, počet evidencí, velikost úložiště, počet záznamů, priorita podpory atd. Existují také firmy, které nabízí řešení, kdy přímo provádí revize techniky a zajišťují jejich správu.

V následujících sekcích popisují některé dostupné systémy. Jejich výhody a nevýhody vypisují na základě popisu služby přímo, z jejich webových stránek a z demo verzí, které jsem vyzkoušel.

2.3.1 Facman

Dostupné na webových stránkách Facman (<https://www.facman.cz/about.php>). Umožňuje vyzkoušení si demo verze zdarma. Odkaz na demo verzi je na stránce s ceníkem, hned pod cenami jednotlivých variant.

Facman nabízí tři varianty aplikace: Basic, Standard a Pro. Každá varianta umožňuje spravovat různé množství revizí, má určité kvantum dat pro ukládání a jiný počet uživatelů. Zákazníci platí pouze za to, co skutečně využijí. Možnost objednat si aplikaci na půl roku nebo na rok dává zákazníkům flexibilitu. [19]

Výhody

- mobilní i stolní zařízení,
- cloudové řešení,
- vlastní subdoména pro přístup,
- možnost barevného odlišení,
- automatické upomínky.

Nevýhody

- není možnost řadit podle jednotlivých kritérií,
- pevná struktura, pouze možnost skupin a podskupin,
- pouze výčet podporovaných zařízení, zbytek jako „ostatní“,
- každý odkaz na vlastní stránce, i v případech kdy to nedává smysl (příložený dokumentu),
- 450 Kč/měsíc až 2 000 Kč/měsíc,
- nad 3000 revizí je potřeba individuální přístup a cena.

The screenshot displays the 'Facman' web application interface. At the top, there is a navigation bar with 'Úvod', 'Administrace', and 'Přehled revizí'. The main content area is divided into several sections:

- Objekt:** A dropdown menu shows '237063 - Corso' with a 'Přejít' button.
- Detail objektu:** A table of object details:

Označení:	237063
Jméno:	Corso
Zakázka:	Žádná
Klient:	Corso jabadabada jaja
Popis:	Celá správa objektu
Ulice:	Na Břehu 267
Město:	Praha 9
PSČ:	19000
Kontakt:	
- telefon:	
- e-mail:	
- Mapa:** A Google Maps view showing the location of the object in Jeseník, with various landmarks like 'Vlastivědné Muzeum Jesenícká' and 'Kebab house Jeseník' visible.
- Revize:** A table listing revisions:

Název	Platnost do	Dodavatel	Počet	Typ	Výrobní číslo	Poznámka
VZT (2m)	01. 03. 2021	První plynařský servis s.r.o.	5			
EPS - funkční zkouška (6m)	02. 08. 2022	EZH,a.s.	5			
- Přílohy:** A section for attachments with a 'Nový' button.

■ Obrázek 2.1 Facman: ukázka revidovaného objektu [19]

2.3.2 Evis Pro

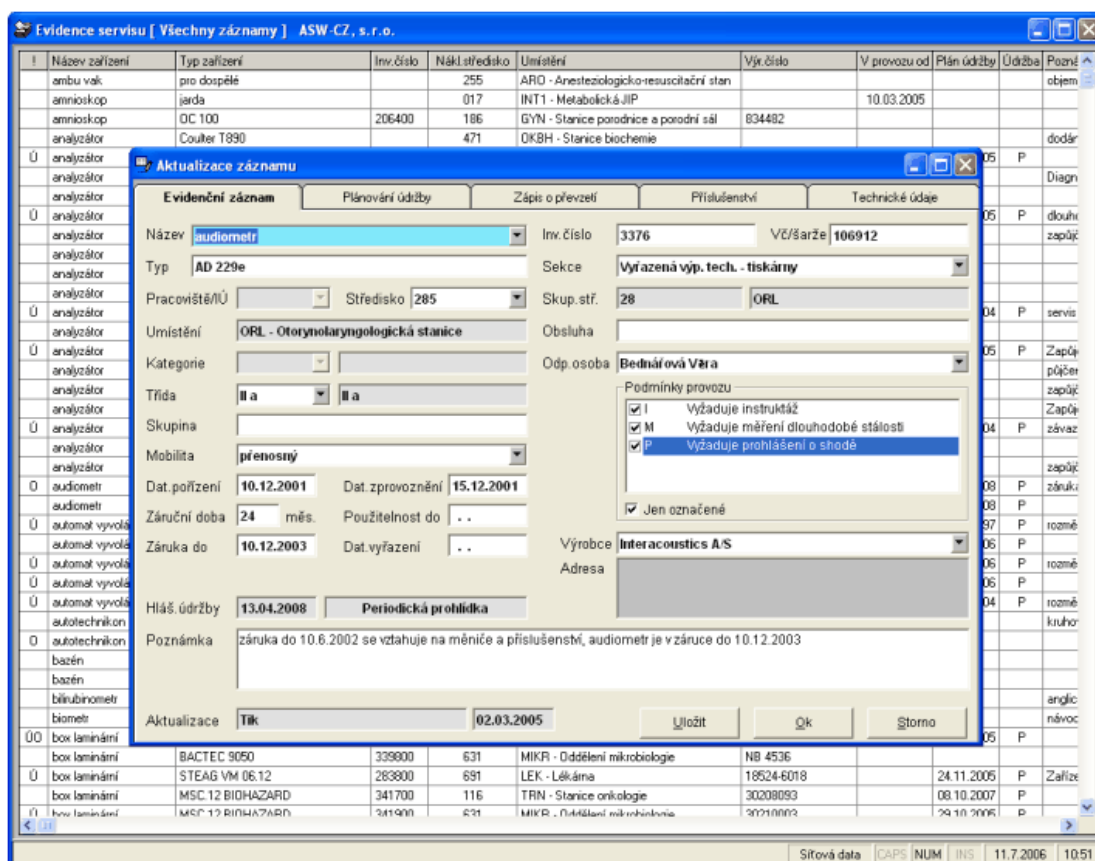
Dostupné na webových stránkách Evis Pro (<https://www.evispro.cz/evispro/index.php>), tento systém umožňuje vyzkoušení zkušební verze zdarma. Pracuje v počítačové síti nad SQL serverem. Obsahuje i grafy nákladů, evidence objednávek a náhradní díly.[20]

Výhody

- Přehled zařízení s možností tisku a exportu dat do Excelu.
- Upozornění na naplánovanou údržbu.
- Archivace a obnovení dat z archivu pro ochranu dat.
- neomezený počet revizí,
- možnost přikládání naskenovaných dokumentů.

Nevýhody

- cena pro 5 počítačů 44 700 Kč,
- každých dalších 5 počítačů +22 300 Kč,
- upgrade na novou verzi 13 900 Kč,
- technická podpora 11 900 Kč/rok.



■ Obrázek 2.2 Evis Pro: ukázka [20]

2.3.3 Online Revize

Dostupné na oficiálních webových stránkách <https://online-revize.cz/>, tento systém nabízí vyzkoušení na 30 dní zdarma, což umožňuje uživatelům otestovat všechny funkce před rozhodnutím o plném nasazení. Online Revize nabízí 3 výchozí balíčky: Start, Medium a Pro. Pro firmy s více než 2000 revizí je k dispozici možnost individuální nabídky. [21]

Výhody

- Flexibilní struktura - Systém nabízí možnost přizpůsobení struktury.
- Cloudové řešení - Data jsou uložena v cloudu, což zajišťuje dostupnost a snadný přístup.
- Přikládání naskenovaných dokumentů - Umožňuje přidávat dokumenty.
- Kalendář - Obsahuje kalendář pro sledování revizí.
- Automatické upomínky - Poskytuje automatické upomínky na nadcházející revize.

Nevýhody

- Cena se pohybuje od 450 Kč/měsíc až po 2 000 Kč/měsíc.
- Při dosažení hranice 2000 revizí je potřeba individuální jednání o ceně, což může zvýšit náklady.

2.3.4 Civop

Dostupné na <https://www.civop.cz>. Jedná se o firmu poskytující celkové převzetí správy revizí, včetně jejich provádění, ukládání historie atd. Poskytuje i samotný systém pouze pro správu revizí, kdy provádění si podnik kupující toto řešení hlídá sám. V tom případě je pak systém nastavený podle potřeb podniku, který řešení kupuje. Ukázka systému se bohužel na webových stránkách nenachází. Stejně tak jsem nenašel ani informace o ceně. Systém používá standardizovanou dokumentaci i procesy. Poskytuje 3 varianty:

- Basic - Pro malé firmy s maximálně 15 zaměstnanci a jedním pracovištěm.
- Medium - Nejrozšířenější varianta pro firmy s 16 až 200 zaměstnanci na jednom nebo více pracovištích.
- Special - Pro korporace s velkým počtem zaměstnanců (cca 300 až 2000) a sítí provozoven po celé ČR a SR.

Systém Civop představuje více než 30 let osvědčený systém, který efektivně dodává služby v oblasti BOZP, PO, ŽP a revizí. Zajišťuje maximální garance souladu s legislativou a efektivní řešení státních kontrol.[22]

2.3.5 EcoSun

Dostupné na <https://www.ecosun.one/cz/cmms>. Řešení je velmi rozsáhlé, s funkcemi, které jsou mimo rozsah zaměření této práce. Ukázka systému je přímo na webových stránkách, bohužel ve velmi nízkém rozlišení. Informace o ceně nejsou na stránkách uvedené.

Uživatelé mohou snadno plánovat, organizovat, a sledovat všechny úkony spojené s údržbou. Systém nabízí přehledné manažerské informace, statistiky a grafy, poskytující důležité údaje o využití zařízení, nákladech na údržbu a dalších relevantních informacích. Vlastnosti systému zahrnují evidenci zařízení, plánování preventivní údržby, řízení pracovníků, mobilní aplikaci pro práci mimo kancelář a mnoho dalších. Kombinuje praktičnost a uživatelskou přívětivost, čímž zjednodušuje a zefektivňuje každodenní údržbové operace.[23]

2.3.6 Eviduj si Revizi

Dostupné na oficiálních stránkách <https://www.evidujsirevizi.cz>. Zkušební verzi neumožňuje, na stránkách jsou uvedené snímky aplikace na stránce „ke stažení“. Firma také nabízí možnost externí správy revizí, kdy si klient pronajímá činnosti prováděné v systému a k přístupu používá pouze náhled.[24]

Výhody

- mobilní i stolní zařízení,
- cloudové řešení,
- generování uživatelských účtů zdarma,
- přehlednost, i na mobilním zařízení,
- kalendář,
- 27 oblastí pro sledování,
- automatická archivace,
- automatické upomínky.

Nevýhody

- pevná struktura,
- chybí zkušební verze,
- oproti konkurenci je cena pro počet pod 1000 položek vysoká (750 Kč/měsíc minimum),
- poplatek za každé založení činnost a PDF dokumentu 30 Kč,
- cena nad 1500 položek smluvní.

STŘEDISKO	OBJEKT	UMÍSTĚNÍ	NÁZEV	DODAVATEL	SLEDOVÁNO	ID	POSLEDNÍ REVIZE	PŘÍŠTÍ REVIZE	INVENTURNÍ ČÍSLO	VÝROBNÍ ČÍSLO	DRUH PALIVA	VÝKON (KW)	SPOTŘEBA (M3/H)	PŘETLAK (MBAR)	KVÍTY (IP)	EL. PŘÍKON (KW)	ROK VÝROBY	PODZÁMKA
FORM Solution Praha			Plynový kotel SAKI PRIME 24		✓	2046	24.09.2020	24.09.2023			plyn	24 kW					0	Gr M
FORM Solution Hradec Králové			Administrativní budova		✓	2282	15.12.2022	15.12.2025									0	Gr M
FORM Solution Praha			Kyslíková stanice	FEROX a.s. DČCh	✓	647	15.10.2020	15.10.2023		33503							1991	Gr M
FORM Solution Hradec Králové			Plynový kotel THERM 28 KDS3-A		✓	1004	01.12.2020	01.12.2023	Inventurní číslo	Výrobní číslo		8,4-24,6					0	Gr M
FORM Solution Praha			Administrativní budova		?	1493				TPG 704 01 a TPG 700 01							0	Gr M
FORM Solution Hradec Králové			Administrativní budova		✓	1967	08.06.2021	08.06.2024		8818	propan-butan		15,6				1998	Gr M
FORM Solution Praha			Vernovník sídlo		✗	1807	18.12.2019	18.12.2022									0	Gr M
FORM Solution Hradec Králové			Kotelna administrativní budova		✓	1068	02.06.2021	02.06.2024			plyn						0	Gr M
FORM Solution Praha			Kotelna AS		✓	1069	02.06.2021	02.06.2024									0	Gr M
411 TSR			Kotelna		✓	2140	02.06.2021	02.06.2024									0	Odsouh Gr M

■ Obrázek 2.3 Eviduj si Revizi: ukázka seznamu zařízení [24]

2.3.7 Inσιο

Inσιο přináší profesionální platformu pro řízení údržby a revizí. Umožňuje zkušební verzi na 30 dní. Je dostupný na webových stránkách <https://insio.cz/aplikace/servisni-system/>.

Inσιο slibuje 100% kontrolu nad zákonnými revizemi. Poskytuje tím klientům jistotu v dodržení termínů servisních úkonů. Děnně zpracovává více než 15 000 dokumentů a spolupracuje se známými firmami. Kromě servisního systému nabízí Inσιο i další produkty, jako např. systém řízení nákupu, podnikový informační systém ERP, řízení zakázek, cestovní příkazy a další. Tím umožňuje klientům sestavit si balíček produktů na míru.[25]

Výhody

- Mobilní i stolní zařízení.
- Cloudové řešení - Data jsou uložena v cloudu, což zajišťuje dostupnost a snadný přístup.
- Nastavení na míru - Možnost přizpůsobit systém potřebám uživatele naznačuje flexibilitu.
- Historie úprav - Historie úprav může být užitečná pro sledování změn a zodpovědnost.
- Plánovač práce - Funkce plánovače práce může být klíčová pro efektivní správu úkolů.
- Možnost přikládání naskenovaných dokumentů.

Nevýhody

- konfigurace systému 1800 Kč/hod,
- úprava systému 2200 Kč/hod.

The screenshot displays the 'Servisní systém' interface. At the top, there are search and navigation options. Below, a complex filter panel allows users to refine results by 'Agende', 'Zákazník', 'Objekt', 'Technologie', 'Zařízení', 'Odborní úkon', 'Odborní skupina', 'Typ úkonu', 'Dne', 'Plán', 'Důl do splnění', 'Skutečnost', 'Zákazník', 'Objekt', 'Zařízení', 'Technologie', 'Řídicí úkon', and 'Řídicí'. A table below lists individual service tasks with columns for ID, name, type, date, plan, completion status, customer, object, device, technology, control action, and control. The table contains 20 rows of data, including tasks like 'Hlášení problému', 'Týdenní servisní prohlídka', 'Zákonná revize', and 'Požadavek na údržbu'.

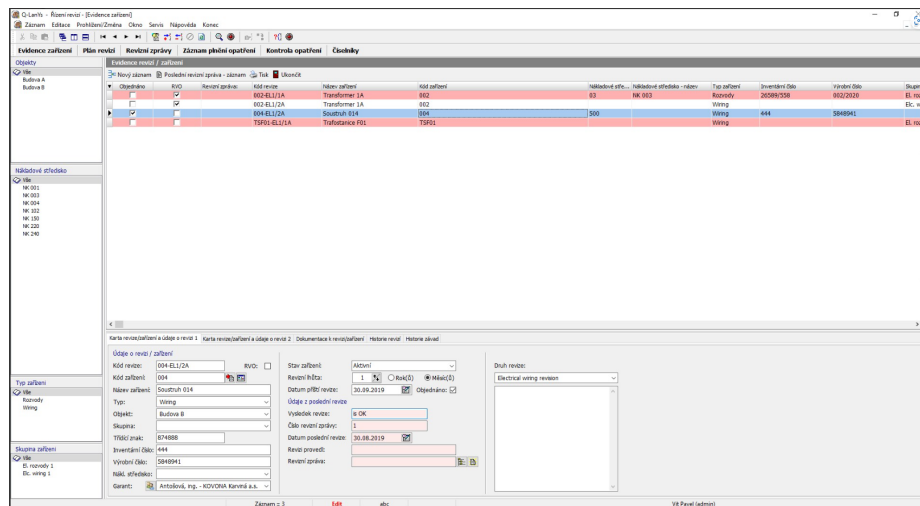
■ Obrázek 2.4 INSIO: ukázka [25]

2.3.8 Q-LanYs

Dostupné na <https://qlanys.cz/cz/revize-zarizeni-a-budov>. Hlavní funkce odpovídají tomu, co by člověk předpokládal pro revizní systém. Evidence strojů, zařízení a budov zajišťuje detailní seznam všech zařízení, strojů a budov s možností záznamu klíčových informací. Evidence revizí umožňuje plánování a sledování revizních činností. Revizní proces zde obsahuje 3 hlavní body:

- Revizní zprávy - Generování podrobných zpráv ze zjištěných revizí.
- Záznam zjištěných závad - Ukládání a kategorizace závad zjištěných během revizí.
- Návrhy nápravných opatření - Definování opatření k odstranění zjištěných závad.[26]

Systém umožňuje i přístup přes mobilní zařízení pomocí mobilní aplikace pro lepší flexibilitu a rychlost. Zároveň aplikace obsahuje kalendář. Na stránkách není uvedena informace o ceně řešení, je potřeba nechat si rovnou nezávazně vytvořit poptávku.



■ Obrázek 2.5 Q-LanYs: ukázka [26]

2.4 Vlastní systém

Může se jednat například o jednoduchou aplikaci vytvořenou v Microsoft Access, s použitím sdílené databáze. Lepší je potom vlastní aplikace nebo aplikace webová. I takový systém, pokud je udělaný důkladně dokáže obstarat veškeré potřebné funkce.

Systém vytvořený vlastními zaměstnanci, nejlépe i těmi, kteří se v dané doméně pohybují je ten ideální z hlediska jeho použitelnosti přesně tak, jak si to nadřízený představuje. Obsahuje hlavně ty funkcionality, které jsou potřeba. Provádění změn je jednoduché a reakce na změny jsou v rámci možností rychlé. Systém je takto flexibilní pro možnosti rozšíření v souladu s budoucími potřebami organizace.

Pro takový vývoj je potřeba mít alespoň jednoho zdatného programátora, nebo rovnou celý tým, který systém vytvoří, následně ho bude spravovat a provádět případné úpravy, změny a aktualizace. Je tedy potřeba rozmyslet se nad tím, jak vysoké budou náklady spojené s provozem systému a hlavně platy zaměstnanců při vývoji a údržbě.

Nasazení vlastního vytvořeného systému bude jednodušší, než zavedení koupeného, hotového řešení. Při vývoji se už analyzují veškerá potřebná data a funkce, které budou potřeba, včetně způsobů, jakým půjde jednoduše data převést, nebo rovnou použít, například pokud už se

používal nějaký druh databáze. Zároveň proces migrace dat bude proveden s minimálním časem přerušení.

Uživatelské rozhraní je vytvořené přívětivě pro uživatele, podle prvků na které jsou zvyklí, například v jiných aplikacích, které se ve firmě používají. Zároveň minimalizuje potřebu školení zaměstnanců a zajišťuje rychlou adaptaci nového systému.

Z bezpečnostního hlediska je možné použít způsob zabezpečení, který se ve firmě používá a na míru ho upravit, aby správně splňoval všechny požadavky, které firma na zabezpečení vyžaduje. Například se může jednat o přihlašování pomocí firemního účtu Windows, Google, nebo jiného účtu.

Výhody

- vytvořen přímo na míru,
- jednoduché zahrnutí existujících dat ze starého řešení,
- škálovatelnost a flexibilita,
- individuální podpora,
- rychlá reakce na nové potřeby,
- nezávislost na externích dodavatelích,
- není potřeba placená licence,
- centralizovaná správa dat,
- možnost integrace s interními systémy,
- dlouhodobá udržitelnost.

Nevýhody

- potřeba zaměstnanců, kteří jsou schopní systém vytvořit,
- potřeba vlastního serveru,
- cena a čas potřebný na vývoj systému,
- cena a čas strávený správou systému.

2.4.1 Důvody pro volbu vlastního řešení v konkrétní firmě

Každá firma má svoje způsoby a nařízení, která se musí dodržovat. Ve firmě, pro kterou systém předělávám, jsou nařízení velmi přísná. Pokud by se úspěšně prošlo procesem na získání povolení použití externího software se nesmí zapomenout na náklady spojené se zakoupením řešení. Žádné z existujících řešení, které jsem analyzoval nelze použít bez úprav. Vytvoření nebo upravení externího softwaru by bylo velmi drahé, a je možné, že se ani nenažde řešení, které by splňovalo veškeré požadavky (například ověřování uživatele pomocí účtu z Active Directory). Další jsou náklady spojené s údržbou po zavedení softwaru.

Pokud by byla snaha použít řešení bez jeho upravení, museli by se upravit procesy, které se ve firmě používají. S tím je prvně spojená důkladná analýza ještě před výběrem řešení, hledání řešení, které se nejvíce pro procesy hodí a zakoupení řešení. Tím to ale nekončí, je potřeba migrovat všechna data, zaškolit zaměstnance, pořádně otestovat jestli všechno funguje, tak jak se předpokládalo. Tohle všechno je čas strávený bez jakéhokoli zvýšení výdělku nebo usnadnění práce. Jak to často ve firmách chodí, „Dokud to funguje správně a spolehlivě, tak to měnit nebudeme.“

Jelikož vše ve staré aplikaci funguje, tak není možné získat důvod pro volbu nového, navíc externího řešení. V případě vytváření systému interně, na základě již fungující aplikace je tato situace pro firmu výhodnější. Spojeno s tím, že mají na oddělení studenta, na pozici praktikanta, se zaměřením na vývoj software se situace dále zlepšuje. Navíc pokud by si to student vzal

na starost jako diplomovou práci, bude na systému pracovat i ve volném čase. Tento způsob předělání aplikace pro správu revizí byl vyhodnocen jako proveditelný. Nezvednou se náklady, žádná práce navíc se zaměstnancům nepřidá a přechod na novou aplikaci, jelikož bude vyvinuta na míru, nebude náročný.

Vytvořením vlastního řešení se docílilo toho, že nejsou potřeba získat náročná povolení. Při vývoji lze počítat s aktuálním uložením dat a připravit jednoduše jejich změny pro použití v novém řešení. To sníží náklady na vývoj, zároveň bude aplikace vytvářena praktikantem a pro jeho údržbu a změny po jeho odchodu je na oddělení ještě jeden člověk se znalostmi v programování, který bude možný díky dokumentaci aplikaci spravovat. Změny a nové funkcionality půjde jednoduše a rychle udělat. Vyvíjet se bude jen pro dané oddělení, tedy určité vybrané druhy zařízení. Pro další druhy zařízení stačí aplikaci pouze připravit, aby šli jednoduše přidat.

Důvody

- Přizpůsobení přímo jednoduše z aktuálního uložení dat.
- Ušetření peněz, bude jednoduché udělat změny a přidat nové funkce.
- Obecně pružnost aplikace.
- Lokální podpora a rychlá odezva na problémy.
- Vlastní systém fyzických revizí.
- Nejedná se o obecnou aplikaci, která by měla podporovat všechno.
- Dlouhodobá udržitelnost.
- Strašně přísné kontroly aplikací mimo firmu.
- Nezávislost na externích dodavatelích.
- Možnost integrace s interními systémy.

2.5 Shrnutí existujících řešení

Fyzické ukládání, nebo používání kancelářských nástrojů pro ukládání revizí není vhodné, je dost nepraktické a pokud se nejedná o vážně malou firmu, tak se nevyplatí. Zakoupení specializovaného systému je většinou ideálním řešením, když firma začíná. Je to dobrá investice, jak předejít velkým nákladům při zavedení systému později. Vlastní řešení záleží především na kompetenci zaměstnanců, jestli je člověk schopný takový systém vytvořit nebo ne. Může se i jednat o jednoduchou aplikaci v Microsoft Access, ze které v této práci vycházím. Pokud je taková aplikace dobře navržena, dokáže sloužit velice dlouho. Další úroveň je pak vlastní aplikace, například webová, ta je náročnější na vytvoření, ale je to ideální řešení, pro připojení a použití odkudkoliv i ve více lidech zároveň.

Zakoupení nového řešení je ideální co nejdříve v průběhu životního cyklu firmy. Čím déle se zakoupení a nasazení takového systému odloží, tím větší budou náklady a pracnost spojené s jeho nasazením.

Prodávaných řešení na trhu existuje mnoho. Existuje i více druhů podle různých kritérií. Jsou firmy, které prodávají hotový systém s příplatky za aktualizace a podporu. Cloudová řešení, která se platí měsíčně podle různých parametrů, jako velikost úložiště, počet revizí, počet uživatelů atd.

Existují i firmy, které přímo převezmou provádění a správu revizí. Toto řešení je také dobré důkladně zvážit, jestli se pro danou firmu vyplatí po stránce nákladové, nebo jestli si firma bude revize provádět sama a bude potřebovat vyškolené zaměstnance na tuto pozici. Také je potřeba zvážit jestli bude zaměstnanec zodpovědný o samotnou správu revizí, jejich plánování a ukládání, nebo veškerou administrativní správu a práci s revizním systémem zvládne vykonat zaměstnanec, který revizi fyzicky provádí.

V následujících dvou tabulkách 2.1, 2.2 jsem shrnul prodávaná řešení, které jsem zmínil v předešlých sekcích. Do tabulek jsem nezavedl řešení od firmy EcoSun z důvodů překročení rozsahu služeb oproti ostatním řešením a systém od firmy Civop pro nedostatek informací.

■ **Tabulka 2.1** Souhrn prodáváných systémů 1

Specifikace	Facman	Evis Pro	Online Revize
Mobilní zařízení	✓	X	✓
Cloudové řešení	✓	X	✓
Flexibilní struktura	X		✓
Přiložení dokumentů	✓	✓	✓
Kalendář	X	✓	✓
Automatické upomínky	✓	✓	✓
Nastavení na míru	X	X	X
Počet uživatelů	5 - 30	5 počítačů	5 - neomezený
Počet revizí	200 - 3000	Neomezený	100 - 2000
Cena	450 - 1950 Kč/měs	44 700 Kč, příplatky	450 - 2000 Kč/měs

■ **Tabulka 2.2** Souhrn prodáváných systémů 2

Specifikace	Eviduj si Revizi	Insio	Q-LanYs
Mobilní zařízení	✓	✓	✓
Cloudové řešení	✓	✓	✓
Flexibilní struktura	X	✓	
Přiložení dokumentů	✓	✓	
Kalendář	✓	✓	✓
Automatické upomínky	✓	✓	✓
Nastavení na míru	X	1 800 Kč/h	
Počet uživatelů			
Počet revizí	100, více za poplatky		
Cena	850+ Kč/měsíc+poplatky		

Kapitola 3

Analýza

Cílem nového revizního systému je nahradit stávající a připravit se na rozšíření systému o další velkou kategorii elektronických zařízení, zároveň umožnit jednoduše provádět změny, které by se v aktuálním řešení prováděly složitě. Při analýze jsem vycházel jednak ze stávajícího řešení a dále z užitečných funkcí řešení existujících na trhu. Jak už jsem zmiňoval, nejedná se o obecnou aplikaci, ale o aplikaci vytvořenou přímo na míru procesů prováděných v podniku, kde pracuji.

Popisuji zde existující řešení a způsoby, kterými jde správa revizí provádět. V druhé části kapitoly se věnuji analýze funkčních a nefunkčních požadavků z pohledu stávajícího i nového řešení.

3.1 Analýza současného systému

Aplikace slouží jako jednotné místo pro správu revizí. Slouží pro přehledný pohled na platnosti revizí a termíny dalších. Je to pomocný nástroj, který pomáhá zavčas vytvořit objednávky (přípravou správných zařízení pro objednání) a jednoduše a rychle zapisovat a kontrolovat provedené revize. Nemí propojený s jinými systémy. Pro představu přikládám obrázek toho, jak vypadá úvodní stránka současné aplikace a formulář pro správu revizí o tlakových lahvích na obrázcích 3.1 a 3.2.

Aplikace obsahuje informace a funkce potřebné pro správu revizí spravovaných oddělením firmy, ve kterém se nachází. Obsahuje funkce od zobrazení seznamu nadcházejících revizí pro dané kategorie, po jejich zadávání a vytváření. Spolu s dalšími podpůrnými nástroji, například pro zobrazení všech revizí k danému stroji (stroj může mít několik revidovaných zařízení – tlaková lahev, senzor atd.) Pro přehlednost jsou zobrazeny upozornění na revize nadcházející v nejbližší době a případná varování na revize, které už měly být provedeny. Většinou se jedná o případy, kdy je revize hotová, ale ještě není protokol, nebo se stroj stěhuje. Při stěhování se revize provádí až na novém umístění, než se začne používat. Uživatel si v tomto případě napíše k zařízení poznámku a zařízení buď deaktivuje, nebo ho v aplikaci nechá s červeným varováním, aby nezapomněl. Důležitou součástí aplikace je také logování veškerých provedených změn uživateli, kteří jsou identifikováni uživatelským účtem systému Windows.

3.1.1 Kategorie

Zařízení jsou rozdělené do jednotlivých kategorií, které se zpracovávají zvlášť. Člověk má na starost jednu nebo více kategorií, o které se stará. Vytváří objednávky, kontroluje a zapisuje protokoly. Je tak zodpovědný za platnosti revizí na strojích.



■ Obrázek 3.1 Úvodní stránka současné aplikace

BOSCH Revision System - karta tlakové nádoby

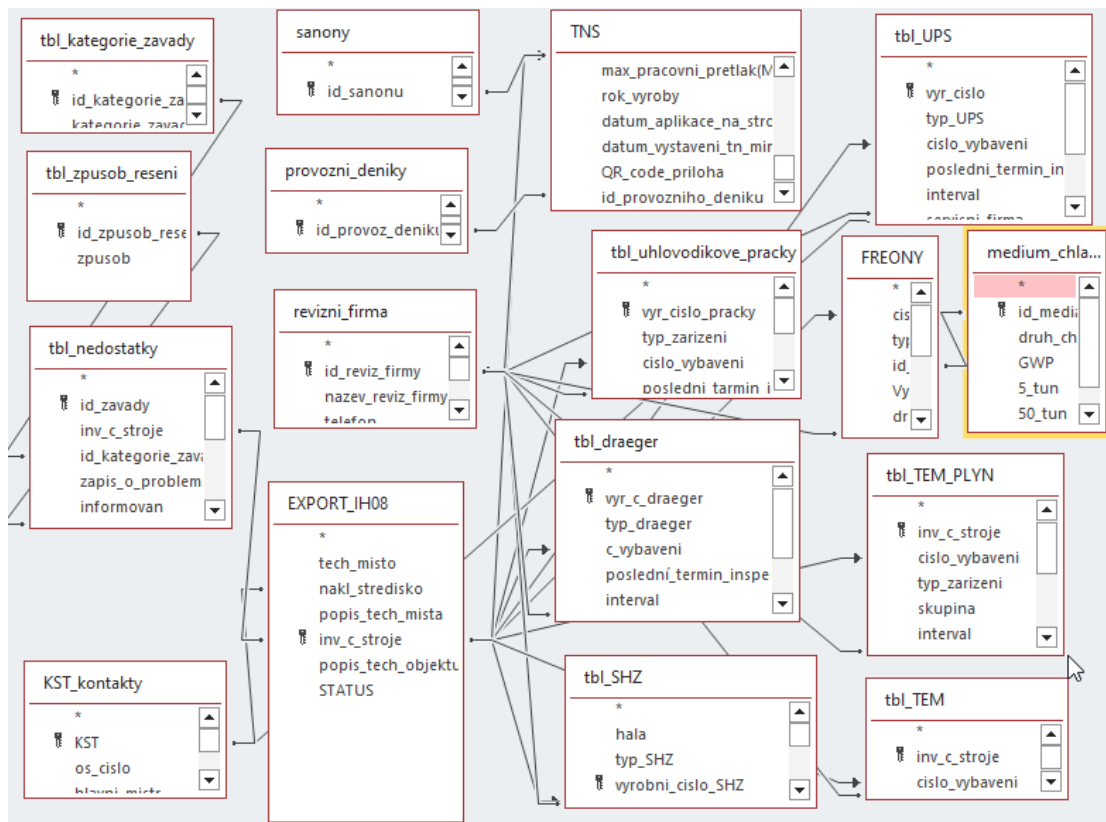
V.Č. tlakové nádoby: [text] Pasport odkaz: [text]
 Rok výroby: 2013
 Vých. revize provedena: [text] Max. prac. přetlak(Mpa): 21
 PR byla provedena: 10. 7. 2022 Objem nádoby(l): 10
 VR byla provedena: 16. 7. 2018 Aplikace na stroj (datum): [text]
 TR byla provedena: 10. 7. 2022 Vyst. mimo provoz (datum): [text]
 Provádí firma: [dropdown] Jištěno zdrojem tlaku:
 inv_c_stroje [text] Aktivní:
 Poznámka: [text area] Oprava reviz. dokladu:
 Nucené zobrazení:
 Umístění v šanonu: 070113
 Označení provozního deníku: [dropdown]

Umístění nádoby:

Technické místo: [text] Kontakt: [text]
 Nákladové středisko: [text] HL.Mistr: [text]
 Popis tech. místa: [text] Tel.: [text]
 popis tech. objektu: [text] Hala: [text]
 SAP status: [text]

Record: 1 of 1293

Obrázek 3.2 Formulář pro správu tlakových lahví



Obrázek 3.3 Databázové tabulky staré aplikace

Kategorie:

- TNS - tlakové nádoby.
- SHZ - automatická zhašecí zařízení.
- DRAEGER - snímače úniku plynu (DRAEGER je výrobce).
- Uhlovodíkové pračky - zařízení používaná k čištění těles uhlovodíkovými směsi
- Plynová zařízení - nádoby a zařízení obsahující plynové vedení.
- UPS - záložní zdroje (používaná u zařízení, která při výpadku musí standardně doběhnout).
- Freony - inspekce úniku regulovaných látek stanovených státem.
- Odstředivky - rotační zařízení.
- TEM - zařízení, používající termické odhroťování.

Jednotlivé kategorie mají různé vlastnosti, ale požadavky na funkce jsou podobné. Základními vlastnostmi položek jsou: číslo položky, datum poslední revize, číslo stroje (u kterého se nachází), servisní firma, poznámka, aktivní/deaktivovaný a další vlastnosti, které záleží na kategorii.

Revize tlakových nádob je historicky první část, pro kterou se aplikace v minulosti vyvíjela. Tato část aplikace obsahuje některé zastaralé prvky, které se už nepoužívají, ale i po jejich vynechání je kategorie tlakových lahví nejvíce odlišná od ostatních. Důvodem je mimo jiné velké množství, největší ve všech kategoriích. Fyzické revize provádí 2 firmy, kde každá má svůj seznam. Existuje možnost exportování do PDF, pro výtisk. Ten se používá k manuálnímu zápisu při provádění revizí a slouží k jednoduššímu přepisu do systému. Navíc ještě obsahuje seznam provozních deníků, které se používají k zápisu o kontrolách obsluhou nádob. Aplikace tedy obsahuje seznam o tom, kolik deníků existuje. K jakým strojům a jaké nádoby jsou v nich zapsané jde zjistit pouze manuálně procházením tlakových nádob.

3.1.2 Aktuální struktura databáze

Aktuální aplikace obsahuje 32 tabulek dat, 23 formulářů, 13 sestav a 2 skripty. Pro funkčnost formulářů a sestav je navíc vytvořeno 32 přidávacích dotazů, 2 mazací a 45 dotazů pro výběr dat s různými podmínkami omezení.

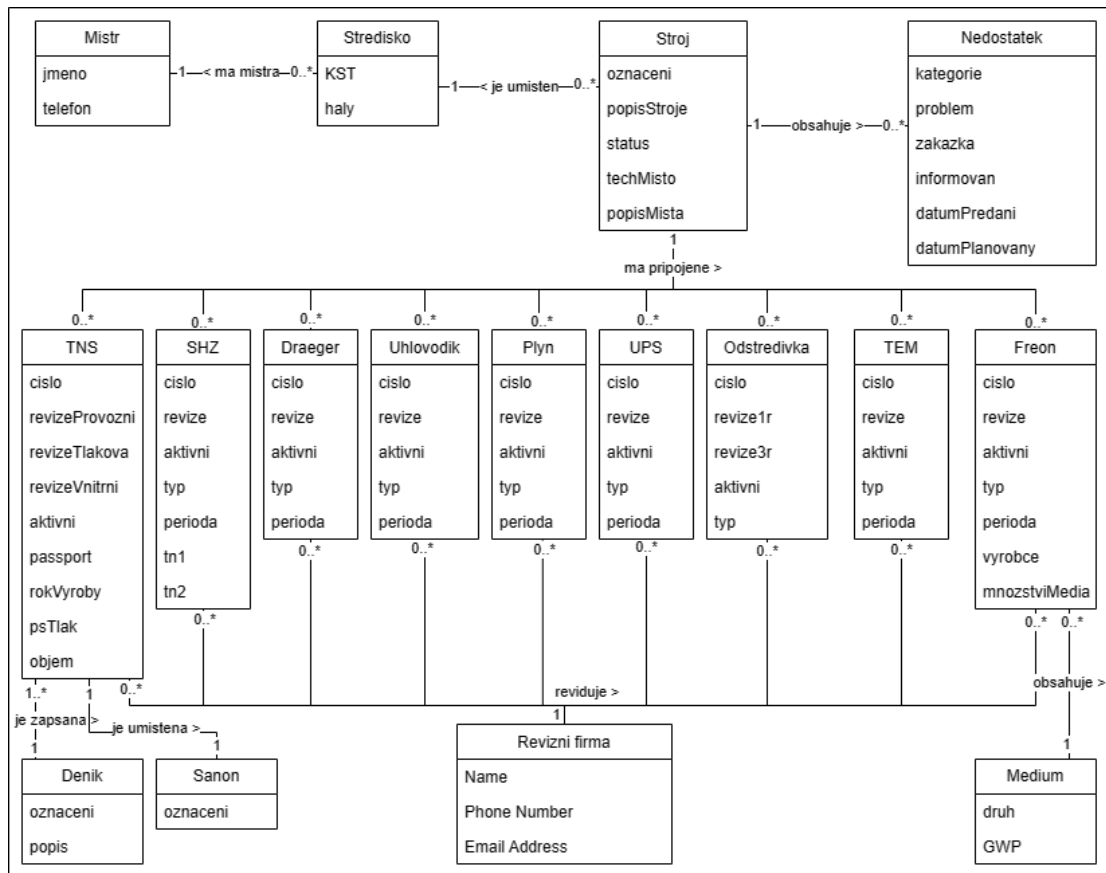
Každá kategorie má svojí tabulku, což je logické, ale postupem času jak se tabulky přidávali tak nejsou sjednocené názvy sloupců, některé názvy obsahují překlepy nebo nelogické názvy. Další tabulky jsou podpůrné tabulky například pro výběr do select boxů, nebo seznamy deníků a šanonů. Pro vedení záznamu o změnách je pro každou kategorii vlastní tabulka, navíc se neukládají informace o změnách v nedostatecích a nakonec existuje i tabulka, v které jsou logovány všechny změny posledních revizí. Logovací tabulky by šlo sloučit do jedné a pouze přidat sloupec pro indikaci, v které tabulce se změna provádí.

Důležitou částí správy revizí jsou protokoly. V aplikaci se cesta k protokolu zadává přímo jenom v jedné kategorii, pro ostatní kategorie je zadán standardizovaný způsob pojmenování souborů, který obsahuje číslo stroje a datum provedení. Protokoly jsou uloženy ve složkách pro každou kategorii zvlášť, a pokud je protokol špatně pojmenovaný, tak ho aplikace nenajde. Aplikace s protokoly pracuje jen při otevírání. Ukládání na správné místo a pod správným názvem musí uživatel udělat mimo aplikaci.

Na obrázku 3.3 přikládám ukázkou většiny tabulek spolu s jejich vztahy. Je to spíše obrázek se seznamem tabulek, než diagram, jelikož pro přehlednost by diagram musel být velký, nebo rozdělený do několika částí. Přehlednější a větší verzi jsem přidal jako přílohu A.2, příloha A navíc obsahuje další obrázky z aplikace.

3.1.3 Doménový model

Doménový model slouží k popisu klíčových prvků a vzájemných vztahů. Pomáhá porozumět dané doméně. V modelu na obrázku 3.4 jsou vidět vztahy mezi jednotlivými prvky aplikace.



Obrázek 3.4 Doménový model aplikace

3.1.4 Požadavky vycházející ze současné aplikace

Funkční požadavky vycházející z analýzy jsem sepsal v následujícím seznamu, jedná se o veškeré požadavky, které byly vytvořeny při vývoji a následných úpravách současného řešení. Vynechal jsem zastaralé funkce, které se už nepoužívají, ale z nějakého důvodu v aplikaci zůstaly. V této části analýzy zmiňuji jen seznam požadavků a v následující kapitole požadavky přiblížím více do hloubky, spolu s potřebnými úpravami pro nové řešení.

Funkční požadavky

- identifikace uživatele pomocí účtu Windows,
- logování provedených změn,
- zobrazení nadcházejících revizí (časové období záleží na firmě),
- upozornění na nadcházející revize a jejich souhrn,
- výstražné varování na propadlé revize,

- vytváření položek v jednotlivých kategoriích,
- upravování vlastností položek,
- zobrazení revizí provedených na konkrétním stroji,
- vytváření záznamů o nedostatcích,
- zobrazení nevyřízených nedostatků,
- vytváření a zobrazování kontaktů na externí firmy,
- možnost tisku seznamu pro tlakové lahve (v nové aplikaci musí být zachován stejný styl),
- provozní deníky pro tlakové lahve,
- zařazení do šanonů pro tlakové lahve.

Nefunkční požadavky

- přístup více uživatelů naráz,
- oprávnění přístupu je omezeno právy přístupu ke složce,
- všechna data uložena na jednom místě,
- oddělený přístup pro administrátora (zobrazení skrytých struktur).

3.1.5 Současné uživatelské rozhraní

Způsobu, jakými analyzovat uživatelské rozhraní je několik. Například pomocí dotazníku, pomocí rozhovorů s uživateli systému, pomocí uživatelských scénářů nebo při skupinové diskuzi. Pro účely této práce jsem zvolil kvalitativní výzkum se zadavatelem projektu. Vedli jsme diskuze nad jednotlivými problémy a změnami v uživatelském rozhraní. První diskuze proběhla po analýze současného řešení, pro kterou jsem si připravil první návrh. Při diskuzi jsme prošli návrh a představil jsem svojí vizi, jak bych mohl nový systém obohatit za účelem zvýšení použitelnosti. Následně jsme vedli další diskuze nad každou další verzí návrhu a postupně jsem se dostal k řešení, s kterým jsem byl spokojený a začal implementovat. Diskuze byly různě dlouhé, občas se jednalo pouze o textovou konverzaci, při návrhu s více změnami, potom osobní nebo vzdálené rozhovory.

Obecně je uživatelské rozhraní v rámci možností MS Access velmi použitelné a funkční, ale mohlo by se vylepšit. V této kapitole se budu zabývat funkcemi, které mi v uživatelském rozhraní chyběly. Některé věci jsem našel při prvním použití aplikace a další až v průběhu iterativních diskuzí nad jednotlivými návrhy. Nejedná se o žádné zásadní nedostatky, jelikož aplikace se aktivně používá a na požadavky uživatelů se i několikrát změnila. I přes to je zde pár věcí, které uvedu.

3.1.5.1 Chybějící funkce a nedostatky

Hlavní věc, co mi chybí je možnost upravit zařízení přímo ze seznamu. Například pomocí tlačítka pro přesun do formuláře zařízení. Další na seznamu je nejednotnost názvů, jak názvů vlastností ve formulářích, tak i názvy sloupců v tabulkách. Navíc nejsou ani formuláře podobně seřazené. Například bych očekával, že ve všech formulářích bude nejprve číslo zařízení, datum poslední revize, stroj a následně další vlastnosti. Prvně ty co má více kategorií společné a nakonec ty specifické. Jediné pole vždy na stejném místě je pole pro poznámku. Důvodem této nejednotnosti je především postupné stavění aplikace, ale i to, že každý uživatel spravuje jen jednu nebo dvě kategorie a tak mu to nepřijde tak důležité.

Pro vyhledávání zařízení v dané kategorii se používá funkce MS Access „Vyhledat“ pomocí umístění kurzoru do pole kde chce uživatel hledat a klávesové zkratky CTRL+F. Takto je možné vyhledávat podle kteréhokoliv pole. Hledání podle čehokoli je ale zbytečné, navíc to s sebou

přináší místo, kde uživatelé občas dělají chyby. Místo vyhledávání nechtěně přepíše pole, ve kterém se nacházejí a když pole opustí tak se změna rovnou uloží. To mě přivádí k další funkci, co mi chybí, kterou je potvrzování změn nebo alespoň upozornění na provedené změny.

Pro zobrazení všech revizí prováděných na stroji se používá formulář, pro otevření je nutné zadat číslo stroje, zde bych očekával možnost výběru ze seznamu, nebo alespoň našeptávání. Po otevření formuláře jsou zobrazené informace o stroji a pole pro každý typ revize. Pokud se na stroji daná revize neprovádí, je uvedeno „Neprovádí se“. To by ještě takový problém nebyl, ten hlavní problém nastává v situaci, kdy se na stroji nachází více než jedno zařízení ze stejné kategorie, pak je nutné přepínat mezi záznamy pomocí vestavěné funkce MS Access na přepínání záznamů ve formuláři pomocí malého tlačítka v levé dolní části okna.

Shrnutí nedostatků a chybějících funkcí

- Chybí možnost úpravy zařízení ze seznamu.
- Nejednotnost názvů a pořadí ve formulářích a seznamech.
- Chybí potvrzení nebo upozornění na provedenou změnu.
- Nepřehledná stránka souhrnu revizí na stroji.

3.2 Analýza požadavků pro novou aplikaci

Při návrhu nové aplikace jsem se zaměřil na zachování klíčových prvků stávajícího systému a zároveň jsem identifikoval příležitosti k optimalizaci a nové funkčnosti pro zjednodušení práce. Na základě analýzy současné aplikace jsem našel místa, kde přidáním funkcionality, která nebyla v původním systému, docílím efektivnější práce.

Tato kapitola obsahuje celkový soupis všech požadavků, s krátkým popisem, prioritou a odhadem složitosti. Pro představu také uvádím příklady, při kterých se daný požadavek používá.

3.2.1 Uživatelé

Typ uživatelů z pohledu systému je, pokud nepočítám uživatele, úplně bez přístupu, pouze jeden. Jedná se tedy o uživatele, který objednává, kontroluje a zapisuje do systému informace o provedených revizích. Veškerí uživatelé aplikace budou před použitím systému s ním seznámeni, nebo projdou školením. Jediné omezení na uživatele je omezení přístupu do jednotlivých kategorií.

3.2.2 Funkční požadavky

V této kapitole podrobně popisují funkční požadavky. Ke každému požadavku navíc přidávám popis toho, co očekává uživatel a scénář toho, jak to systém bude řešit.

3.2.2.1 F1: Identifikace uživatele pomocí účtu Windows

Systém ověřuje uživatele přes Active Directory (AD). Přístupy a oprávnění v aplikaci se budou odvíjet z informací z AD. Může se jednat o oprávnění pro skupiny nebo jednotlivce.

Priorita: vysoká

Složitost: vysoká

Popis: Zadavatel očekává, že přístup do aplikace budou mít jenom vybraní zaměstnanci.

Scénář: Systém umožní přístup do aplikace pouze zaměstnancům s oprávněním přístupu.

Zaměstnanci jsou identifikováni účtem, pod kterým jsou přihlášení v systému Windows.

3.2.2.2 F2: Zaznamenávání změn

Aplikace loguje všechny prováděné změny v datech a umožňuje přístup k logu změn.

Priorita: střední

Složitost: střední

Popis: Uživatel očekává možnost kontrolovat provedené změny v systému.

Scénář: Systém umožní zobrazení tabulky s logem provedených změn. Bude obsahovat informace o uživateli, času změny a o samotné změně (data před i po změně).

3.2.2.3 F3: Zobrazení nadcházejících revizí

Aplikace podporuje zobrazení nadcházejících revizí v přehledné formě. Obsahuje možnosti zobrazení podle kategorií zařízení a možnost exportu.

Priorita: vysoká

Složitost: lehká až vysoká

Popis: Uživatel očekává přehledné zobrazení nadcházejících revizí pro danou kategorii. Navíc očekává i možnost exportu dat pro případné objednávky nebo tisk.

Scénář: Systém zobrazí počty nadcházejících revizí pro dané kategorie přímo na úvodní stránce aplikace. Po otevření kategorie zobrazí podrobný seznam s možností exportu.

3.2.2.4 F4: Souhrn revizí a upozornění

Aplikace zobrazuje celkový počet nadcházejících revizí k jednotlivým kategoriím. V případě revize, která už měla být provedena zobrazí výrazné varování.

Priorita: vysoká

Složitost: lehká

Popis: Uživatel očekává jednoznačně a ihned viditelné indikátory počtu nadcházejících a propadlých revizí.

Scénář: Systém zobrazí počty nadcházejících revizí pro dané kategorie přímo na úvodní stránce aplikace. V případě propadlých revizí bude počet zobrazený viditelně červeným písmem.

3.2.2.5 F5: Vytváření položek

Aplikace umožní vytváření nových položek ve všech kategoriích se všemi jejich vlastnostmi.

Priorita: střední

Složitost: střední

Popis: Uživatel očekává možnost přidání nového zařízení do systému.

Scénář: Systém umožní v kategoriích přidávat nové položky, spolu se všemi jejich náležitostmi.

3.2.2.6 F6: Upravování položek

Aplikace umožní provádět změny u položek ve všech kategoriích, především u vlastnosti „poslední revize“. Položky, které se často nemění mohou být změněny pouze po odemčení změn, nebo potvrzení upozornění, že se změna opravdu provádí.

Priorita: střední

Složitost: střední

Popis: Uživatel očekává, že bude možné změnit vlastnosti jednotlivých zařízení. Při změně vlastností, které mění málokdy chce být uživatel upozorněn, nebo přinucen se zamyslet nad tím, že změnu provádí.

Scénář: Systém umožní pro každé zařízení provádět změny. Změny data revize jsou umožněné kdykoliv. Pro změny jiných vlastností se buď zobrazí upozornění, nebo bude potřeba nejprve změny povolit kliknutím na tlačítko u položky.

3.2.2.7 F7: Zobrazení revizí provedených na stroji

Aplikace umožní zobrazit informace ke stroji, zároveň se všemi provedenými revizemi, které se ke stroji provádí. Stroje jde vyhledávat podle inventárního čísla. Provedené revize obsahují odkaz na naskenovaný dokument revize.

Priorita: střední

Složitost: střední

Popis: Uživatel očekává možnost zobrazit si všechny revize ke stroji na jednom místě.

Scénář: Systém umožní pro vyhledaný stroj podle jeho čísla zobrazit všechny jeho informace včetně všech souvisejících zařízení a jejich revizí, s platnostmi a odkazy na protokoly.

3.2.2.8 F8: Vytváření záznamu o zjištěném nedostatku

Aplikace umožní zadávání nalezených nedostatků.

Priorita: střední

Složitost: lehká

Popis: Uživatel očekává možnost zápisu informace o nalezeném nedostatku při provedení revize. Tento zápis pak používá jako upomínku, aby nezapomněl, že má být nedostatek odstraněn.

Scénář: Systém umožní vytvořit nedostatek s veškerými potřebnými vlastnostmi, včetně možnosti úprav a potvrzení o odstranění.

3.2.2.9 F9: Seznam nedostatků

Aplikace zobrazí seznam všech zadaných, nevyřešených i vyřešených nedostatků.

Priorita: vysoká

Složitost: lehká

Popis: Uživatel očekává možnost zobrazit si všechny nedostatky.

Scénář: Systém umožní zobrazit seznam všech nebo jen otevřených nedostatků.

3.2.2.10 F10: Kontakty na externí firmy

Aplikace umožní zobrazit a vytvářet kontakty na externí firmy.

Priorita: nízká

Složitost: lehká

Popis: Uživatel očekává možnost zadat novou firmu pro provádění revizí, s poznámkami a časovým předstihem, o kolik měsíců dopředu se revize prováděná danou firmou má zobrazit jako nadcházející.

Scénář: Systém umožní vytvořit a upravit informace o externí firmě.

3.2.2.11 F11: Kontakty na mistry

Aplikace umožní zobrazit, vytvářet a upravovat kontakty na mistry zodpovědné za dané střediska.

Priorita: nízká

Složitost: lehká

Popis: Uživatel očekává možnost zadat nebo upravit mistra ke středisku, včetně informací o umístění a telefonu.

Scénář: Systém umožní zobrazit, vytvořit a upravit informace o mistrech.

3.2.2.12 F12: Možnost tisku seznamu pro tlakové lahve

Aplikace umožní zobrazení, nebo generování dokumentu vhodného pro tisk u nadcházejících revizí tlakových lahví. Je potřeba zachovat formát ze staré aplikace.

Priorita: střední

Složitost: střední

Popis: Uživatel očekává zachování formátu tištěného seznamu pro tlakové lahve. Tento seznam předává revizní firmě, a ta požaduje seznam v tomto formátu.

Scénář: Systém umožní exportování seznamu v předdefinovaném formátu.

3.2.2.13 F13: Tlakové lahve – provozní deníky

Aplikace umožní zobrazení provozních deníků u kategorie tlakové lahve. Každý deník obsahuje informace o zařízeních a strojích, které jsou v něm zavedené.

Priorita: nízká

Složitost: lehká

Popis: Uživatel očekává možnost zobrazení a zadání provozního deníku k tlakovým lahvím.

Scénář: Systém umožní deníky zobrazovat a vytvářet, zároveň podporuje i možnost zařazení zařízení do deníku.

3.2.2.14 F14: Tlakové lahve – zařazení v šanonu

Aplikace umožní vytvoření nového záznamu o zařazení do šanonu u kategorie tlakové lahve. Záznam je unikátní a může se na dané místo zařadit vždy jen jedna láhev.

Priorita: nízká

Složitost: lehká

Popis: Uživatel očekává možnost zapsání informace, v jakém šanonu se fyzicky uchovávají protokoly k dané tlakové lahvi. Může i založit nový šanon, potřebuje tedy i možnost vytvořit nové možnosti.

Scénář: Systém umožní přiřadit umístění v šanonu k dané lahvi. Na jedno umístění je možné zařadit jenom jedno zařízení. Umožní také vytvořit nová umístění.

3.2.2.15 F15: Omezení přístupu na části aplikace

Aplikace umožní omezovat přístup uživatelům na jednotlivé části aplikace.

Priorita: nízká

Složitost: lehká

Popis: Zadavatel očekává možnost regulovat přístupy uživatelů do jednotlivých částí aplikace.

Scénář: Systém umožní řídit přístupy uživatelů.

3.2.3 Nefunkční požadavky

V této kapitole popisují nefunkční požadavky, které vyšly z analýzy nebo byli přímo zadané zadavatelem.

3.2.3.1 N1: Webová aplikace

Bude se jednat o webovou aplikaci, která bude přístupná jen automatickým přihlášením přes Active Directory firmy.

3.2.3.2 N2: Technologie

Aplikace bude používat technologie podporované společností. Zároveň se musí jednat o technologie, které zvládne někdo ze stálých zaměstnanců spravovat. Konkrétně se jedná o framework Nette založený na php.

3.2.3.3 N3: Dokumentace

Za účelem udržitelnosti se bude používat samo dokumentující kód a případné složitější věci budou zdokumentované v kódu. V případě potřeby budou vytvořeny návody pro potřebná nastavení nebo další nejednoznačné funkce.

3.2.3.4 N4: Data

Data jsou aktuálně pouze uložena v Access souboru, bude potřeba je přesunout do databáze na serveru, případně navrhnout a implementovat změny v databázi, pokud to bude potřeba.

3.2.3.5 N5: Uložení protokolů

Protokoly jsou ukládané na sdílený disk, který je pravidelně a důkladně zálohovaný. Zadavatel chce zachovat místo uložení protokolů. Server, na kterém aplikace poběží přístup k disku má. Je potřeba zajistit způsob, jakým bude umožněno protokoly uživatelům zobrazovat.

3.2.3.6 N6: Zachování základního vzhledu

Zadavatel očekává rychlou adaptaci uživatelů na novou aplikaci, proto je potřeba aby aplikace do určité míry zachovala vzhled a umístění základních funkcí.

Kapitola 4

Návrh

V této kapitole se zabývám převodem uživatelského rozhraní ze staré aplikace do nové podoby ve webovém prostředí spolu s novými funkcemi a potřebnými změnami, které práci s aplikací zrychlí. Vycházím také z předchozí provedené analýzy. Pro přehled funkčnosti z pohledu uživatele jsem vytvořil soupis případů užití.

První částí návrhu GUI byl jednoduchý hi-fi prototyp, ve kterém jsem se zaměřil pouze na rozmístění funkčních prvků, navigace, tlačítek a formulářů. V podstatě se jednalo o papírový model, ve kterém jsem použil různé barvy pro oddělení prvků a vytvoření větší představy o výsledném řešení, kde jsem ale barvy úplně stejně nezachoval.

4.1 Uživatelské role

Aplikace druh uživatele nerozlišuje. Uživatel bez oprávnění se do aplikace nedostane a uživatel s přístupem má přístup ke všem funkcím. Jediné omezení, které se na uživatele může přidat je omezení přístupu na jednotlivé stránky. Tuto změnu musí provést správce aplikace v databázi.

4.2 Případy užití

Případy užití slouží k popisu funkcionality aplikace z pohledu uživatele. Případy užití začínají z úvodní stránky aplikace.

4.2.1 Správa zařízení

Zobrazení nadcházejících revizí

Uživatel klikne na dlaždici kategorie. Aplikace zobrazí seznam s nadcházejícími revizemi.

Zobrazení všech zařízení

Uživatel klikne na dlaždici kategorie. Aplikace zobrazí seznam s nadcházejícími revizemi. V pravém horním rohu klikne na „Filtrovat vše“. Aplikace zobrazí všechna zařízení.

Zkopírování tabulky

Uživatel klikne na dlaždici kategorie. Aplikace zobrazí seznam s revizemi. V pravém horním rohu klikne na „Kopírovat tabulku“. Aplikace zkopíruje tabulku do schránky uživatele.

Přidání zařízení

Uživatel klikne v dlaždici kategorii na tlačítko „Karta“. Aplikace zobrazí formulář pro zařízení dané kategorie. Pokud uživatel v kategorii zařízení nehledal, je formulář prázdný a uživatel

ho může vyplnit a odeslat pro vytvoření zařízení. V případě, že je formulář předvyplněný jinými daty, uživatel klikne na tlačítko s ikonou plus, která formulář vyprázdní pro vytváření nového zařízení.

Upravení zařízení

Uživatel klikne v dlaždici kategorie na tlačítko „Karta“. Aplikace zobrazí formulář pro zařízení dané kategorie s vyhledávacími poli pro zařízení a stroje. Uživatel vyhledá zařízení, které chce upravit. Do formuláře se načtou data. Uživatel data upraví a formulář odešle.

4.2.2 Tlakové lahve

Zobrazení provozních deníků

Uživatel v menu klikne na TNS a vybere „Provozní deníky“. Aplikace zobrazí seznam provozních deníků.

Přidání provozního deníku

Uživatel v menu klikne na TNS a vybere „Provozní deníky“. Aplikace zobrazí seznam provozních deníků. V hlavičce, v pravém rohu klikne na tlačítko pro přidání deníku. Po jeho kliknutí se objeví formulář pro přidání. Uživatel vyplní informace a formulář odešle.

Upravení provozního deníku

Uživatel v menu klikne na TNS a vybere „Provozní deníky“. Aplikace zobrazí seznam provozních deníků. Uživatel si najde, který deník chce upravit a tlačítkem pro upravení na konci řádku tabulky otevře formulář. Informace ve formuláři upraví a odešle.

Export pro firmu

Uživatel klikne na dlaždici kategorie TNS. Aplikace zobrazí seznam s nadcházejícími revizemi. Na levé straně si uživatel vyfiltruje firmu pomocí tlačítka s názvem firmy. Následně klikne na tlačítko „Export“. Aplikace export vygeneruje a nabídne ke stažení.

4.2.3 Správa firem

Zobrazení kontaktů

Uživatel klikne na dlaždici „Kontakty firem“. Aplikace zobrazí seznam s kontakty na firmy.

Přidání kontaktu

Uživatel klikne na dlaždici „Kontakty firem“. Aplikace zobrazí seznam s kontakty na firmy. V hlavičce, v pravém rohu klikne na tlačítko pro přidání kontaktu. Objeví se formulář, který vyplní a odešle.

Upravení kontaktu

Uživatel klikne na dlaždici „Kontakty firem“. Aplikace zobrazí seznam s kontakty na firmy. Uživatel si najde, který kontakt chce upravit a tlačítkem pro upravení na konci řádku tabulky otevře formulář. Informace ve formuláři upraví a odešle.

4.2.4 Správa středisek

Zobrazení středisek

Uživatel v menu klikne na Extra a vybere „KST Kontakty“. Aplikace zobrazí seznam kontaktů na mistry na střediscích.

Přidání kontaktu

Uživatel v menu klikne na Extra a vybere „KST Kontakty“. Aplikace zobrazí seznam kontaktů na mistry na střediscích. V hlavičce, v pravém rohu klikne na tlačítko pro přidání kontaktu. Objeví se formulář, který vyplní a odešle.

Upravení kontaktu

Uživatel v menu klikne na Extra a vybere „KST Kontakty“. Aplikace zobrazí seznam kontaktů na mistry na střediscích. Uživatel si najde, který kontakt chce upravit a tlačítkem pro upravení na konci řádku tabulky otevře formulář. Informace ve formuláři upraví a odešle.

4.2.5 Správa nedostatků

Zobrazení nesplněných nedostatků

Uživatel klikne na dlaždici „Nedostatky“. Aplikace zobrazí seznam nesplněných nedostatků.

Zobrazení všech nedostatků

Uživatel klikne na dlaždici „Nedostatky“. Aplikace zobrazí seznam nesplněných nedostatků. Uživatel klikne na tlačítka „Filtrovat vše“ v pravém horním rohu. Aplikace zobrazí seznam všech nedostatků.

Přidání nedostatku

Uživatel klikne na dlaždici „Nedostatky“. Aplikace zobrazí seznam nesplněných nedostatků. V hlavičce, v pravém rohu klikne na tlačítko pro přidání nedostatku. Objeví se formulář, který vyplní a odešle.

Upravení nedostatku

Uživatel klikne na dlaždici „Nedostatky“. Aplikace zobrazí seznam nesplněných nedostatků. Uživatel si najde, který nedostatek chce upravit a tlačítkem pro upravení na konci řádku tabulky otevře formulář. Informace ve formuláři upraví a odešle.

4.2.6 Ostatní

Přístup do aplikace

Uživatel otevře aplikaci. Pokud uživatel nemá oprávnění přístupu, zobrazí se stránka se zamítnutým přístupem.

Rychlý přehled

Uživatel vidí na úvodní stránce počty nadcházejících revizí a počty propadlých revizí pro jednotlivé kategorie. Je vidět i počet otevřených nedostatků.

Zobrazení všech revizí na stroji

Uživatel klikne na dlaždici „Stroj“. Do vyhledávacího pole zadá číslo stroje, který hledá. Aplikace zobrazí všechny revize na stroji prováděné.

Zobrazení protokolu revize

Uživatel klikne na dlaždici „Stroj“. Do vyhledávacího pole zadá číslo stroje, který hledá. Aplikace zobrazí všechny revize na stroji prováděné. Uživatel klikne na odkaz v revizi, u které chce vidět protokol. Aplikace protokol nabídne ke stažení.

Zobrazení provedených změn

Uživatel v menu klikne na Extra a vybere „Log“. Aplikace zobrazí seznam provedených změn.

Změna oprávnění přístupu

Správce aplikace v databázi změní nebo přidá roli uživateli.

4.2.7 Pokrytí Funkčních požadavků

Pro shrnutí případů užití přikládám tabulky pokrytí funkčních požadavků 4.1 a 4.2.

Případy užití	F1	F2	F3	F4	F5	F6	F7	F8
Správa zařízení			X		X	X		
Tlakové lahve								
Správa firem								
Správa středisek								
Správa nedostatků								X
Ostatní	X	X		X			X	

■ **Tabulka 4.1** Pokrytí funkčních požadavků případy užití - část 1

Případy užití	F9	F10	F11	F12	F13	F14	F15
Správa zařízení				X		X	
Tlakové lahve					X		
Správa firem		X					
Správa středisek			X				
Správa nedostatků	X						
Ostatní							X

■ **Tabulka 4.2** Pokrytí funkčních požadavků případy užití - část 2

4.3 Uživatelské rozhraní

Uživatelské rozhraní je prostředník, který zprostředkovává komunikaci mezi uživatelem a aplikací nebo počítačem. Jeho vzhled, funkčnost, použitelnost nebo nepoužitelnost ovlivňují, zda aplikaci uživatel rád používá, vrátí se k aplikaci nebo ne. Uživatelské rozhraní je tedy nedílnou součástí aplikace. Ať už je aplikace sebelepší, maximálně optimální a ideální v operacích prováděných na pozadí, špatné rozhraní může uživatele odradit natolik, že aplikaci ani nevyzkouší.

Návrh uživatelského rozhraní se z pohledu nezaláhaného uživatele může zdát jako jednoduchá záležitost, opak je ale pravdou. Správné navržení a vytvoření uživatelského rozhraní je často velmi složitou částí vývoje. Rozhraní musí být intuitivní, jednoduché na používání, jednoznačné, konzistentní, efektivní a v dnešní době téměř ve všech případech responzivní. Je potřeba i myslet na pokročilé uživatele, kteří například rádi využívají více klávesových zkratk než myši. Další jsou skupiny lidí s postižením nebo barvoslepostí, i pro tyto uživatele je někdy potřeba rozhraní uzpůsobit. Tohle všechno je potřeba jenom k návrhu rozmístění prvků, nesmíme zapomenout na samostatný design.

Grafický design uživatelského rozhraní jde brát jako samostatnou část při návrhu. Existují různá doporučení jaké kombinace barev používat, jaké fonty zvolit, způsob rozložení textu (dlouhé řádky nebo třeba 2 sloupce). Grafika zároveň nesmí být moc výrazná ani nevýrazná. Musí uživatele upoutat, aby aplikaci uživatel používal déle nebo se k ní vracel. Jakým způsobem upozorníme uživatele na důležité informace, a jak udržíme jeho pozornost na relevantních informacích.

4.3.1 Druhy prototypů

Při návrhu uživatelského rozhraní se začíná vytvořením prototypů. Existují 2 typy, low-fidelity (lo-fi) a high-fidelity (hi-fi). Mezi lo-fi prototypy patří papírové modely, neboli wireframe. Wi-

reframe definuje textový i grafický obsah, rozmístění funkčních prvků a také navigaci či tlačítka. Grafický obsah je tvořen pouze pomocí čar a nepoužívají se ani barvy. Papírový model následně umožňuje konzultovat návrh uživatelského rozhraní se zadavatelem před samotnou implementací, která je časově náročná na změny. Provádění změn v papírovém modelu je jednoduché a rychlé, zároveň po vytvoření a otestování modelu zadavatelem se může grafik plně soustředit na svoji práci a nemusí řešit rozmístění prvků a obsah stránky. [27]

Hi-fi prototypy se liší od lo-fi zejména tím, že jejich obrazové prvky, popisky, grafika a interakce obsahují velké množství detailů a svým vzhledem a chováním takřka odpovídají finální aplikaci. Může to pak vypadat, že hi-fi prototypy jsou mnohem lepší, ale to není úplně pravda. Lo-fi prototypy mohou být v mnoha případech výhodnější. Hi-fi prototypy jsou nákladnější na výrobu, zadavatel si může myslet, že návrh co vidí je už samotná hotová aplikace a pak se diví, že výsledek není úplně stejný. Proto se začíná lo-fi prototypem, který projde několika iteracemi a až po odladění wireframu se v případě potřeby vytváří hi-fi prototyp. Ten ale není potřeba vždy a z lo-fi prototypu se může rovnou přejít k implementaci.

4.4 Návrh prototypu

Pro tvorbu prototypu jsem využil nástroj Figma [28]. Jedná se o nástroj pro rychlou tvorbu prototypů GUI, bez nutnosti psaní programového kódu. Každý jednotlivý návrh jsem iterativně konzultoval (testoval se zadavatelem) a následně pozměnil. Celkem proběhly zhruba 3 iterace, ve kterých jsem našel chybu, nebo jsem byl upozorněn na nějaký nedostatek. Každou iterací se návrh zlepšoval a přibližoval řešení, které už bylo vhodné implementovat.

Prvním krokem návrhu bylo navrhnout, jak jednoduše a podobně převést uživatelské rozhraní ze staré aplikace do webového prostředí. Jelikož jedním z požadavků, zadaných zadavatelem bylo zachování základního vzhledu a umístění hlavních funkcí za účelem rychlé adaptability uživatelů na nové rozhraní.

Celý návrh je dostupný na zkráceném odkaze rb.gy/60igj7 (originální odkaz je Zde, dostupný pouze v digitální verzi). Celý link uvádím pro případ, že by zkrácený link přestal fungovat.

4.4.1 Úvodní stránka

Úvodní stránku jsem se rozhodl téměř celou zachovat. Jedná se o stránku, na které jsou vidět dlaždice jednotlivých kategorií, které obsahují odkaz na stránku seznamu nadcházejících revizí, odkaz na formulář pro jednotlivé zařízení a to hlavní, přehlednou informaci o počtu nadcházejících a propadlých revizí. V druhé iteraci jsem navíc zavedl odkazy na kategorie i do horního menu aplikace, aby se uživatelé nemuseli vždy vracet na úvodní stránku.

4.4.2 Stránka kategorií

Ve staré aplikaci se jednalo o dva pohledy. Seznam nadcházejících revizí a formulář pro úpravu a zadávání zařízení. Tyto dva pohledy zachovám, ale provedu menší změny. V aplikaci nebylo možné se ze seznamu přepnout do formuláře a naopak. Musel se prvně formulář nebo seznam zavřít a následně otevřít to druhé. Formulář byl použitelný, až na chyby prováděné při vyhledávání zařízení přes CTRL+F, které už jsem zmiňoval. Seznam nadcházejících revizí se zobrazoval pouze jako výsledek dotazu v tabulce, bez jakéhokoliv formátování, až na kategorii tlakových lahví, která seznam ukazovala rovnou v tisknutelném formátu potřebného přímo pro firmu. Na obrázku 4.1 jsou vidět hlavičky seznamů kategorií. Z obrázku je i vidět jakým způsobem vypadají ve staré aplikaci seznamy nadcházejících revizí (obyčejné MS Access tabulky).



■ **Obrázek 4.1** Seznamy vlastností jednotlivých kategorií

V prototypu jsem tedy připravil návrh, který bude jednotný pro všechny kategorie. Jedná se o dvě stránky. Stránku s formulářem, kde je možné vyhledávat podle zařízení nebo stroje. Je na ní také možné zadávat nová zařízení a zobrazuje i informace o umístění stroje a informace o mistrovi, který je za stroj zodpovědný. Pro možnost editace jsem zvolil tlačítko, které bude sloužit pro odemčení možnosti editace polí ve formuláři. Editace data poslední revize bude možná kdykoliv. Pro uložení změn je nutné změny uložit odesláním formuláře. Ukázka návrhu je na obrázku 4.2.

Kategorie: Formulář

Vyhledávání podle ID

Vyhledávání podle Stroje

Zařízení

ID

Umístění

Technické místo

Aktivní

Typ

Kontakt

Hlavní mistr

Datum

Nákladové středisko

Example

Popis tech. místa

Example

Popis tech. objektu

Poznámka

SAP status

■ **Obrázek 4.2** Návrh formuláře pro zadávání zařízení v otevřené kategorii

Druhá stránka obsahuje seznam nadcházejících revizí. Jedná se o jednoduchou tabulku, ve které bude možné nastavit formátování pro jednotlivé řádky, například zvýraznit propadlé revize, nebo ty co se mají provádět aktuální měsíc. Navíc bude v seznamu možnou rovnou upravit dané zařízení pomocí vyskakovacího formuláře. Tato funkce ve staré aplikaci chyběla, a v nové to urychlí práci, i když jen o několik hledání. Navíc bude možné i v tabulce po kliknutí na tlačítko „Zobrazit všechny“, možnost vidět všechny zařízení, která jsou v systému zavedená. Tato funkce nebyla ve staré aplikaci pro normální uživatele dostupná, přístup měl pouze správce.

Ve třetí iteraci jsem navíc přidal tlačítko, které umožní celou tabulku zkopírovat. Informace z tabulky se pak dají jednoduše použít pro objednávání revizí nebo další operace, které se nad daty uživatel rozhodne provádět. Ukázka stránky se seznamem, i když na ní není moc co vidět, je na obrázku 4.3.

Kategorie: seznam nadcházejících revizí							
Example	Example	Example	Example	Example	Example	Example	Example
Ex	Ex	Ex	Ex	Ex	Ex	Ex	Ex
Ex	Ex	Ex	Ex	Ex	Ex	Ex	Ex
Ex	Ex	Ex	Ex	Ex	Ex	Ex	Ex
Ex	Ex	Ex	Ex	Ex	Ex	Ex	Ex

■ **Obrázek 4.3** Ukázka stránky se seznamem nadcházejících revizí v otevřené kategorii

4.4.3 Stránka souhrnu stroje

Stránka se souhrnem informací ke stroji je asi největší vylepšení oproti staré verzi. Navíc už při navrhování této stránky se přišlo s nápady o rozšíření, kde by se mohli zobrazovat například náhradní díly, prováděné opravy a všechny další informace, které jsou pro stroj známé. V systému který firma používá tyto informace na jednom místě nejsou a některé informace se špatně hledají. To jsou ale informace mimo téma revizí, nápad do budoucna, nebo možná práce co mě čeká po dokončení nového systému na revize.

Ve staré aplikaci informace k revizím na stroji nebyli úplně jednoznačné. Pokud se na stroji vyskytovalo více než jedno zařízení, alespoň v jedné kategorii, tak byly informace o posledních revizích rozděleny na více stránek. Navíc bez otevření protokolu nebylo možné zjistit výrobní číslo zařízení, ani platnost revize. Stránka obsahovala kromě informací ke stroji pouze odkaz na protokol a datum poslední revize. Ukázka je vidět na obrázku 4.4. Všimněte si, v levém dolním rohu, že formulář obsahuje dva záznamy, pro zobrazení všech informací je tedy nutné se přesunout malým tlačítkem vedle počtu záznamů. Uživatel, který na to nebyl dříve upozorněn si toho pravděpodobně ani nevšimne.

V návrhu jsem tuto stránku navrhl tak, abych tyto nedostatky odstranil. Po příchodu na stránku uvidí uživatel vyhledávací pole pro číslo stroje. Po zadání stroje pomocí výběrového pole s možností vyhledávání ze seznamu strojů se zobrazí informace o stroji a dynamicky se objeví políčka s revizemi zařízení. Jednotlivé pole budou obsahovat kategorii, výrobní číslo a datum nebo data revizí, které zároveň budou sloužit jako odkaz pro stažení protokolu revize. Zde jsem v návrhu zapomněl místo data revize uvést datum platnosti, ale to při implementaci už byla jen maličkost na úpravu. Ukázka návrhu je vidět na obrázku 4.5. Místo „Zařízení“ budou jednotlivá pole obsahovat název kategorie, například TNS, SHZ, Freon atd. Navíc jsem se později rozhodl i zvolit různé podbarvení pro kategorie, které přidá další úroveň přehlednosti.

frm_posledni_revize			
Revize prováděné na zařízení			
Technické místo:		Nákladové středisko:	
Popis technického místa:		Popis technického objektu:	
Číslo vybavení:		Status zařízení v SAP:	
Provozní revize TNS:	11.11.2018	Revize F-Plyny:	Neprovádí se
Vnitřní revize TNS	20.11.2017	Revize "C" praček:	Neprovádí se
Tlaková revize TNS	20.11.2017	Revize Draeger:	19.09.2022
Revize SHZ:	Neprovádí se	Revize UPS:	Neprovádí se
Revize plyn. zařízení:	02.12.2022	Revize TEM zařízení	25.11.2022

Record: 1 of 2 | No Filter | Search

■ Obrázek 4.4 Souhrn revizí pro stroj ve staré aplikaci

Revize prováděné na stroji

Vyhledání podle čísla vybavení

Vyhledání podle čísla vybavení

Číslo vybavení

Číslo vybavení

SAP status

SAP status

Popis tech. objektu

Popis tech. objektu

Technické místo

Technické místo

Popis tech. místa

Popis tech. místa

Nákladové středisko

Nákladové středisko

Zařízení

Zařízení

Provedení revize

Provedená revize

Zařízení

Zařízení

Provozní revize

Provozní revize

Vnitřní revize

Vnitřní revize

Tlaková revize

Tlaková revize

Zařízení

Zařízení

Provedení revize

Provedená revize

Zařízení

Zařízení

Provozní revize

Provozní revize

Vnitřní revize

Vnitřní revize

Tlaková revize

Tlaková revize

Zařízení

Zařízení

Provedení revize

Provedená revize

Zařízení

Zařízení

Provozní revize

Provozní revize

Vnitřní revize

Vnitřní revize

Tlaková revize

Tlaková revize

■ Obrázek 4.5 Návrh stránky pro souhrn revizí na stroji.

4.4.4 Tlakové nádoby

Kategorie tlakových nádob je odlišná od ostatních kategorií, navíc obsahuje provozní deníky, umístění v šanonu a seznam opravovaných protokolů. Seznam opravovaných protokolů je opravdu jenom seznam zařízení, která mají zaškrtnuté pole „Oprava reviz. dokladu“, nenavrhol jsem

tedy nic složitého. Tlakových nádob je hodně a tak je potřeba i seznam nadcházejících revizí rozdělit podle firem, které revize dělají. Pro jednu z firem je potřeba export pro tisk ve formátu, který musí být identický, nebo co nejvíce podobný jako ve staré aplikaci. Ten obsahuje veškeré potřebné informace k lahvi (závod, středisko, stroj, označení tlakové nádoby, termíny revizí, poznámky atd.), spolu se zvýrazněnými políčky nadcházejících revizí (u tlakových lahví se provádí 3 druhy revizí), pod kterými je místo pro zapsání naměřených hodnot tlaku a teploty. Export je vidět na obrázku 4.6.

4.4.4.1 Export pro tisk

Ve staré aplikaci je export vytvářený pomocí sestavy, v rámci návrhu jsem připravil šablonu v MS Excel, která vypadá téměř identicky. Z této šablony budu vycházet při implementaci. Na obrázku 4.6 je vidět export ze staré aplikace a vytvořené šablony. Sudé a liché řádky se liší v odstínu barev pozadí pro lepší čitelnost. Obrázek je zde málo čitelný, slouží pouze pro ukázkou, při tisku se totiž používá rozložení stránky na šířku.

Závod	Nákl.Středisko	Popis technického místa	Označení technického objektu	Inv.č. stroje	Číslo tlakové nádoby	Provozní R.	Vnitřní R.	Tlaková R.	OK
1	515504	Stanice QMM Z2 - W1	Burst test	J220196	763595/2016	05.01.2024	07.01.2026	20.04.2026	<input type="checkbox"/>
Toto zařízení je na závodě 1_QMM. Kontakt: [redacted]									
1	123456 2014	Rotorwelle	Soustruh	J223506	6569/2021	23.1.2024	23.1.2026	23.1.2031	<input type="checkbox"/>
010103	John Doe	123456789	Pozor, změna inventárního čísla. Původní J223490.						

■ **Obrázek 4.6** Export pro tlakové lahve. Nahoře: stará aplikace. Dole: nová šablona

Spolu s exportem pro tisk je pro tlakové lahve potřeba i další druh exportu, který pouze obsahuje číslo stroje a výrobní číslo zařízení odděleně pro jednotlivé druhy revizí které se provádí příští měsíc, s těmi daty se pak dále pracuje přednastavenými způsoby v excelu. Exporty bude možné vygenerovat po kliknutí na tlačítka „Export PDF“ a „Karbon Export“, které se budou nacházet na levé straně nad seznamem revizí.

4.4.4.2 Umístění do šanonu

Vytváření možných umístění do šanonu se ve staré aplikaci dělalo přímo v tabulce, ve které je už připraveno velké množství umístění, takže se málokdy přidávalo nové. Samotné přiřazení umístění k danému zařízení se provádí ve formuláři a na jednom místě může být uloženo pouze jedno zařízení. V nové aplikaci jsem se pro zadávání umístění v šanonu rozhodl použít select box, který už bude mít omezené možnosti, jenom na ty volné. Vedle boxu bude tlačítko pro případné vytvoření nového umístění šanonu. Samotné vytvoření umístění není nic jiného, než jednoduchý vyskakovací formulář s jedním zadávacím polem pro označení umístění.

4.4.4.3 Provozní deníky

Funkce provozních deníků se v aplikaci používá jen zřídka. Při potřebě zavedení nového deníku se uživatel zodpovědný za tlakové lahve vždy obracel na správce aplikace s požadavkem na vytvoření deníku a zapsání zařízení do nově vytvořeného deníku. Pro zjištění seznamu všech deníků bylo potřeba otevřít přímo tabulku, ve které jsou definované a nebyla zde možnost zobrazit si všechna zařízení vyskytující se v deníku. Jediný způsob jak tohoto dosáhnout bylo pomocí hledání ve formuláři tlakových lahví v poli pro zařazení deníku.

Navrhl jsem stránku, na které bude možné zobrazit všechny deníky, upravit jejich vlastnosti a vytvořit nový. Navíc je v seznamu možné „rozbalit“ deník a vidět v něm všechna zařízení, která obsahuje. Samotné přiřazení zařízení do deníku jsem zachoval jako ve staré aplikaci, je možné pouze z formuláře daného zařízení. Je to především z důvodu, že člověk zodpovědný za tlakové lahve je zvyklý si navíc napsat do poznámky, v jakém deníku je zařízení vedeno a navíc se deníky

vytváří velmi málo (zhruba dva ročně). Ukázka návrhu pro lepší představu je vidět na obrázku 4.7. Názvy sloupců a jejich obsah bude vyplněn při implementaci.

Example	Example	Example	Example	Example	Example	Example	Example
Ex	Ex	Ex	Ex	Ex	Ex	Ex	Ex
Example	Example	Example	Example	Example	Example	Example	Example
Ex	Ex	Ex	Ex	Ex	Ex	Ex	Ex
Ex	Ex	Ex	Ex	Ex	Ex	Ex	Ex
Ex	Ex	Ex	Ex	Ex	Ex	Ex	Ex

■ **Obrázek 4.7** Návrh seznamu pro provozní deníky tlakových lahví.

4.4.5 Nedostatky a kontakty na externí firmy

Na stránce nedostatků a kontaktů na externí firmy jde pouze o seznam s možností editace a přidání nového záznamu. Nenavrhoval jsem nic složitého, na stránce bude seznam s tlačítkem pro editaci na konci řádku, tlačítko pro přidání bude v hlavičce tabulky a pro formuláře použiji vyskakovací formulář. Názvy sloupců a polí ve formulářích doplním až při implementaci podle staré aplikace.

4.4.6 Kontakty na střediska

Stránka s „Kontakty“ má zobrazovat informace o tom, jaký mistr je zodpovědný za která střediska, spolu s informacemi o hale a telefonu na mistra. Ve staré aplikaci se jedná o jeden seznam. Jsou v něm jména a telefony psaná duplicitně, při změně telefonu na mistra, nebo nahrazení mistra jiným je tedy nutné upravit záznamy na všech místech, kde se mistr vyskytuje. Z důvodu ponechání funkčnosti staré aplikace po nějakou dobu jsem se rozhodl použít stejné rozložení jako na stránce s kontakty na externí firmy.

Po odstavení staré aplikace seznam rozdělím na dvě části. Seznam, kde se bude ke středisku přidělovat mistr z výběrového seznamu a druhý seznam, v kterém budou mistři a jejich telefony. Navíc oproti ostatním seznamům, v těchto dvou půjdou i záznamy mazat. Pro mazání se použije tlačítko s ikonou koše a po jeho stisknutí na uživatele vyskočí potvrzovací okno, v kterém buď uvidí důvod, proč záznam nejde odstranit, nebo možnosti s potvrzením a zrušením. V retrospektu po implementaci, jsem měl rovnou použít dvě tabulky a práci s daty do staré aplikace vyřešit na backendu.

4.4.7 Log provedených změn

Stránka s historií změn bude velmi jednoduchá, bude pouze obsahovat tabulku s možností filtrace a řazení sloupců. Bude obsahovat sloupce uživatel, tabulka, typ změny a provedené změny. Typem změny mohou být vytvoření, změna a smazání, z názvu tabulky uživatel jednoduše pozná o které kategorii se jedná a ve sloupci provedených změn uvidí všechny informace o tom co se provedlo. Při úpravě budou viditelné data před i po změně pro možnost vrácení změn.

4.4.8 Kalendář

Kalendář s revizemi je nástroj, který se pro plánování revizí doslova hodí. Pokud firma nemá moc zařízení tak bude i přehledný. Pro vysokou použitelnost bych očekával možnosti se rovnou z kalendáře přesunout na formuláře zařízení a posouvat je na dřívější termíny. Různé kategorie

revizí by měli být barevně odlišné, přidal bych možnost zobrazení jen vybraných kategorií a možná i další funkce, které se mohou lišit v jednotlivých firmách.

Pro firmu, pro kterou aplikaci vytvářím, z analýzy vyšel závěr, že kalendář nemá smysl. Většina revizí platí do konce měsíce, ve který má termín, takže by kalendář jen obsahoval měsíce, a v každém měsíci by byl obrovský seznam zařízení. Revize zajišťují externí firmy, takže přesný termín, kdy revizi provedou je na nich. Zjišťoval jsem, jestli by uživatelé zodpovědní za správu revizí chtěli vidět kolik revizí mají každý měsíc, ale informace se ukázala jako zbytečná, počet nadcházejících je bohatě dostačující údaj.

V případě, že by se v budoucnu nebo v nově přidávaných kategoriích po dokončení práce něco v tomto smyslu přidávalo bych očekával například graf s počty pro jednotlivé roky, který bude vizuálně velmi přehledně počty zobrazovat. Posuny termínů se zatím nikde nepoužívají, ale i v případě, že by to bylo potřeba, nebude se jednat o velké množství posunů. Po prvním nastavení termínů se už budou generovat nadcházející termíny automaticky. Posuny by tedy stačily spravovat přímo ve formuláři zařízení.

4.5 Databáze

Struktura databáze je v aktuálním řešení v podstatě navržena dobře. Každá kategorie má svojí tabulku, pouze není použité jednotné pojmenování sloupců mezi kategoriemi. Primární klíče jsou správně definované v každé tabulce a vazby mezi tabulkami databáze také obsahuje.

Pouze pro zaznamenávání změn je v databázi zbytečně logovací tabulka pro každou kategorii zvlášť. Ty v novém řešení nahradím jednou tabulkou pro změny. Bude obsahovat sloupce s přihlašovacími údaji uživatele, názvem tabulky, ve které se změna provedla, typ provedené změny (vytvoření, upravení, smazání) a samotná změna. V případě upravení bude změna obsahovat, jak informace před změnou, tak i po ní.

Rozhodl jsem se databázi přenést stejně jako ze staré aplikace, což umožní jednoduché umožnění chodu obou aplikací naráz, kterou si při konzultaci ohledně databáze zadavatel vyžádal, jelikož nejsou potřeba zásadní změny ve struktuře. Pro novou aplikaci pouze připravím pohledy, které budou mít jednotné pojmenování sloupců. Po zrušení staré aplikace pak nebude problém sloupce přejmenovat přímo v tabulkách samotných.

4.6 Shrnutí

Navrhoval jsem uživatelské prostředí na základě už existující, dlouze používané aplikace. V analýze jsem vyhodnotil, že je potřeba přidat některé funkce pro urychlení práce a na ty jsem se v návrhu zaměřil. Seznamy nadcházejících revizí s vyskakovacím formulářem a samotný formulář, který je potřeba odemknout pro umožnění změn, jsou hlavní části nového rozhraní. Celkový vzhled bude přívětivější uživateli a odstranil jsem potřebu používání funkcí vestavěných v MS Access. Největší změna vzhledu je na stránce pro souhrn stroje, která bude oproti starému mnohem přehlednější a jednodušší pro vyhledávání. Bonus je i stránka s provozními deníky tlakových lahví.

Do přílohy B jsem vložil obrázky návrhů úvodní stránky B.1, vyskakovacího formuláře pro zařízení B.2, který se otevírá ze seznamu a stránku s formulářem pro tlakové nádoby B.3. Jak už jsem zmiňoval na začátku kapitoly 4.4, celý návrh je dostupný na zkráceném odkaze rb.gy/60igj7.

V této kapitole se budu věnovat postupu vývoje nové aplikace na základě vytvořeného návrhu, který jsem popsal v předchozí kapitole. Nejprve představím framework Nette, na kterém bude aplikace postavená. Důvodem k výběru Nette je nefunkční požadavek 2 - kapitola 3.2.3.2 (Aplikace bude používat technologie podporované společností - Nette). Při vývoji software, kde není přímo určená technologie by se zde programátor rozhodoval, co za technologii použít.

5.1 Nette

Nette je kompletní Framework pro jazyk PHP, který výrazně zjednodušuje tvorbu webových aplikací. Je založen na samostatných knihovnách, které dohromady tvoří celý framework. Jedním z hlavních zaměření Nette je vysoká míra zabezpečení pomocí technologií eliminujících bezpečnostní mezery. Nette je postavené na architektuře MVC. Autorem je český vývojář David Grudl. [29]

5.1.1 Architektura Nette

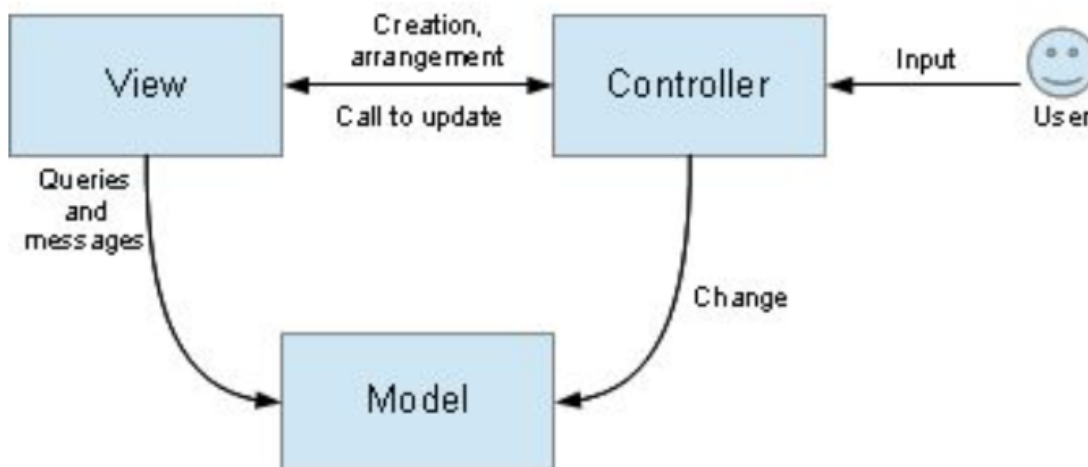
Model-View-Controller je softwarová architektura, která dělí aplikaci na 3 logické části, aby šli samostatně upravovat a dopad změn byl na ostatní části co nejmenší. Tyto části jsou Model, View a Controller.[30]

- Model - Reprezentuje data a business logiku aplikace. Jakákoliv akce uživatele (přihlášení, změna hodnoty v databázi, zobrazení stránky) představuje akci modelu. Model si spravuje svůj vnitřní stav a ostatním částem aplikace nabízí pevně dané rozhraní. Model o existenci view nebo kontroleru neví.
- View - Vrstva aplikace, která má na starost zobrazení výsledku požadavku. Obvykle používá šablonovací systém a ví jak zobrazit jednotlivé komponenty nebo výsledky získané z modelu.
- Controller - Zpracovává požadavky uživatele a na jejich základě volá patřičný model. Poté požádá view o vykreslení dat. V Nette Framework jsou obdobou kontrolerů presentery.[31]

Komunikace mezi jednotlivými částmi je znázorněna na obrázku 5.1.

5.1.2 Další technologie

Klasická webová stránka se skládá z několika součástí. Základem je HTML, představující kostru stránky včetně obsahu, a CSS, starající se o vzhled a grafickou podobu. Často bývá přítomný



■ Obrázek 5.1 Architektura MVC [30]

JavaScript, ať už jen pro analytické nástroje, nebo pro dodání dynamiky uživatelského rozhraní. V Nette jsou stránky generované šablonovacím systémem, ten vygeneruje HTML a následně je možné dynamicky překreslovat části stránky, nebo stránku celou.

5.1.2.1 HTML

HTML je jazyk popisující strukturu webových stránek [32]. Umožňuje strukturovat dokument do nadpisů a odstavců, vkládat tabulky, obrázky a jiná multimédia přímo do dokumentu, vytvářet formuláře pro práci s daty a také vzájemně propojovat dokumenty pomocí hypertextových odkazů. Struktura stránky je popsána značkami, které představují jednotlivé elementy (např. <table>, nebo <button>). Objektově orientovaná reprezentace struktury dokumentu je označována jako rozhraní DOM.

5.1.2.2 CSS

Pro popis vzhledu stránky slouží CSS [32]. Jednotlivým elementům přiřazuje styly, které tvoří atributy jako barva, velikost, font a spousta dalších. Umí také základní animace při interakci uživatele (např. při najetí kurzorem myši na element). Dokáže navrhnout rozvržení stránky pro malá mobilní zařízení, velké desktopové obrazovky nebo tiskárny. Oddělení HTML od CSS umožňuje jednodušší správu stránek a také sdílení stylů mezi více stránkami.

5.1.2.3 JavaScript

JavaScript je programovací jazyk, který se běžně používá ve webovém prostředí pro vylepšení interaktivity a dynamiky webových stránek. Dnes je JavaScript podporován všemi moderními webovými prohlížeči a umožňuje tvorbu interaktivních prvků, jako jsou animace, interaktivní formuláře, aktualizace obsahu stránky bez nutnosti načítání nové stránky (AJAX) a mnoho dalších funkcí. [33]

5.1.3 Databáze

Aplikace bude používat databázový server Microsoft SQL Server 2019. Tento databázový server se ve firmě používá, a obsahuje některá data, které aplikace bude potřebovat. Kromě dat už existuje

tujících na databázovém serveru budu vytvářet samotnou databázi pro aplikaci. Pro zachování funkčnosti starého systému se zpočátku struktura databáze zachová ve stejném stylu, pouze s jednou logovací tabulkou.

5.1.4 Použité knihovny

V této části stručně popisuji některé podpůrné knihovny, které jsem se rozhodl použít a jejich využití v aplikaci.

[contributte/translation](#)

- Slouží k implementaci překladů v aplikaci. Poskytuje nástroje a strukturu pro efektivní práci s jazykovými lokalizacemi a textovými řetězci.
- Zvolena pro jednoduchou a efektivní správu překladů.
- Využita v celé aplikaci.

[ublaboo/datagrid](#)

- Datagrid je komponenta umožňující zobrazování tabulek s možností řazení a filtrování.
- Zvolena pro tabulku s možností řazení a filtrování nad větším množstvím dat.
- Využita pro tabulku s historií změn.

[nunnick/chartjs](#)

- Tato knihovna umožňuje vytváření interaktivních grafů pomocí HTML5 a JavaScriptu. Podporuje různé typy grafů a nabízí možnosti přizpůsobení vzhledu.
- Zvolena pro vizualizaci dat a jednoduchost nastavení.
- Využita pro stránku s grafem rozložení revizí na jednotlivé měsíce.

[select2/select2](#)

- Knihovna pro vytváření lepších select boxů s pokročilými funkcemi, včetně vyhledávání a možností více výběrů naráz.
- Zvolena pro zajištění přívětivějšího výběru ze seznamu.
- Využita pro většinu select boxů. Například vyhledávání podle zařízení a stroje.

[phpoffice/phpspreadsheet](#)

- Tato knihovna umožňuje čtení a vytváření tabulkových souborů v PHP. Podporuje různé formáty, včetně Excelu.
- Zvolena pro manipulaci s tabulkovými daty, čtení a generování souborů ve formátu Excel, což je užitečné pro export dat.
- Využita pro exporty tabulek ze seznamů a pro vytvoření správného formátu pro TNS export.

5.2 Vývoj

Aplikaci jsem vyvíjel takovým způsobem aby nebylo potřeba v aplikaci dělat změny při malé změně v databázi u existujících kategorií. Jak je potřeba při každém vývoji, vytvořil jsem si obecné komponenty, které se použijí na více místech. Pro každé téma jsem si připravil presenter, šablonu a model (v aplikaci nazvané jako repository.) Jelikož presentery a modely kategorií budou dost podobné, vytvořil jsem si abstraktní třídy, z kterých jednotlivé třídy dědí. Pokud se něco

liší je možné funkci přepsat, umožňuje to tedy vše co je potřeba. Na jednotlivých stránkách jsou používané AJAX volání pro obsluhu tlačítek a formulářů. AJAX umožňuje webovým stránkám komunikovat s webovým serverem bez nutnosti znovu načítání celé stránky. V kombinaci s Nette jde pak překreslit část stránky pomocí tzv. „snippetů“.

5.2.1 Vývojové prostředí

Pro implementaci jsem používal program Visual Studio Code. Je to bezplatný, open-source textový editor od společnosti Microsoft. Jedná se o chytrý editor kódu, poskytuje velké množství rozšíření pro podporu různých jazyků, včetně PHP, HTML, CSS a JavaScript. Pomáhá psát kód za pomoci kontextového doplňování kódu, detekcí chyb, kontrolou a opravami kódu. [34]

Pro verzování kódu jsem využil program SourceTree [35]. Sourcetree je zdarma dostupné grafické uživatelské rozhraní pro správu verzí, usnadňuje práci s distribuovaným verzovacím systémem Git. Git se používá ve většině softwarových firem, umožňuje sdílet kód mezi počítači a kooperovat v týmu více lidí. Kdykoliv se dá vrátit k původní stabilní verzi a vrátit nepovedené změny bez toho, aniž byste o svůj kód přišli. [36]

Pro lokální testování aplikace jsem využíval Apache HTTP Server. Jedná se o jeden z nej-používanějších webových serverů na světě. Apache poskytuje robustní a flexibilní prostředí pro hostování webových stránek a aplikací. [37]

5.2.2 Struktura aplikace

Struktura aplikace je standardní jako u většiny webových aplikací používajících Nette. Krátce zde popisují adresářovou strukturu projektu.

.docker	Soubory pro Docker
├─ app	
│ └─ Components	Komponenty aplikace
│ └─ lang	Soubory s překlady
│ └─ Model	Modely
│ └─ Presenters	Presentery
│ └─ templates	Šablony/view
├─ config	Konfigurační soubory
├─ log	Chybové logy
├─ temp	Dočasné soubory
├─ vendor	Knihovny aplikace
├─ www	Kořenový adresář webu
└─ readme.md	Instrukce pro spuštění a popis

5.2.3 Názvy tříd - URL

V aplikaci v názvech tříd míchám český a anglický jazyk. Důvodem jsou zobrazující se URL, které jsou v Nette odvozené od názvu presenterů. V URL lépe vypadají české názvy jako „Plyn, Uhlovodík, Odstředivky“, kde s anglickými názvy by se jednalo o „Gas, Hydrocarbon, Centrifuges“. Navíc aplikaci uživatelé používají v českém jazyce a v případě potřeby opravy kódu stačí vědět o jakou kategorii se jedná a nemusím přemýšlet, jak je název v překladu. Především je to dobré pokud by úpravy v kódu dělal někdo, kdo kód nezná.

5.2.4 Modely

Modely jsou v mém případě „repositáře“, mám vytvořenou abstraktní třídu „KategorieAbstractRepository“, která obsahuje pomocné funkce pro konvertování dat získaných z databáze.

Konkrétně se jedná o převod formátu data a boolean proměnné, které databáze vrací jako '0' nebo '1'. Dále obsahuje funkce pro získání mistra a umístění z databáze podle čísla zařízení.

Abstraktní funkce, které musí mít všechny další třídy jsou funkce zajišťující získání všech zařízení, nadcházejících revizí, všech strojů v kategorii, získání všech zařízení na základě čísla stroje, funkce pro získání zařízení podle čísla, vytvoření a upravení zařízení.

Při vytváření a upravování dat zařízení se navíc používá třída „LogRepository“ pro zápis veškerých provedených změn.

5.2.4.1 Souhrn stroje

V třídě „StrojRepository“ jsou funkce pro získání všech strojů, stroje podle čísla a funkce pro získání všech prováděných revizí na daném stroji. Ty jsou pak v presenteru předávány komponentě revizí.

5.2.4.2 Kontakty a nedostatky

V třídách obsluhující kontakty a nedostatky se opět jedná o funkce pro získání, vytvoření a upravení. Nedostatky navíc obsahují funkce pro získání pouze nesplněných nedostatků a funkci pro získání nedostatků vyfiltrovaných pomocí pole kategorií.

5.2.4.3 Ověření přístupu

Pro ověření přístupu se používá třída „RolesRepository“, která zajišťuje logování přístupu do aplikace a z databáze zjišťuje, jestli má uživatel na stránku přístup. V databázi jsou přístupové role vyřešeny jednoduchou tabulkou, která obsahuje uživatele, projekt (skupina stránek) a úroveň oprávnění (user/admin). Je to řešeno takto jednoduše, jelikož se aplikace nachází na vnitřní síti firmy.

5.2.5 Komponenty

Vlastní komponenty jsem si do aplikace vytvořil čtyři. Jedná se o komponenty používané na více místech, zejména v různých kategoriích, nebo pro vykreslování více stejných prvků.

5.2.5.1 Formulář

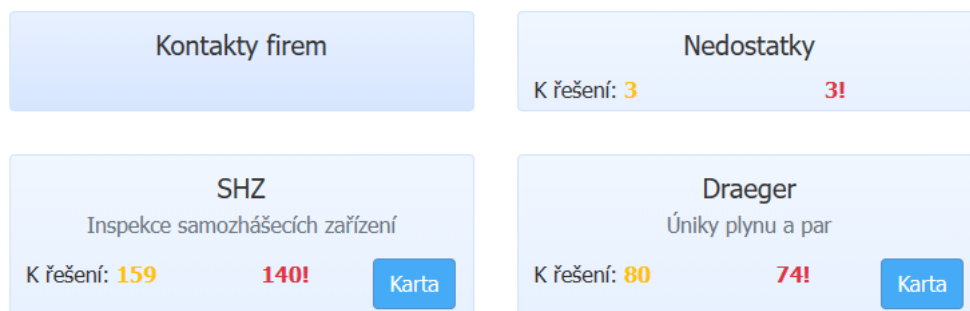
Komponenta s vlastním způsobem vykreslení formuláře, za použití vestavěných formulářů z nette. Komponenta je nazvaná jako „CustomFormComponent“. Obsahuje samotné vytvoření komponenty v konstruktoru za pomocí třídy FormLabelTranslator. FormLabelTranslator je singleton třída, která pouze zajišťuje vytvoření správného popisu a typu pole na základě názvu sloupce z databáze. Dále obsahuje funkce pro nastavení hodnot formuláře a jeho vyprázdnění. Šablona komponenty zprostředkovává vykreslení formuláře do dvou sloupců spolu s potřebnými styly. Je zde také ošetřeno zamčení změn a možnost odemčení pomocí JavaScriptu.

5.2.5.2 Tabulka

Komponenta „TableComponent“ používaná pro vykreslování tabulek nadcházejících revizí v kategoriích. Samotná třída má pouze funkci pro renderování. Vstupním parametrem je celý obsah tabulky. V šabloně je vykreslování tabulky spolu s logikou podbarvování dat. Tabulka obsahuje odkazy na otevření formuláře zařízení pro jeho editaci.

5.2.5.3 Karta kategorie

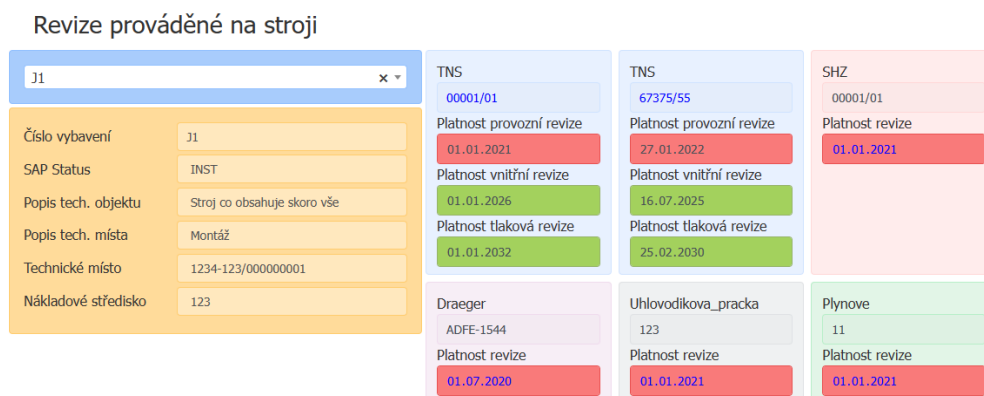
Nazvaná „KategorieComponent“ je komponenta zajišťující vykreslení jedné dlaždice na úvodní stránku. Vykreslování je podmíněné několika parametry, které jsou při vytvoření komponenty předány. Na jejich základě je potom dlaždice vykreslena. Pro představu výsledných dlaždic zde přidávám obrázek 5.2.



■ **Obrázek 5.2** Ukázka výsledného vzhledu dlaždic z úvodní stránky aplikace

5.2.5.4 Karta revize

Komponenta „RevizeComponent“ zajišťuje vykreslování jedné revize na stránce stroje. Podle parametrů je nastavená barva a informace k zobrazení. Barva slouží pouze k vizuálnímu oddělení různých zařízení. Některé kategorie mají dva až tři termíny revizí, to je zde taky vyřešeno. Pro představu výsledné stránky stroje zde přidávám obrázek 5.3.



■ **Obrázek 5.3** Stránka stroje z výsledné aplikace

5.2.6 Presentery

Pro každou stránku mám vytvořený presenter. Mám dvě abstraktní třídy, jednu pro presentery kategorií („KategorieAbstractPresenter“) a druhou pro ostatní stránky jako jsou nedostatky, kontakty atd. („AbstractPresenter“).

AbstractPresenter obsahuje pouze funkce pro ověření uživatele a získání jazyku prohlížeče. Presentery, které z něj dědí jsou od sebe rozličné. Vytvářejí se v nich potřebné komponenty a zajišťují jejich obsluhu, zejména vyvoláním potřebných operací jako překreslování snippetů a volání do modelů pro načtení/uložení dat do databáze.

Presentery kategorií jsou do velké míry stejné, veškeré obslužné funkce se nacházejí v abstraktním presenteru a pouze u kategorií, kde je něco navíc nebo se něčím liší, jsou funkce přidávány a přepisovány. Abstraktní presenter obsahuje funkce pro ověření uživatele, vytvoření komponent formuláře a tabulky, funkce ovládající tlačítka pro filtrace a exportování do Excelu. Překresluje části stránky podle potřeby a volá funkce modelu pro práci s daty. Příkladem odlišné kategorie je TNS, která má navíc dva speciální exporty.

5.2.6.1 Souhrn stroje

Presenter pro stránku stroje kromě renderovací funkce obsahuje funkce pro generování souborů protokolů ke stažení. Základní cesta je uložena v proměnné a upravuje se na základě kategorie, pro kterou má soubor vrátit. Samotné soubory protokolů nejsou na serveru, ale na sdíleném disku, není tedy možné se na ně odkazovat na přímo. Vyřešil jsem to způsobem, že se při každém požadavku soubor ze sdíleného disku zkopíruje do dočasného souboru na serveru a ten se použije pro vytvoření souboru ke stažení na požadavek uživatele.

5.2.6.2 Kontakty a nedostatky

Stránka s kontakty firem je pouze seznam s možností úpravy a vytváření. Stránka se středisky a kontakty na mistry obsahuje 2 seznamy, opět s možností vytvářet a upravovat, navíc je zde i možnost mazat.

Stránka nedostatků je seznam s možností filtrace všech, nebo jenom nesplněných nedostatků a je zde možnost filtrovat podle kategorií pomocí select boxu s možností výběru více záznamů.

5.2.7 Šablony

Šablonu, kterou bude presenter používat lze nastavit pomocí funkce `setView(„cesta_k_sablone“)`. Kategorie, které nejsou specifické používají šablony „`categoryTable.latte`“ pro tabulku a „`categoryForm.latte`“ pro formulář. V případě speciálních šablon stačí v presenteru přepsat funkci `render` a vytvořit vlastní šablonu.

Šablona stránky se seznamem revizí obsahuje tlačítka pro filtraci, kopírování a export, a samozřejmě samotnou tabulku vytvořenou komponentou. Šablona formuláře obsahuje pole pro vyhledávání podle zařízení a stroje. Samotný formulář s daty k zařízení a pod formulářem informace o umístění zařízení včetně mistra. Pro představu přidávám obrázek z výsledné aplikace 5.4.

5.3 Shrnutí


Celkově za dobu práce na projektu jsem provedl 339 commitů. Aplikaci jsem vytvářel dynamicky, hlavně v oblasti kategorií. Vytvořil jsem komponenty, které se využívají na několika místech. Pro zamezení duplikování kódu jsem vytvořil abstraktní presentery a repositáře a pokud je to možné využívám stejnou šablonu. Kód jsem řádně dokumentoval v průběhu implementace a pro případné scénáře, co by se mohli stát, jako je například přidání nové kategorie, jsem napsal potřebné kroky do „`readme`“ souboru.

Vyhledávání podle výrobního čísla Vyhledávání podle stroje

Zařízení

Výrobní č.

Typ

Datum poslední revize 



Perioda prohlídky

Umístění (stroj)

Firma

Poznámka

Aktivní

Uložit

Umístění

Technické místo

Nákladové středisko

Popis tech. místa

Popis tech. objektu

SAP Status

Kontakt

Hl. mistr

Telefon

Hala

■ **Obrázek 5.4** Ukázka formuláře z výsledné aplikace

Kapitola 6

Testování

Jedním z bodů zadání práce je otestování vytvořeného prototypu. Jedná se o proces kontroly kvality softwaru. Jeho podstata spočívá v ověření, že požadované funkce nastanou a nechtěné nenastanou. [38]

V případě této práce jde o ověření celkové funkčnosti aplikace, včetně otestování použitelnosti nového uživatelského rozhraní. Použitelnost vyjadřuje zejména snadnost použití a efektivnost. Výsledná aplikace používá v řadě věcí stejné komponenty jako staré řešení. Byly přidány nebo změněny některé funkce (editace ze seznamů, stránka stroje, kontakty na mistry atd.), u kterých je potřeba se přesvědčit, že půjdou intuitivně využít koncovými uživateli.

Z těchto důvodů proběhne uživatelský test. V této kapitole nejprve popíše teorii uživatelského testování. Dále zde bude popsána příprava, průběh a zhodnocení testování nové aplikace pro správu revizí.

6.1 Uživatelské testování

Uživatelské testování je jednou z nejčastějších metod testování použitelnosti aplikace. Tato metoda se zaměřuje na samotné chování potenciálních uživatelů, čímž lze jednoduše odhalit chyby co by vývojáře, které svůj vyvíjený systém znají, nenapadli. Uživatelské testování se dá aplikovat jak při fázi návrhu, například nad jednotlivými papírovými modely, tak i v rámci samotného vývoje, při kterém může odhalit chyby zavčas a sníží tím počet problémů, které by se dali vytvořit při další práci s chybnou částí kódu. Nejčastější nalezené problémy se týkají například nestandardního chování prvků, špatně nazvaných prvků, nebo špatně dohledatelné informace. [39]

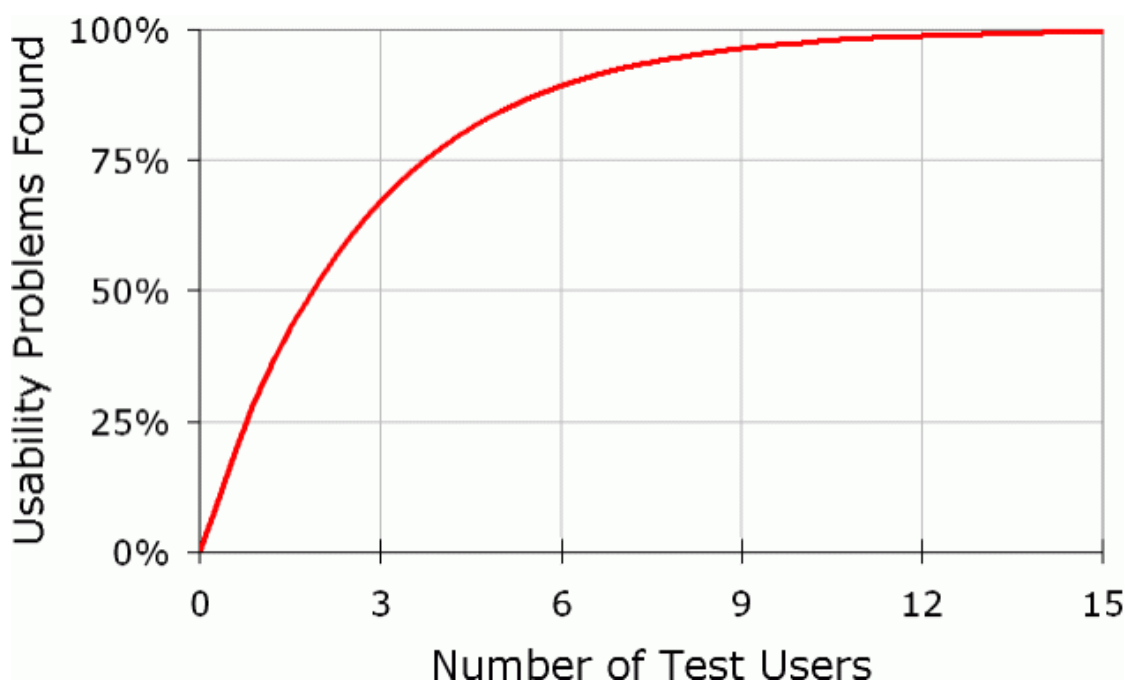
6.1.1 Respondenti

K uživatelskému testování nesmí chybět samotní respondenti – potenciální uživatelé naší aplikace. Vybraná skupina uživatelů by měla být rozmanitá – různý věk, pohlaví, technická úroveň atd. V případě této práce se jedná o testování všech funkcí nového systému pro správu revizí. Revize spravují zaměstnanci firmy a se zápisem vypomáhají praktikanti. Uživatelé tedy znají starý systém, ale například nepoužívali některé funkce, jako je správa kontaktů na mistry nebo firmy. Z řad praktikantů jsem navíc vybral 2, kteří starý systém dlouhou dobu nepoužívali.

6.1.2 Počet účastníků

Dle Jakoba Nielsena [40] odhalíme 60-80% chyb při testování s 3-5 uživateli. Mohlo by se zdát, že čím více testů provedeme, tím více problémů odhalíme, ve skutečnosti se nalezené problémy budou duplikovat. Některé problémy zjištěné při prvním testování se budou překrývat s těmi, které nalezne uživatel v druhém testování atd. Graf vztahu množství nalezených chyb a počtu respondentů je na obrázku 6.1. Pro uživatelské testování nové aplikace pro správu revizí jsem si zvolil skupinu o 8 respondentech, kteří podle zmíněného grafu odhalí více než 95% problémů.

Mezi testy jsem průběžně opravoval chyby, na které se postupně v testech přišlo. Dělal jsem i úpravy podle nápadů uživatelů, které mě samotného při vývoji nenapadli nebo jsem si jich nevšiml. Většina úprav přispěla ke zjednodušení nebo zvýšení přehlednosti aplikace. Tímto jsem odstranil to, že bych z testování dostával velké množství duplicit. Nápady, které mi nepřišli nutné jsem nechal na konec, kdy jsem provedl vyhodnocení. Pokud mělo stejný návrh více uživatelů tak jsem změny provedl.



■ **Obrázek 6.1** Procento nalezených chyb v závislosti na počtu respondentů. [40]

6.1.3 Scénáře

Před začátkem testování je potřeba připravit si scénář testování, který obsahuje popis nějaké realistické situace, s cílem a úkoly pro respondenty, které vedou k naplnění cíle a ověření použitelnosti testované části aplikace daným scénářem. Vytváření úkolů by se mělo držet několika pravidel.

- Úkoly nesmí být rozděleny na jednotlivé části (například úkol: nakupte černé triko nerozdělovat na: vyberte produkt, vložte do košíku, otevřete košík, atd.)
- Úkoly nesmí navádět jak samotný úkol vyřešit. [41]

V aplikaci pro správu revizí nejsou žádné dlouhé scénáře, které by zahrnovali většinu funkcí aplikace. Připravil jsem tedy osm kratších scénářů, které se jednotlivě zaměřují na funkce systému. Scénáře se zkráceně týkají těchto témat:

1. Kontrola platnosti revizí na zadaném stroji.
2. Změna informací o mistrech na dané středisko.
3. Zavedení nalezeného nedostatku při provádění revize do systému.
4. Přišla informace o odstranění daného nedostatku.
5. Zavedení nového provozního deníku k tlakovým lahvím.
6. Zápis nově provedených revizí v dané kategorii.
7. Výtisk seznamu nadcházejících revizí pro danou firmu provádějící revize na tlakových lahvích.
8. Provedení fiktivní objednávky v dané kategorii.

Kategorie ve scénářích jsem měnil podle toho, za kterou je respondent v systému zodpovědný. U praktikantů jsem objednávání rovnou přeskočil, nebo stačilo, když zkopírovali tabulku se seznamem. Navíc jsem na konec přidal „Extra“, v kterém odkazuji uživatele na části systému, které v rámci testů nepoužili. Jedná se například o opravované protokoly u tlakových lahví a způsob zobrazení obsahu provozních deníku u tlakových lahví. Log (historii změn), do kterého se mohou podívat pro kontrolu provedených změn. Stránku kontaktů firem a kategorie, které uživatel sám nespravuje.

6.1.4 Dotazník

Po testování je dobré mít připravené otázky pro další zhodnocení aplikace. V případě, že se testující zeptá pouze otázkou: „Máte nějaké připomínky, nápady, nebo něco dalšího co byste chtěl dodat?“ Respondent bude těžce vzpomínat nebo bude dlouho přemýšlet o tom, co říct. Pokud ale dostane už připravené otázky, které se budou týkat konkrétních prvků, nebo částí aplikace, bude zpětná vazba získaná od respondenta mít vyšší hodnotu. Otázky mohou být například koncipovány na různé stránky aplikace nebo na nové funkce. Dobré je, aby si při odpovídání na otázky respondent i stránku otevřel, aby si lépe vzpomněl, nebo si vyzkoušel části, které nevyužil při plnění scénářů v průběhu testu.

Do dotazníku po testování jsem připravil celkem dvanáct otázek. Zahrnul jsem jak obecné tak i specifické otázky. Uvádím zkrácenou formulaci otázek, nejedná se přesně o otázky uvedené v dotazníku.

1. Jak hodnotíte možnost zobrazení všech zařízení pro jednotlivé kategorie.
2. Jak se vám líbí způsob vyhledávání podle zařízení a stroje na kartě zařízení?
3. Jak hodnotíte stránku se všemi informacemi ke stroji (karta stroje)?
4. Jak hodnotíte způsob filtrování na stránce tlakových lahví?
5. Jak hodnotíte způsob editace a vytváření pomocí formulářů?
6. Jak hodnotíte možnost editace přímo ze seznamu zařízení, která ve starém systému nebyla.
7. Chybí vám něco v novém systému?
8. Pracovalo se vám v novém systému lépe?
9. Přijde vám v systému něco navíc?
10. Co se vám na novém systému líbí?
11. Máte nějaké další připomínky nebo návrhy?
12. Potřebujete nějaké roční exporty nebo jiné informace?

6.2 Průběh

Před začátkem samotného testování je dobré uživatele seznámit s obecným průběhem testování. Ujistit je, že pokud se zaseknou u některého úkolu, není to jejich chyba, ale jedná se o chybu v návrhu uživatelského rozhraní. Domluvit se na způsobu záznamu z testu (video nebo jen audio), zda souhlasí s případným zveřejněním záznamu (například do diplomové práce). Dále je dobré požádat uživatele, aby „přemýšleli nahlas“ – vyjadřovali své myšlenky při řešení úkolů nahlas a v neposlední řadě, že by se neměl s testerem radit v průběhu testu. [42]

Uživatelské testování probíhalo od 21.11.2023 do 27.11.2023. Jako testová data jsem použil měsíc starou zálohu ze starého systému. Testování na datech, která se v systému opravdu nacházejí má výhody v tom, že uživatel data zná a nebude muset přemýšlet nad tím co například generovaná data mají znamenat. Jediné úpravy v datech jsem udělal tam, kde byli jména nebo kontaktní informace, z důvodu zachování ochrany osobních údajů zaměstnanců a externích firem.

Forma testu byla podle potřeby respondentů, s jedním uživatelem proběhlo online formou, kde jsme pro komunikaci použili aplikaci Microsoft Teams [43]. S ostatními respondenty už testování proběhlo osobně, přímo na jejich počítači, kde byli ve své „komfortní“ zóně. Pro záznamy jsem využil program Screenpresso [44], který je na pracovních počítačích předinstalovaný. Před začátkem jsem uživatele seznámil s průběhem testování a poslal jim scénáře. Při plnění úkolů jsem získával zpětnou vazbu z jejich „hlasitého myšlení“ a dělal jsem si poznámky o tom jak scénářem prochází. Po dokončení scénářů jsem s nimi prošel dotazník po testování, kdy jsem jim i připomněl, že si aplikaci mohou otevřít a znovu se na ni podívat.

Záznamy z testování s respondenty naleznete na přiloženém médiu. Dva z respondentů nechtěli zveřejnit hlasový záznam, v těchto dvou případech se tedy jedná pouze o záznam obrazovky. Z jednotlivých testů v následujících podkapitolách popisují nejdůležitější poznatky z testování. Celkový zápis nalezených problémů a případných návrhů na zlepšení jsem shrnul do seznamu, který je na konci kapitoly testování 6.3. Celý dokument s úvodním dotazníkem a scénáři jsem přidal do příloh C.1 a C.2. V příloze C jsou navíc i jednotlivé vyplněné dotazníky respondentů spolu s poznámkami, které jsem si v průběhu testu zapsal.

6.2.1 Respondent 1

Muž 42 let, zaměstnanec s vysokou znalostí domény revizí, administrátor aplikace, programátor

Při kontrole platnosti revizí na stroji si všiml, že jedna z revizí je propadlá, ale musel si všechny datumy přečíst a informaci neviděl na první pohled. Navrhl zvýraznění polí podbarvením, zvolil jsem zelené a červené podbarvení, které je na první pohled jasné. Tuto úpravu jsem zavedl hned po testování. Při změně mistrů navrhl více zvýraznit řádek v seznamu, na kterém se nachází myš.

Při třetím scénáři na vytvoření nedostatku se mu podařilo najít chybu v ukládání dat do databáze, kterou jsem špatně otestoval a aplikace spadla. Nejednalo se o nic vážného a chybu jsem ihned opravil. Potvrzení o odstranění nedostatku už bylo v pořádku.

Při vytváření provozních deníků řekl, že by se jejich označení mělo generovat automaticky. Věděl, že pro přidání zařízení do deníku musí jít do formuláře zařízení a následně deník přiřadit. Nevšiml si ale, že jsou změny zamčené, jelikož se dalo otevřít rozbalovací pole pro výběr deníku (samotný výběr ale nešel). Po testování jsem toto opravil.

Zbytek scénářů už proběhl jen s menšími poznámkami, ale samotné splnění scénářů zvládl bez problému. Chyběla mu možnost vložení zkopírovaného seznamu do formuláře a po vložení zkopírované tabulky do excelu by rád rovnou měl lépe naformátovanou velikost sloupců. Při procházení aplikace mimo testování si všiml, že se často objevuje stroj bez jakékoliv revize. To bylo způsobeno tím, že velké množství strojů má pouze elektro revizi, které jsou ale mimo rozsah této práce. Rozhodl jsem se tedy pouze přidat informace o platnosti elektro revizí na stránku stroje.

Celkově aplikaci hodnotil kladně, pouze si musí trochu zvyknout. Nové funkce se mu líbily a dobře se mu v aplikaci pracovalo. Nic co by mu chybělo nenašel, ale říká, že jestli něco chybí, tak se to ukáže až po delší době užívání aplikace. Na aplikaci se mu nejvíce líbí to, že se jedná o webovou aplikaci a změny prováděné tak budou probíhat jednoduše oproti starému způsobu. Do připomínek zmínil podbarvení platnosti revizí, export tabulky přímo do excelu s formátovanými sloupci a výraznější indikaci aktuálního řádku v seznamech.

6.2.2 Respondent 2

Muž 22 let, praktikant s velmi nízkou znalostí domény revizí, aplikaci nepoužívá, pokročilý v práci s počítačem

Přidávání kontaktů na nové středisko hledal na konci seznamu, ale když ho nenašel tak rovnou hledal v hlavičce seznamu, kde se tlačítko nachází. Z neznalosti systému přesně nevěděl, kde najít nedostatky. Hledal to přímo u kategorie. Následně už nedostatek vytvořil bez problému a uvítal, že se mu nepodařilo vytvořit nedostatek bez zadaného stroje.

Nenašel stránku s provozními deníky. Po nalezení stránky mu vytvoření nedělalo žádný problém, ale přidání zařízení do deníku nenašel ihned. Při děláni změn ve formuláři nevěděl, proč nemůže změny provádět, ale po ujasnění funkce odemykacího tlačítka se mu tato funkčnost líbila. Zbytek testu proběhl bez problému.

Celkově aplikaci hodnotil pozitivně. Jelikož systém nepoužívá, tak ho žádné další nápady nebo chybějící funkce nenapadli. V aplikaci se mu v rámci testování pracovalo dobře. Nejkladněji hodnotil potřebu odemčení formuláře pro provádění změn a nutnost uložení. Zamezí to počtu chyb, které se ve starém systému stávali (vyhledávání a formuláře v MS Access používá v jiné aplikaci).

6.2.3 Respondent 3

Muž 39 let, zaměstnanec se střední znalostí domény revizí, správce 5 kategorií, středně pokročilý v práci s počítačem

Neměl znalost o existenci stránky stroje, se samotnou stránkou problémy neměl, ale neviděl na první pohled odkazy na stažení protokolů, které jsou zvýrazněny modrou barvou textu. Při druhém scénáři opět nevěděl ihned kde stránku se středisky a mistry hledat, to je způsobeno tím, že ani ve staré aplikaci o této funkci nevěděl. Při vytváření provozního deníku měl stejný problém jako respondent 2 s uzamčenými poli ve formuláři. Navíc se snažil vracet pomocí šipky zpět v prohlížeči a očekával, že mu zůstane vyhledané zařízení. Zbytek scénářů prošel bez problémů.

Novou aplikaci hodnotí jako přehlednější než minulé řešení. Dokáže si představit využití všech nových funkcí a rychlejší zápisy do aplikace. V seznamu revizí mu chyběla možnost vyhledávání podle čísla vybavení, ale po zmínění funkce prohlížeče CTRL+F vyhodnotil, že by samotný seznam neměnil. Mezi připomínky a návrhy zmínil potřebu seřazení seznamu nadcházejících revizí podle data revize pro kategorie které spravuje (řazení bylo nastavené jako u tlakových lahví podle firmy). Navíc se mu líbila informace o tom, jestli je revize objednaná, která se nachází pouze u plynových zařízení.

6.2.4 Respondent 4

Muž 55 let, zaměstnanec se střední znalostí domény revizí, správce TNS a SHZ, základní schopnost práce s počítačem

I přes úvodní informace o tom, jak má testování probíhat, a i po upozornění na to, že by testem měl projít sám, se často doptával na věci mimo scénáře, které se týkají kategorií, které spravuje. Nebyl to tedy úplně test použitelnosti, ale více test funkčnosti aplikace a kontrola, jestli obsahuje všechny funkce, které má mít pro dané kategorie. Na začátku testu si myslel, že

se testování týká pouze jeho dvou kategorií, musel jsem znovu upřesnit, že se jedná o test celé aplikace.

Po vytvoření provozního deníku se vrátil na formulář pomocí šipky zpět a vadilo mu, že mu nezůstalo vyhledané zařízení. Nevšiml si odemykacího tlačítka ve formuláři. V průběhu testu říkal jak by chtěl přejmenovat některá pole ve formuláři. Pro objednávání by ocenil export seznamu přímo do excelu. Ostatní scénáře v rámci možností zvládl projít, ale často se ptal, jestli to dělá správně.

Celkově aplikaci hodnotil kladně, odpovědi v dotazníku byli velmi stručné. Při dávání zpětné vazby byl velmi zaměřený na své kategorie, pro které chtěl hned několik změn, ale často se jednalo o maličkosti, jako změna názvu sloupce, nebo pole ve formuláři. Nejvíce se mu líbila karta stroje a možnost editace zařízení rovnou ze seznamu. Chyběla mu možnost zobrazit si přímo vybraný měsíc pro tlakové lahve, tak jak tomu bylo ve starém systému.

6.2.5 Respondent 5

Muž 20 let, praktikant s velmi nízkou znalostí domény revizí, občasná kontrola a zadávání protokolů, pokročilý v práci s počítačem

Kvůli neznalosti systému nevěděl, kde přesně hledat, ale pro hledání použil CTRL+F a nakonec i stránku stroje našel. Zámku polí ve formuláři si hned nevšiml. Obecně všemi scénáři prošel rychle a nikde se dlouho nezasekl, navíc používal CTRL+F pro hledání v seznamech a v posledním scénáři při fiktivní objednávce i deaktivoval zařízení, které bylo na odstaveném stroji.

Na dotazník po testování odpovídal velmi stručně v pozitivním směru, říká, že vyhledávání je v aplikaci přehlednější a editace přímo ze seznamu zrychlí práci. Celkově se mu líbí zpracování, přehlednost a jednoduchost aplikace. Přípominky k testování neměl.

6.2.6 Respondent 6

Muž 21 let, praktikant s velmi nízkou znalostí domény revizí, občasná kontrola protokolů, pokročilý v práci s počítačem

Scénáři prošel velmi rychle a bez delšího zdržení. Jediné delší hledání bylo u provozních deníku tlakových lahví. Při práci nedostatků navrhoval možnost zaškrtnutí všech vyřešených nedostatků naráz, aby se nemuseli postupně otevírat jednotlivé formuláře. Jako jediný z respondentů zadával nové data provedených revizí přes formuláře ze seznamu, všichni ostatní prováděli zadávání z karty zařízení.

Na dotazník po testování odpovídal velmi stručně v pozitivním směru, líbí se mu přehlednost stránky stroje a editaci zařízení hodnotí jako rychlejší a jednodušší oproti staré aplikaci.

6.2.7 Respondent 7

Muž 46 let, zaměstnanec s vysokou znalostí domény revizí, dříve správce několika kategorií, aktuálně jedné, pokročilý v práci s počítačem

Sedmý respondent si nejprve celý systém důkladně prohlédl a vyzkoušel, i přes to, že jsem ho žádal aby prvně prošel scénáři, a po otestování použitelnosti jsme aplikaci mohli projít dopodrobna. Když se nakonec dostal ke scénářům, tak je bez problémů a velmi rychle prošel.

Při procházení aplikace měl pár připomínek a funkcí co by chtěl přidat. U freonů chce vidět datum aktivace a deaktivace. Navrhoval možnost vytvoření nedostatku rovnou z formuláře zařízení a možnost uložení protokolu ve formuláři. Rád by také měl možnost rovnou vytvoření emailu pro objednávku revizí na freony. V aplikaci objevil malou chybu, kdy se špatně načítal seznam všech nedostatků, místo jen těch otevřených při prvním spuštění stránky nedostatků. V seznamech si všiml, že se zvýrazňují data i u deaktivovaných zařízení.

Aplikaci hodnotil kladně. Líbí se mu zvýšená přehlednost a intuitivnost nové aplikace. Oproti

starému řešení vidí velký pokrok v odezvě systému, kde dříve musel čekat několik desítek vteřin na první načtení staré aplikace. Velmi ocenil možnost editace zařízení ze seznamu nadcházejících revizí.

6.2.8 Respondent 8

Muž 44 let, zaměstnanec se střední znalostí domény revizí, správce 1 kategorie, základní schopnost práce s počítačem

Scénáře prošel rychle. Funkce které normálně nepoužívá dlouho nehledal, pouze provozní deníky tlakových lahví ho trochu zdržely. Byl druhým uživatelem, který při změně data poslední revize využil možnost editace zařízení ze seznamu nadcházejících revizí. Přípomínky téměř neměl, je to spíše typ uživatele, který pracuje s tím co má a nad tím co by mohl přidat moc nepřemýšlí. Po testování jsem se zmínil o informaci o objednání revize, která aktuálně byla jen u plynových zařízení a po vysvětlení funkce se přiznal, že i u své kategorie by tuto informaci ocenil.

Na dotazník po testování odpovídal velmi stručně v pozitivním směru. Píše, že je aplikace kompaktní a dá se v ní lépe vyhledávat.

6.3 Shrnutí

Celkově se otestovala funkčnost a použitelnost aplikace. Nenašli se žádné závažné problémy a aplikace fungovala bez problémů, až na první test, při kterém u vytváření nedostatků spadla. Půžitelnost aplikace je také ověřená. Uživatelé byli schopní najít to co potřebovali bez větší znalosti aplikace, jediné krkolomné místo jsou provozní deníky u tlakových lahví, ale to mám v plánu do budoucna změnit. Navíc noví uživatelé budou aplikaci před prvním použitím provedeni.

V této kapitole vypisují seznam většiny provedených a plánovaných úprav, které se objevili testováním. Seznam je dlouhý, ale převážná většina jsou malé změny věcí, kterých jsem si sám při implementaci a testování nevšiml nebo chytré nápady, které mě nenapadli. Ani jedna z chyb nebyla závažná, jednalo se především o věci, které uživatelé chtěli změnit nebo přidat, ale při sběru požadavků v analýze se o tom nezmínili. Seznam jsem rozdělil na tři části. Změny, které jsem zavedl ihned po provedení testu s respondentem (další testování už probíhalo s úpravami). Změny, které jsem provedl po testování se všemi uživateli a plánované změny, kterými se budu zabývat postupně při psaní nebo po dokončení textu k diplomové práci.

Změny prováděné ihned po testu.

1. Podbarvení platnosti revizí na stránce stroje.
2. Automatické generování označení provozních deníků.
3. Přidání platnosti elektro revizí na kartu stroje.
4. Lepší zvýraznění aktuálního řádku v seznamu
5. Možnost filtrace podle kategorie v nedostacích.
6. Možnost vytvoření nedostatku jen na aktivní stroj.
7. Přidání odkazu na filtrované nedostatky do menu k jednotlivým kategoriím.
8. Přidání odkazu na vytvoření provozního deníku do formuláře.
9. Změna řazení seznamů nadcházejících revizí podle potřeby jednotlivých kategorií.
10. Změny názvu sloupců a polí ve formulářích.

11. Změna pořadí sloupců v seznamech.

Změny provedené po dokončení testů se všemi respondenty.

1. Export seznamu rovnou do naformátované excel tabulky.
2. Uložení posledního hledaného zařízení ve formuláři.
3. Informace o tom, jestli byla revize objednaná ve většině kategorií.
4. Možnost zobrazit si vybraný měsíc u TNS.
5. Přidání informace o tom, kdy se zařízení aktivovalo a deaktivovalo u freonů.
6. V seznamech zvýraznit vybavení se zrušeným statusem.
7. V seznamech nezvýrazňovat datum u neaktivních zařízení.
8. Nedostatky zrušení podbarvení u ukončených a stopnutých.
9. Při vytváření nedostatku nastavit datum předání automaticky na dnešní.
10. V seznamech z čísla stroje udělat odkazy na kartu stroje.
11. Přidání data odstranění u nedostatku automaticky po potvrzení.

Navržené změny, které se budou implementovat v blízké budoucnosti.

1. Z formuláře možnost vytvořit nedostatek.
2. Z karty stroje umožnit otevřít formulář daného zařízení.
3. Možnost uložení PDF na správné místo, pod správným jménem z formuláře.
4. Možnost exportovat tabulku rovnou do emailu pro objednávání.
5. Rovnou přiřadit vytvořený šanon k zařízení, kde byl vytvořený.
6. Upravit způsob vytváření a přiřazení do provozního deníku.
7. Potvrzení odchodu bez uložení změny z formuláře.

Kapitola 7

Nasazení

Aplikace byla nasazena k používání od 4.12.2023. O předchozím víkendu jsem provedl migraci dat. Všichni uživatelé, kteří systém používají o vývoji nové aplikace i o době testování věděli. Po nasazení jsem jim dal vědět emailem, ve kterém jsem znova poslal seznam navržených změn a upřesnil jsem i hlavní změny, které jsem od testování zavedl. V emailu jsem se dále ptal, jestli nemají další návrhy nebo nápady co přidat a upravit.

Nasazením aplikace cyklus vývoje nekončí, pouze začíná fáze údržby. Aplikaci je postupem času potřeba měnit, přibývají funkce, je potřeba aplikaci aktualizovat atd. V případě aplikace pro správu revizí se bude jednat především o přidávání dalších kategorií. Počítal jsem, že se po vytvoření aplikace bude následně postupně předělávat kategorie „elektro“ revizí, která je velmi rozsáhlá, ale jak už to chodí, vedoucí se rozhodl přidat kategorii jinou.

7.1 Migrace dat

Celkové nasazení aplikace nebylo náročné. S migrací dat se počítalo už od analýzy, takže bylo vše připraveno jak bylo potřeba. Migrace dat spočívala v přesunu dat z databáze v MS Access na databázový server do databáze Microsoft SQL Server 2019. Jednalo se tedy o přesun mezi dvěma aplikacemi od Microsoftu.

Člověk by očekával, že takovýto přesun bude bez jakéhokoliv zásahu uživatele. Pouze se zvolí jaké tabulky je potřeba exportovat z MS Access na server. Ve skutečnosti tomu tak není. Tabulky se musí exportovat po jedné a klíče se po přesunu nezachovávají. Navíc v některých případech ani neodpovídají správně typy sloupců. Po exportu tabulek jsem musel opět nastavit klíče, u automaticky generovaných klíčů zapnout generování a ve všech tabulkách zkontrolovat typy. Nejhorší chyba byla v případě, kdy se sloupec typu string s dlouhým číslem zakázky převedl na float. Zmizelo tak přesné označení zakázky, místo kterého byl zápis ve tvaru „7.23E+10“.

7.2 Docker

Po nasazení aplikace jsem potřeboval spustitelnou verzi aplikace mimo firmu pro účely diplomové práce. Aplikace obsahuje jednu knihovnu přímo od firmy pro získávání informací o uživateli z active directory, tu jsem musel odebrat. Navíc bylo potřeba odstranit veškeré věci co s firmou souvisí, včetně designu, loga a dat.

Pro vygenerování dat jsem použil stránku mockaroo. Jelikož se jedná o náhodně generovaná data, tak nedávají velký smysl. V seznamech jsou často revize propadlé, nebo jsou aktivní zařízení na „odstavených“ strojích. V případě použití jinde, bude potřeba přizpůsobit si databázi, názvy sloupců a polí v aplikaci nebo přizpůsobit své procesy aplikaci. Aplikace přiložená k diplomové

práci slouží spíše pro ukázkou toho jak takový systém vytvářený na míru může vypadat.

Pro vytvoření této verze aplikace spustitelné kdekoliv, jsem se rozhodl použít docker. Vývoj celé aplikace probíhal na pracovním zařízení na operačním systému Windows spolu s Apache serverem pro lokální testování. Docker image pro apache server ale běží na linuxu a jsou potřeba nainstalovat i další podpůrné knihovny pro php, jako například rozšíření pro MS SQL server. Vytvoření obrazu tedy nebylo tak jednoduché, jak jsem očekával.

S dockerem jsem dříve nepracoval, tak mi vytvoření spustitelné aplikace trvalo déle než známému uživateli. Pomocí dokumentace [45] a docker hubu [46] jsem našel správný image (php:8.1.13-apache), se kterým jsem začal. Postupoval jsem podle návodu na konfiguraci apache serveru a snažil se nainstalovat veškeré potřebné knihovny. Zasekl jsem se u knihovny pro připojení do MS SQL Serveru, kterou bylo potřeba stáhnout přímo z microsoft stránek. Přidávám ukázkou kódu 7.1.

■ **Výpis kódu 7.1** Ukázkou docker kódu pro instalaci ODBC a SQL Server ovladačů

```
RUN curl https://packages.microsoft.com/keys/microsoft.asc |apt-key add -
RUN curl https://packages.microsoft.com/config/ubuntu/20.04/prod.list
  > /etc/apt/sources.list.d/mssql-release.list
RUN apt-get update
# Install msodbcsql17
RUN ACCEPT_EULA=Y apt-get -y --no-install-recommends install msodbcsql17
# Install sqlsrv and pdo_sqlsrc
RUN pecl install sqlsrv
RUN pecl install pdo_sqlsrv
```

Následně jsem potřeboval vlastní konfiguraci apache serveru, která je potřeba do dockeru nakopírovat. Dalším krokem bylo připravení obrazu s databází a nakonec pomocí docker-compose propojení obou obrazů. V docker-compose jsem připojil složku se zdrojovým kódem aplikace a mělo být hotovo. Narazil jsem ale na další problémy spojené s rozličnostmi Windows a Linux.

Byli potřeba změnit lomítka v cestách souborů z Windows značení (\) do značení Linuxu (/). Zde jsem se také zasekl na nějaký čas, jelikož jsem neúmyslně přepsal i lomítka v názvech některých tříd. Po vyřešení tohoto problému následoval další. Pro stránky kategorií, které jsou podobné používám jednotnou latte šablonu, na kterou se v presenter třídách odkazují přes dynamickou cestu („../sablonu.latte“), ale pro samotné kategorie složka neexistuje (složky UPS, Plyn atd.) Windows toto zvládne, ale Linux potřebuje mít prázdné složky vytvořené.

Po zprovoznění aplikace po několika odpolednech, jsem zjistil, že aplikace má strašně dlouhou odezvu. Na načtení každé stránky jsem musel čekat několik vteřin, někdy i více než 10. Zjistil jsem, že je to způsobeno nutností časté komunikace mezi Windows a Linuxem uvnitř dockeru, jelikož jsem zdrojové soubory pouze připojil jako složku. Upravil jsem tedy soubory s konfigurací dockeru a docker-compose aby se zdrojové soubory do dockeru zkopírovali, navíc je potřeba nastavit oprávnění zápisu do složek log a temp. Po provedení této úpravy je obraz funkční a aplikace funguje v rámci možností s nízkou odezvou.

Instrukce ke spuštění jsou napsané v „readme“ souboru v kořenovém adresáři příloženého média a v kořenovém adresáři zdrojových souborů aplikace.

7.3 Přidání nové kategorie

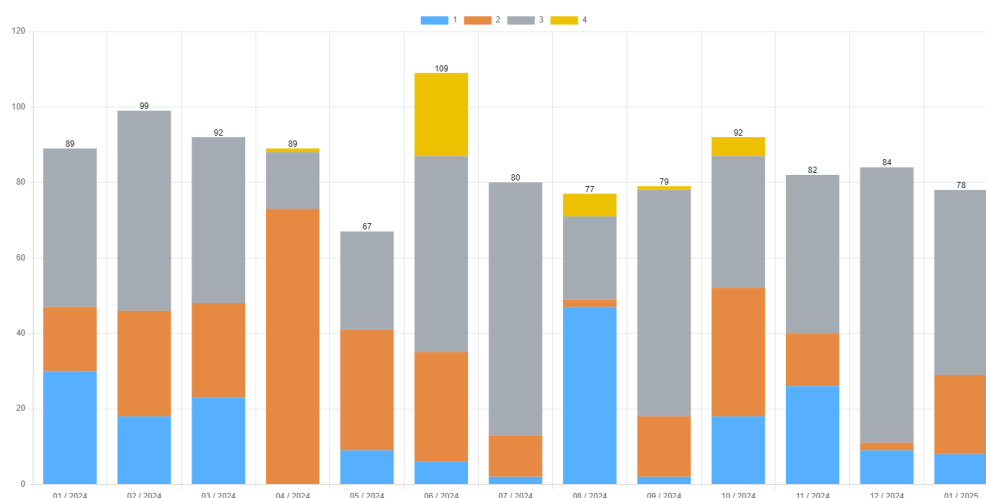
Po nasazení se ke mě ihned dostal požadavek na přidání nové kategorie takzvaných „termo“ revizí. O tomto tématu jsem nikdy při analýze neslyšel, ale aplikaci jsem připravoval tak, aby se další kategorie dali jednoduše přidat. Kategorie termo byla trochu odlišná od ostatních. Tento druh revizí provádí technici z oddělení, na kterém pracuji a potřebovali mít možnost vyvažovat počet revizí na každý měsíc.

Za dlouhou dobu provádění revizí za pomoci jiného systému měli počty na měsíc do jisté míry vyvážené, ale přesouvání nebylo otázkou jednoduchého přepsání data, nebo nastavení posunu.

Přesouvání revizí technikům zabíralo velké množství času (více než 3 minuty na jednu revizi, kde pro samotnou revizi včetně vytvoření protokolu mají nastavenou 1 hodinu času). Tento čas mohli trávit samotným prováděním revizí. V aplikaci jsem to vyřešil jednoduchým číselným vstupem, do kterého se zapisuje o kolik měsíců se má revize posunout (dozadu se revize nemůže posunovat, protože by propadla).

Do aplikace jsem tedy kategorii zavedl. Samotné zavedení trvalo zhruba 3 hodiny, Jelikož kategorie byla odlišná od ostatních. Termo revize jsou identifikovány umělým identifikátorem. Nemají žádné výrobní číslo jako ostatní kategorie, jsou pouze dělané přímo na daný stroj. Musel jsem proto na stránce odstranit možnost vyhledávání podle zařízení. Pro jednoduchá plánování a posouvání revizí jsem vytvořil novou stránku s grafem, který obsahuje počty revizí na jednotlivá místa. Pro představu je graf vidět na obrázku 7.1.

Termo: Počet inspekci podle měsíce



■ Obrázek 7.1 Graf plánovaných termo revizí na jednotlivá místa

7.4 Další vývoj

Po zavedení termo revizí jsem dostal požadavky na přidání dalších dvou kategorií. Jedna z nich má požadavky stejné jako termo revize, její přidání tak zabralo zhruba hodinu. Druhá kategorie se opět trochu liší, revize nejsou dělané přímo na měsíce, ale na půlrok. Každý půlrok je potřeba udělat revize všech zařízení v této kategorii.

Vedoucí chce postupně přidávat další funkce, o kterých žádná zmínka při analyzování nezazněla. Jednou z nich je například stránka „Opatření“ na které se má objevit ihned záznam o jakémkoliv „propadlé“ revizi (cokoliv kde v aplikaci zčervená). Čtenář si řekne, že to logicky smysl dává, ale červené podbarvení se v aplikaci zobrazí i v případech kdy se pouze čeká na protokol nebo se stroj přesouvá. Ve výsledku to bude pouze přidávat práci lidem, kteří revize spravují, a ty důležité informace o fakticky propadlých revizích se ztratí v dlouhém seznamu. Zadávatel (vedoucí) ale tuto funkci v systému chce, i po řádném vysvětlení, tak jsem ji do aplikace přidal.

Postupem času předpokládám požadavky na zavedení dalších kategorií a když bude dostatek času tak i předělání celých elektro revizí. To je vize do budoucna. Aplikace se proto bude dále rozvíjet a měnit. Aktuální verze aplikace je stabilní a její používání je otestované, je to tedy skvělý základ na který může navazovat další vývoj, ať už to budu já nebo někdo jiný kdo vývoj bude provádět, kód je psaný čitelně a v „readme“ souboru jsou sepsané návody na to kde co najít v případě potřeby. Navíc je možné si přečíst tuto práci pro další pohled.



Kapitola 8

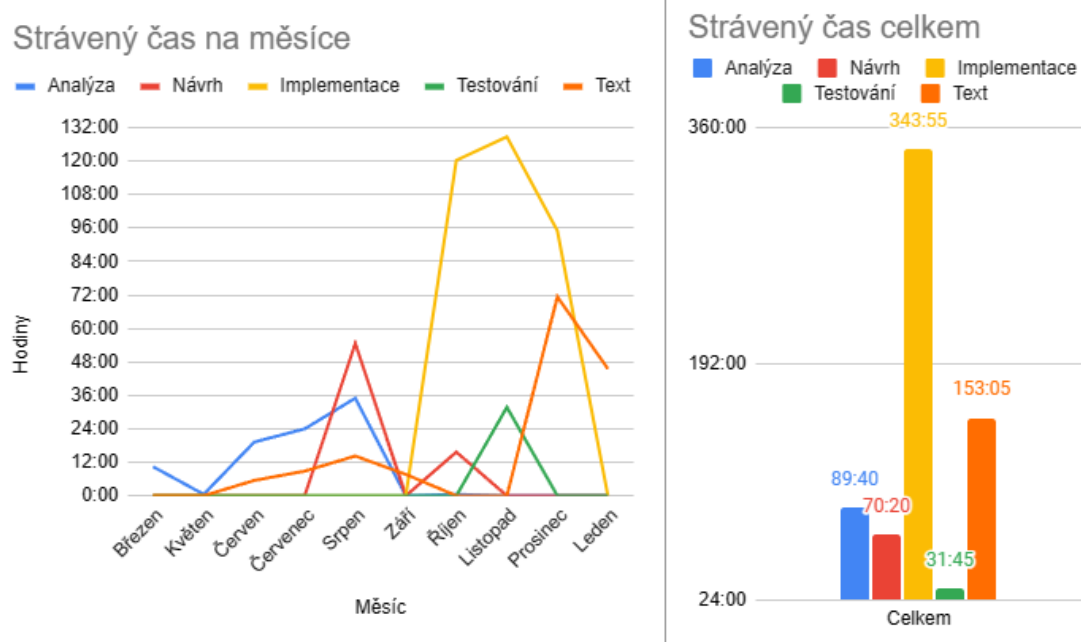
Závěr

Závěrem lze říci, že vytvořená aplikace je funkční, použitelná a už v době dokončování textu se ve firmě používá. Je to stabilní základ, na který půjde dále navázat dalším vývojem. Plánují se další kategorie, které by se do aplikace mohli zavést a používáním se mohou ukázat věci, které bude potřeba doladit.

Během realizace jsem se naučil do větší míry používat framework Nette, s kterým jsem dříve pracoval pouze okrajově a moc se mi s ním práce nelíbila. Po delší době strávené implementací ale mohu říci, že práce s Nette nebyla tak špatná, jak jsem si myslel. Nakonec se mi s ním i práce do určité míry zalíbila. Při vývoji jsem si také prohloubil své znalosti v oblasti stylování webových stránek pomocí CSS a Bootstrap frameworku.

Tento projekt byl první větší projekt, na kterém jsem se podílel po celou dobu sám. Pronikl jsem více do samotného vývoje webové aplikace a do celého procesu s ním spojeným. Vyzkoušel jsem si práci s Docker, který se v dnešní době často používá. Pronikl jsem i do procesů, které probíhají reálně ve firmě, která není přímo zaměřená na vývoj software, ale vytváří si řešení pro vlastní účely.

Výsledná aplikace, tak jak je přiložená k tomuto textu je použitelná, ale byla vytvářena na míru pro firmu kde pracuji. Pokud bude chtít někdo tuto aplikaci používat, bude muset udělat úpravy především ve struktuře databáze, nebo si rovnou vytvořit zpracování kategorie svým způsobem. Práce je dostupná na odkaze <https://gitlab.fit.cvut.cz/jordapav/ni-dip-revize>.



■ **Obrázek 8.1** Čas strávený prací na diplomové práci, rozdělený podle aktivity

V průběhu celé práce jsem si zaznamenával čas strávený prací na tomto projektu. Příkládám zde měsíční souhrny a celkový čas rozdělený podle typu aktivity v grafech 8.1. Při pohledu na graf je vidět velká převaha strávená právě implementací aplikace. To předpokládám jako očekávané, ale čas strávený samotným psaním textu by mohl být menší. Je to převážně z důvodu, že psaní textu mi jde velmi pomalu, dlouho přemýšlím nad tím co psát a když už mám něco napsané, tak dlouho měním formulace vět.

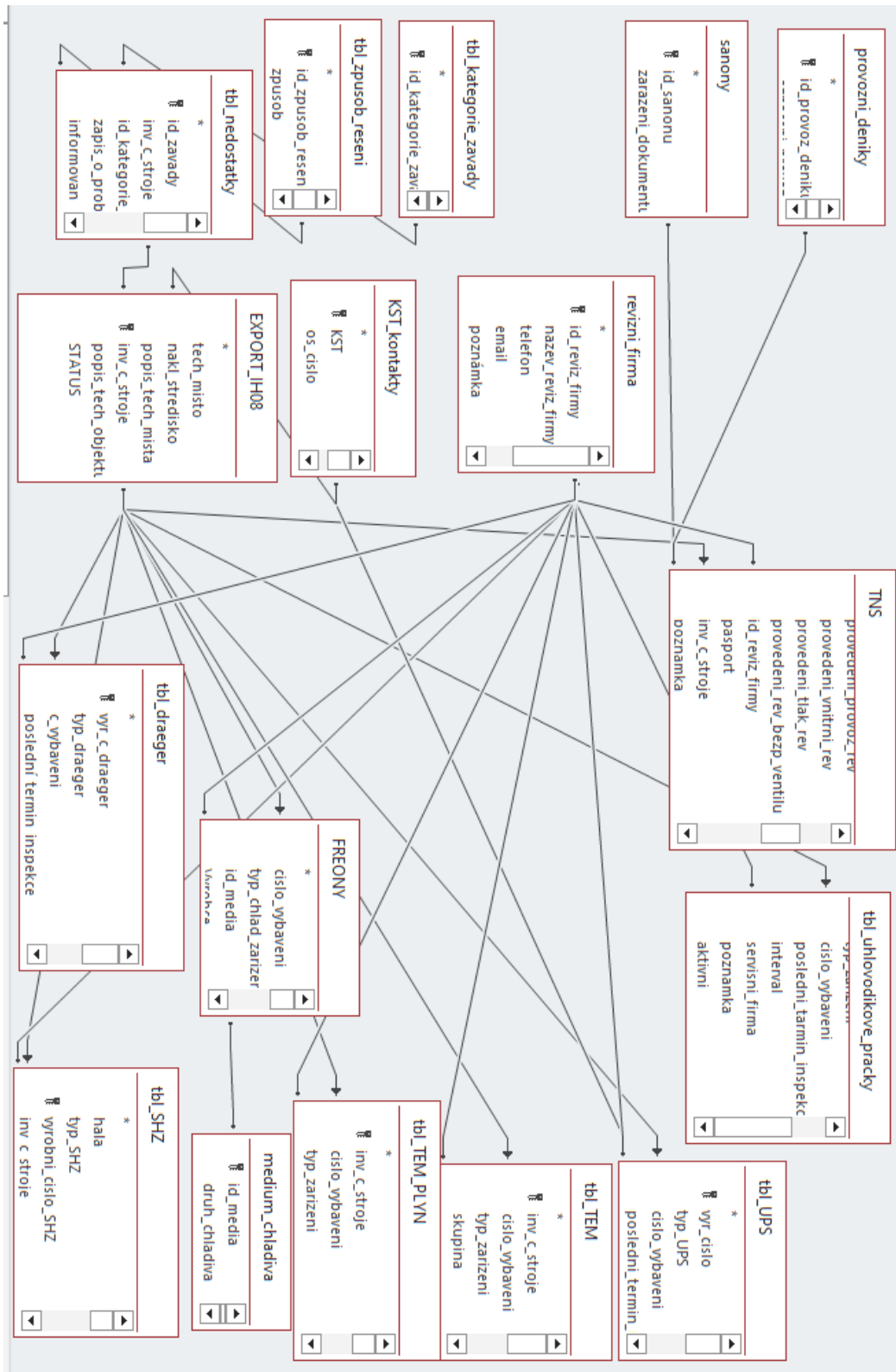
Původní řešení - Access

Kontrola dle vybavení

Kontrola přítomnosti revidovaných zařízení dle čísla vybavení stroje.

Číslo vybavení:	<input type="text" value="████████"/>	<input type="button" value="Zjistí stav"/>
TNS:	<input type="text" value="1"/>	Pozor, jsou přítomna vyhrazená zařízení
SHZ:	<input type="text" value="0"/>	
DRAEGER:	<input type="text" value="1"/>	Pozor, jsou přítomna vyhrazená zařízení
UHLOV._PRACKY:	<input type="text" value="0"/>	
TEM_PLYN:	<input type="text" value="0"/>	
UPS:	<input type="text" value="0"/>	
FREONY:	<input type="text" value="0"/>	

■ Obrázek A.1 Seznam prováděných revizí na vybavení



■ Obrázek A.2 Databázové tabulky staré aplikace

The screenshot shows a web application window titled 'frm_zalozeni_nedostatku'. The main heading is 'Založení nedostatku'. Below this, there are several input fields and buttons:

- *Stroj (číslo vybavení): [dropdown menu]
- *Informovaná osoba: [text input]
- *Kategorie závady: [dropdown menu]
- *Datum podání informace: [text input]
- *Způsob řešení: [dropdown menu]
- *Očekávané datum splnění: [text input]
- Číslo zakázky: [text input]
- *Zápis o problému: [large text area]
- Buttons: 'Uložit' and a button with a save icon.

Below the form is a table titled 'Nedostatky po revizích v řešení'. The table header includes columns: Č. vyb., Stroj, KST, Kategorie, Popis problému, Informován, Předání, Plán do, Zakázka SAP, OK, S, and poznamka. The table body is currently empty.

Obrázek A.3 Vytváření záznamu o nalezeném nedostatku

The screenshot shows a web application window titled 'Karty_SHZ' with the subtitle 'BOSCH Revision System - karta SHZ'. The form contains the following fields:

- V.č. SHZ: [text input]
- Typ SHZ: [text input]
- Datum poslední revize: [text input, value: 10. 9. 2019]
- TN1 - termín revize: [text input]
- TN1 - serial nr.: [text input]
- TN2 - termín revize: [text input]
- TN2 - serial nr.: [text input]
- perioda_prohlidky: [text input, value: 6]
- Umístěn na hale: [text input]
- Skupina: [text input]
- servis: [dropdown menu]
- Spouštěcí mechanismus: [text input]
- TN umístěna (stroj): [text input]
- Poznámka: [text area]
- Aktivní: [checkbox]

Below these fields is a section titled 'Umístění nádoby:' with an orange background, containing:

- Technické místo: [dropdown menu]
- Nákladové středisko: [text input]
- Popis tech. místa: [text input]
- popis tech. objektu: [text input]
- SAP status: [text input]
- Kontakt: [text input]
- HL.Mistr: [text input]
- Tel.: [text input]
- Hala: [text input]

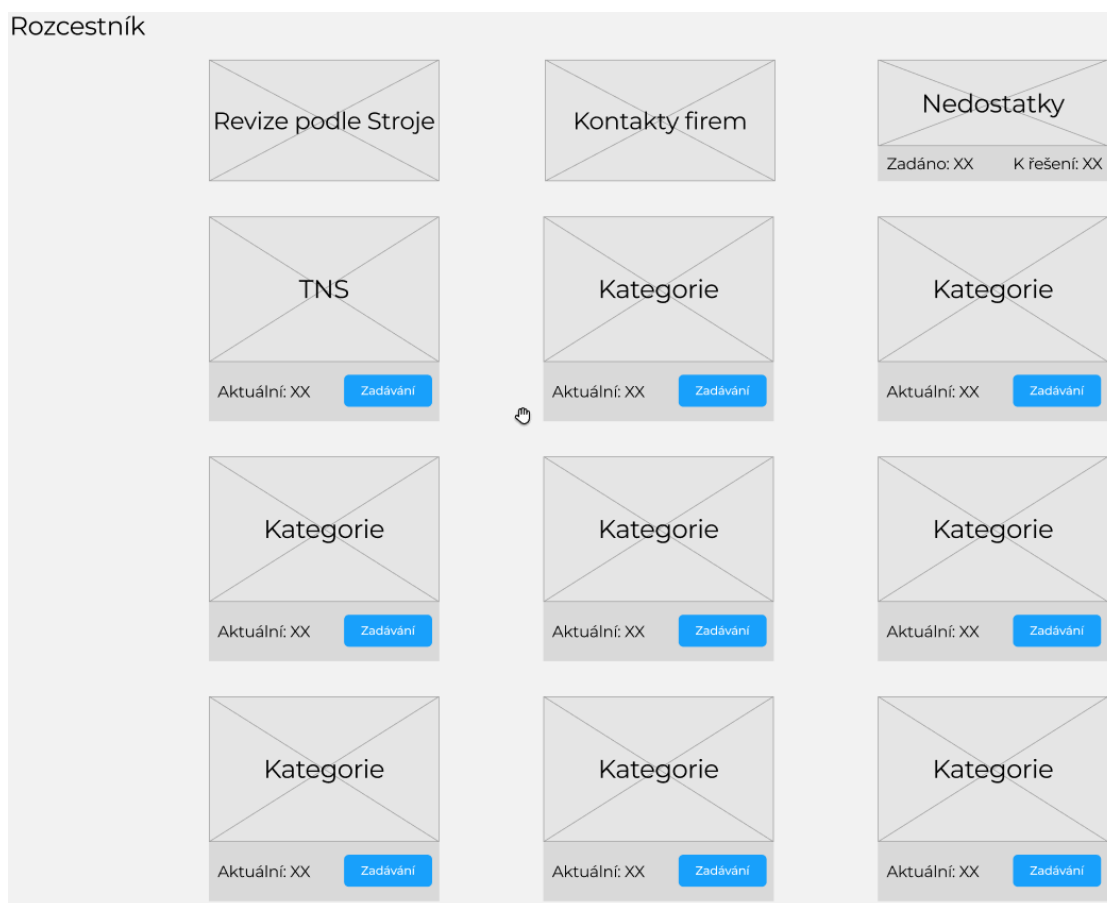
At the bottom, there is a status bar showing 'Record: 1 of 230', 'No Filter', and a 'Search' button.

Obrázek A.4 Formulář pro kategorii SHZ

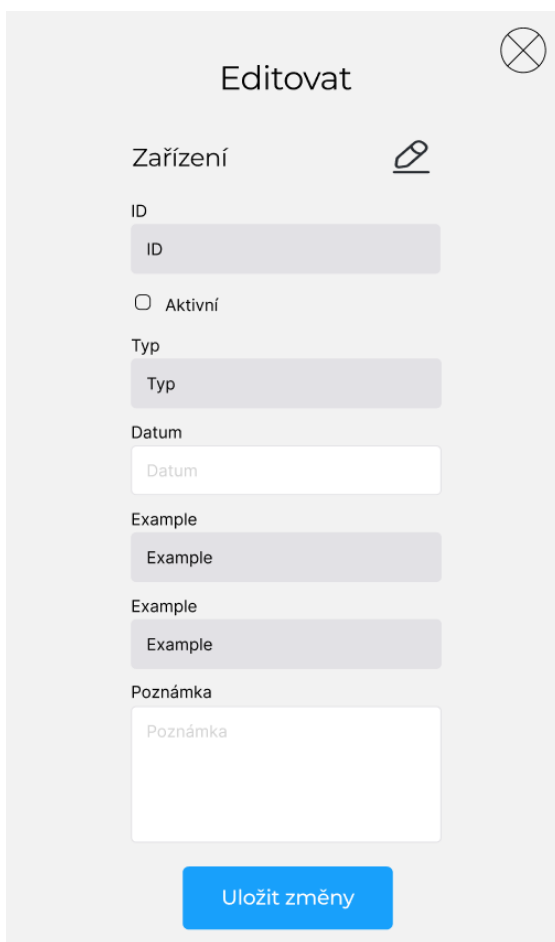
■ Obrázek A.5 Formulář pro kategorii freony

■ Obrázek A.6 Formulář pro kategorií DRAEGER

Návrh prototypu



■ Obrázek B.1 Návrh: Úvodní stránka



The image shows a mobile application screen for editing device information. The screen has a light gray background and a white title bar at the top with the text "Editovat" and a close button (a circle with an 'X'). Below the title bar, the word "Zařízení" (Device) is displayed next to a pencil icon. The form contains several input fields and a checkbox:

- ID:** A gray input field with the text "ID" inside.
- Aktivní:** A checkbox labeled "Aktivní".
- Typ:** A gray input field with the text "Typ" inside.
- Datum:** A white input field with the text "Datum" inside.
- Example:** A gray input field with the text "Example" inside.
- Example:** A gray input field with the text "Example" inside.
- Poznámka:** A large white text area with the text "Poznámka" inside.

At the bottom of the screen, there is a blue button with the text "Uložit změny" (Save changes).

■ **Obrázek B.2** Návrh: vyskakovací formulář pro zařízení

TNS


Plány revizí

Seznam opravovaných protokolů

Provozní deníky

Vyhledávání podle ID

Vyhledávání podle ID

Zařízení 

ID

ID

Aktivní

Typ

Typ

Datum

Datum

Example

Example

Example

Example

Poznámka

Poznámka

Umístění

Technické místo

Technické místo

Nákladové středisko

Nákladové středisko

Popis tech. místa

Popis tech. místa

Popis tech. objektu

Popis tech. objektu

SAP status

SAP status

Kontakt

Hlavní mistr

Hlavní mistr

Telefon

Telefon

Hala

Hala

■ Obrázek B.3 Návrh: stránka formuláře pro tlakové nádoby

Testování - scáněře, dotazníky

Dotazník před testováním

Pohlaví:

Věk:

Znalost domény revizí: nízká / střední / vysoká

Role v systému: zadávání revizí / objednávání revizí / kontrola / ...

Úroveň práce s počítačem: základní práce / středně znalý / pokročilý / expert

Čím trávíte v systému nejvíce času?

Instrukce

V průběhu testování prosím „myslete nahlas.“ Co se chystáte udělat, co hledáte, proč to hledáte tam, kde to hledáte, proč jste na první pohled neviděli tlačítko atd.

Zpětnou vazbu si prosím nechte na konec, po průchodu testovacích scénářů a vyplnění dotazníku.

■ **Obrázek C.1** Testování: Dotazník a instrukce

Scénář 1

Přijela kontrola a namátkově vybrala stroj pro kontrolu: J211793 a J220290. Ujistěte se, že jsou všechny revize platné. Kontrola také potřebuje vidět protokol alespoň jedné tlakové nádoby a jednoho plynového zařízení.

Scénář 2

Jste zodpovědná osoba za udržování platných kontaktů na mistry pro jednotlivá střediska (ve starém systému vedeno jako tabulka KST_kontakty). Přišli vám 2 emaily. Jeden o přidání nového střediska s označením 515999, a mistrem Alexem Skočdopole, hala 12 a telefon 666 456 654. Druhý email o změně mistra na středisku 515315, kde je nový mistr Alex Skočdopole a jeho telefonní číslo je 666 456 654.

Scénář 3

Máte před sebou záznam o nalezeném nedostatku a předal jste záznam Panu X, s nejpozdějším termínem opravy 12.12.2023 na vyřešení. Potřebujete si informace zapsat do systému abyste na něj nezapomněli.

Údaje na protokolu: Vybavení J212674, Zařízení DRAEGER, Závada: Nefunkční spínač, zakázka 7212345678.

Scénář 4

Minulý měsíc proběhl zápis provedených revizí do systému a bylo nalezeno několik nedostatků. Přišel vám email, že už se nedostatky na strojích J211829 a J207709 vyřešily.

Scénář 5

Dostal se vám do ruky nový provozní deník pro tlakovou nádobu 2012/13392. Zaveďte deník do systému (název deníku odpovídá konvenci podle předešlých názvů) a nezapomeňte přiřadit i nádobu.

Scénář 6

Zapište nově provedené revize odstředivek. Revize 1-roční byla provedena 1.11.2023. Jedná se o zařízení s výrobními čísly: 10-911.226, 10-911.260, 118761, 24-801.155.

Scénář 7

Přijela revizní firma Mikulík provést revize na TNS zařízeních. Bohužel si firma zapoměla vytisknutý seznam. Vytiskněte tedy seznam zařízení, na kterých se budou revize provádět, tak jak tomu bylo ve starém systému.

Scénář 8

Proveďte (fiktivní) objednávku revizí na Uhlovodíková zařízení, pro která je to potřeba. (Pouze si připravte email nebo jen potřebná data, jako při reálném objednávání.)

Extra

Podívejte se do dalších částí systému, které byste mohl používat, které jste nepoužil v průběhu procházení připravených scénářů. Jedná se například o opravované protokoly u zařízení TNS, způsob zobrazení obsahu provozních deníků u TNS. Log, do kterého se můžete podívat pro kontrolu provedených změn. Stránku kontaktů firem, kde se určuje interval kolik měsíců dopředu se zobrazí revize. Nebo na kategorie, se kterými pracujete.

Respondent 1

Dotazník před testováním

Pohlaví: Muž

Věk: 42

Znalost domény (revizí): vysoká

Role v systému: admin - znalost dat v databázi

Úroveň práce s počítačem: expert

Čím trávíte v systému nejvíce času? Hledání špatně provedených změn v logu nebo zjišťování informací z dat pomocí SQL dotazů.

Dotazník po testování

1. Jak hodnotíte možnost zobrazení všech zařízení pro jednotlivé kategorie.
 - Tato možnost je fajn pro kontrolu a přehled všech zařízení v tématu. Bude vhodné pro roční statistiky aktivních zařízení pro oddělení HSE.
2. Jak se vám líbí způsob vyhledávání podle vybavení a zařízení na kartě zařízení?
 - V porovnání s předchozí aplikací je možnost vyhledání prakticky o něco rychlejší protože je hned po ruce.
3. Jak hodnotíte stránku se všemi informacemi ke stroji (karta stroje)?
 - Přehledná, obsahuje všechny informace, které potřebujete.
4. Jak hodnotíte způsob filtrování na stránce TNS? Případně, máte nějaké návrhy na změny?
 - Filtrace je řešena jednoduše pomocí tlačítek a plně vyhovuje potřebám daného tématu.
5. Jak hodnotíte způsob editace a vytváření pomocí formulářů? Existuje více způsobů jak editovat záznamy, například možnost editace přímo v řádku tabulky.
 - V pořádku, je vyhovující. Zobrazení karty umožní lepší přehled informací.
6. Jak hodnotíte možnost editace přímo ze seznamu zařízení, která ve starém systému nebyla.
 - Tato možnost je zlepšení přístupu. V některých případech může zjednodušit práci v aplikaci.
7. Chybí vám něco v novém systému?
 - Zatím je těžké zhodnotit vše po krátkém seznámení s aplikací. Po delším období užívání aplikace se ukáže více. Zatím vyhovuje požadavkům.
8. Pracovalo se vám v novém systému lépe?
 - V systému se mi pracuje dobře, jen si musím zvyknout na změnu.
9. Přejde vám v systému něco navíc?
 - Ne
10. Co se vám na novém systému líbí?
 - Líbí se mi že se jedná o webovou aplikaci, kde update a zmeny aplikace proběhnou jednoduše pro všechny uživatele. To bylo u MS Access problematičtější. Aplikace také nabízí více možností a to bude do budoucna přínosem.

11. Máte nějaké další připomínky nebo návrhy?

- Na kartě strojů by mohly být podbarvené platné a neplatné revize.
- Rád bych měl možnost exportu nadcházejících revizí přímo do excelu, formátovanou se zvýrazněním propadlých a se správnými velikostmi sloupců. Dále bych více zvýraznil na kterém řádku jsem v tabulce.

12. Potřebujete nějaké roční exporty nebo jiné informace, které pravidelně nebo nepravidelně často získáváte nestandardním způsobem ze staré aplikace. (prosím uveďte, které jsou pravidelné operace)

- Ano, oddělení HSE a FCM se pravidelně doptávají na data ohledně počtu provedených revizí aktivních zařízení a jiných informací k daným tématům. Přesné info ale zjistíte od kolegů.

Poznámky z průběhu testování

Scénář 1

Zjištěno, že revize na druhé nádobě není platná. Rád by měl podbarvení revizí.

Scénář 2

Správně našel KST kontakty. V seznamu je hůř vidět řádek na kterém je myš.

Scénář 3

Ve scénáři chyběli "informace na protokolu".

Aplikace spadla při vytváření nedostatku. (po testování hned opraveno)

Scénář 4 OK

Scénář 5

Správně našel deníky. Označení deníku si myslí, že by se mělo generovat automaticky.

Správně šel na kartu TNS ale už si nevšiml, že pole je zamčené. Raději by neumožnil rozbalit zamčené pole.

Scénář 6

Revize vyplňoval přes karty. Chyběla mu možnost vložení zkopírovaného datumu.

Scénář 7 OK

Scénář 8

Použil tlačítko zkopírovat tabulku - OK.

Tabulka v excelu měla malé sloupce. Uvítal by export do excelu s lepším formátem sloupců..

Extra OK

Respondent 2

Dotazník před testováním

Pohlaví: Muž

Věk: 22

Znalost domény revizí: nízká

Role v systému: se systémem nepracuji, setkal jsem se s ním v minulosti, kdy jsem zadával

Úroveň práce s počítačem: středně znalý

Čím trávíte v systému nejvíce času? se systémem nepracuji

Dotazník po testování

Pokud nevíte co odpovědět, zkuste se zamyslet jak se to dělalo ve staré aplikaci, jestli je to pro vás lepší nově, nebo se vám pracovalo lépe po staru.

1. Jak hodnotíte možnost zobrazení všech zařízení pro jednotlivé kategorie.
 - Pozitivně.
2. Jak se vám líbí způsob vyhledávání podle vybavení a zařízení na kartě zařízení?
 - Je to užitečná funkce, která dokáže usnadnit práci s vyhledáváním.
3. Jak hodnotíte stránku se všemi informacemi ke stroji (karta stroje)?
 - Velmi pozitivně
4. Jak hodnotíte způsob filtrování na stránce TNS? Případně, máte nějaké návrhy na změny?
 - Pozitivně. Žádné změny mě nenapadají.
5. Jak hodnotíte způsob editace a vytváření záznamech o zařízeních pomocí formulářů? Existuje více způsobů jak editovat záznamy. Například možnost editace přímo v řádku tabulky (ta v tomto případě, kde je dat hodně nevyhovuje).
 - Hodnotím kladně editaci povolenou tlačítkem a nutnost uložení. Společně s vyhledáváním v samostatném poli se eliminuje velké množství chyb vzniklých při zadávání a vyhledávání jednotlivých passportů.
6. Jak hodnotíte možnost otevření editace přímo ze seznamu zařízení, která ve starém systému nebyla.
 - Není třeba otevírat kartu zařízení - užitečná funkce.
7. Chybí vám něco v novém systému?
 - V tuto chvíli ne, systém se mi jeví jako zdařilý.
8. Pracovalo se vám v novém systému lépe?
 - Ano
9. Přijde vám v systému něco navíc (zbytečné)?
 - Ne
10. Co se vám na novém systému líbí?
 - Nové možnosti zadávání a ošetření proti zadání neplatných dat.
11. Máte nějaké další připomínky nebo návrhy?
 - Nic mě v tuto chvíli nenapadá.
12. Potřebujete nějaké roční exporty nebo jiné informace, které pravidelně nebo nepravidelně často získáváte nestandardním způsobem ze staré aplikace. (prosím uveďte, které jsou pravidelné operace)
 - Tyto exporty dat nevyužívám.

Poznámky z průběhu testování

Scénář 1

Po zjištění že protokol tns je v názvu klikal i u plynu na název

Scénář 2

Hnedka šel do menu extra ale přehlédl kst kontakty

Přidávání do seznamu prvně hledal na jeho konci, ale pak tlačítko viděl hned

Zvýraznění řádku v tabulce by mohlo být výraznější

Scénář 3

Z důvodu neznalosti systému hledal nedostatky u kategorie (později přidán odkaz na filtrované nedostatky v horním menu všech kategorií)

Po napsání vybavení klikl mimo místo výběru - nepotvrdil výběr

Uvítal že nejde odeslat formulář bez vybavení

Scénář 4

Snažil se kliknout přímo v tabulce ale pak hned šel do edit formuláře

Scénář 5

Šel správně do TNS, ale do seznamu, kde odkaz na deníky není.

Z neznalosti systému ho nenapadlo podívat se na stránku karet TNS.

Z karty už se dostal na deníky, deník přidal. Snažil se nádobu přidat do deníků na stránce deníků

Po otevření karty ihned neviděl proč mu nejde zadávat, ale po ujasnění se mu funkce zamčených polí líbila a tlačítko pro odemčení našel ihned.

Moje poznámka - zvážít zobrazování i názvu deníků v selectu

Scénář 6

U odstředivek byl prvně v seznamu ale nepoužil zadávání z něj. Šel na kartu, zbytek OK

Scénář 7 OK

Tisk udělal přes export, ale i ten je správně

Extra OK

Log se líbil, chápal

Zajímavost -

Chybovost ručního zápisu vs copy paste při zápisu protokolu - testovaný používá copy paste

Respondent 3

Dotazník před testováním

Pohlaví:muž

Věk:39

Znalost domény revizí: střední

Role v systému: zadávání revizí, objednávání revizí, kontrola

Úroveň práce s počítačem: středně znalý

Čím trávíte v systému nejvíce času?

Kontrolou revizních protokolů, kontrolou neprovedených revizí, tvorba plánu revizí.

Aktualizací dat.

Dotazník po testování

Pokud nevíte co odpovědět, zkuste se zamyslet jak se to dělalo ve staré aplikaci, jestli je to pro vás lepší nově, nebo se vám pracovalo lépe po staru.

1. Jak hodnotíte možnost zobrazení všech zařízení pro jednotlivé kategorie.
 - Použitelné pro revizní firmu, bude mít přehled celkového počtu zařízení. I pro nákup.
2. Jak se vám líbí způsob vyhledávání podle vybavení a zařízení na kartě zařízení?
 - Přehledné, bez problémů.
3. Jak hodnotíte stránku se všemi informacemi ke stroji (karta stroje)?
 - Pro audit ideální, okamžitě vidíme na stroji, jaké se na něm dělají revize. A které jsou v pořádku a které jsou propadlé,
4. Jak hodnotíte způsob filtrování na stránce TNS (seznam TNS)? Případně, máte nějaké návrhy na změny?
 - Vizualizace filtrů v pořádku, přehledné.
5. Jak hodnotíte způsob editace a vytváření záznamů o zařízeních pomocí formulářů? Existuje více způsobů jak editovat záznamy. Například možnost editace přímo v řádku tabulky (ta v tomto případě, kde je dat hodně nevyhovuje).
 - Způsob editace mi vyhovuje, nic jiného mě nenapadá.
6. Jak hodnotíte možnost otevření editace přímo ze seznamu zařízení, která ve starém systému nebyla.
 - Oproti starší verzi, zjednodušení a přehlednější.
7. Chybí vám něco v novém systému?
 - Možnost filtrování v seznamech zařízení, podle čísla vybavení.
8. Pracovalo se vám v novém systému lépe?
 - Ano, líbí se mi karta stroje, kde jsou všechny informace na jednom místě.
9. Přejde vám v systému něco navíc (zbytečné)?
 - Ne
10. Co se vám na novém systému líbí?
 - Možnost editace zařízení přímo ze seznamu zrychlí práci.
11. Máte nějaké další připomínky nebo návrhy?
 - U kategorií, které spravuji bych zvolil jiný způsob řazení, konkrétně podle datumu revize.
 - Navíc bych u těch kategorií také uvítal informaci o tom, jestli jsem revizi objednal, podobně jako u Plynových zařízení.

12. Potřebujete nějaké roční exporthy nebo jiné informace, které pravidelně nebo nepravidelně často získáváte nestandardním způsobem ze staré aplikace. (prosím uveďte, které jsou pravidelné operace)

- Ne

Poznámky z průběhu testování

Scénář 1

Neviděl kde je protokol, šel na seznam tns. Po odkázání na stránku stoje už našel rychle. Nenašel na první pohled odkazy ke stažení podle modré barvy textu

Scénář 2

Přidání střediska se snažil udělat přes kartu stroje klikem na nákladové středisko - neznalost systému
Po nalezení kst kontaktu hledal přidání na konci seznamu. Hledal způsob filtrace. Nakonec tlačítko našel. Edit tlačítko našel ihned.

Scénář 3

Nedostatky našel. Datum zadával ručně. Neznalostí systému chtěl zaškrtnout OK při vytváření.

Scénář 4

Snažil se klikat OK v tabulce. Opět hledal možnosti filtrování.

Scénář 5

Šel prvně na seznam, pak karta stroje, nakonec našel v menu TNS. Vytvoření OK. Přidání nádoby hledal u deníků. Po odkázání na kartu zařízení si nevšiml zamčených polí. Snažil se vracet zpět pro opětovné vyhledání, toho co měl vyhledané (bude vyřešeno).

Scénář 6

Odstředivky zadával přes kartu, vyhledával ručně (bez copy paste)

Scénář 7

Šel do seznamu, chvíli hledal co filtrovat, potom se snažil filtrovat datумы. když šlo o tisk to měl hned

Scénář 8

Zkopíroval OK, formátování v Excelu zvládl jako expert

Poznámky na změny přímo u témat které spravuje

Platnosti revizí - plyn a tem revize na den, ostatní stačí na mesic

Freony filtrovat podle datumu, přidat kontakt na mistra do seznamu

Do karet freonů přidat pole s intervalem (jelikož se počítá automaticky tak needitovatelné)

V kartě stroje odstranit změnu myši při najetí na název inputu

Chyba na stránce Nedostatky po filtraci všeho - přestane fungovat filtrace na kategorie

Přidání informace o objednání a poznámku k objednavce ve většině kategorií. Po změně datumu poslední revize potom vymazat poznámku k objednavce a checkbox

Změny názvů sloupců - Revize změnit na Nadch revize

Sjednocení řazení sloupců -> Kst, vybavení, tech místo, info o místě, výrobní číslo, nadch revize, typ, skupina, firma, poznám, obj, obj_pozn

Při filtraci všeho zachovat sloupce jako u nadcházejících

Respondent 4

Dotazník před testováním

Pohlaví: muž

Věk: 55

Znalost domény revizí: střední

Role v systému: zadávání revizí, objednávání revizí, kontrola

Úroveň práce s počítačem: středně znalý

Čím trávíte v systému nejvíce času? kontrola správného zadání po praktikantech

Dotazník po testování

Pokud nevíte co odpovědět, zkuste se zamyslet jak se to dělalo ve staré aplikaci, jestli je to pro vás lepší nově, nebo se vám pracovalo lépe po staru.

1. Jak hodnotíte možnost zobrazení všech zařízení pro jednotlivé kategorie.
 - Po seřazení podle dohody bude použitelné.
2. Jak se vám líbí způsob vyhledávání podle vybavení a zařízení na kartě zařízení?
 - Vyhledávání je v pohodě.
3. Jak hodnotíte stránku se všemi informacemi ke stroji (karta stroje)?
 - Dobrý.
4. Jak hodnotíte způsob filtrování na stránce TNS (seznam TNS)? Případně, máte nějaké návrhy na změny?
 - Chybí mi možnost přímo vypsání daného měsíce v daném roce.
5. Jak hodnotíte způsob editace a vytváření záznamů o zařízeních pomocí formulářů? Existuje více způsobů jak editovat záznamy. Například možnost editace přímo v řádku tabulky (ta v tomto případě, kde je dat hodně nevyhovuje).
 - OK.
6. Jak hodnotíte možnost otevření editace přímo ze seznamu zařízení, která ve starém systému nebyla.
 - Super.
7. Chybí vám něco v novém systému?
 - Ne.
8. Pracovalo se vám v novém systému lépe?
 - Po zvyku a doladění si myslím že to bude lepší.
9. Přejde vám v systému něco navíc (zbytečné)?
 - Ne.
10. Co se vám na novém systému líbí?
 - Karta stroje a editace ze seznamu.
11. Máte nějaké další připomínky nebo návrhy?
 - SHZ sezaňamy řazení podle firmy.
12. Potřebujete nějaké roční exporty nebo jiné informace, které pravidelně nebo nepravidelně často získáváte nestandardním způsobem ze staré aplikace. (prosím uveďte, které jsou pravidelné operace)
 - Počet aktivních TNS, u SHZ exporty pro firmy.

Poznámky z průběhu testování

Scénář 1

Chtěl hledat u svého tématu TNS
Musel jsem ujasnit že se scénář týká karty stroje
Protokol TNS našel,

Scénář 2

Hledal u kontaktu firem, pak našel pod extra
Edit našel, přidání hledal dole, po chvíli našel
Bavili jsme se o halách u kontaktů - to je potřeba probrat i s ostatními

Scénář 3

Našel, tlačítko na přidání už taky bylo ihned
Uteklo mu vybavení (klikl mimo bez potvrzení)

Scénář 4 Ok

Scénář 5

Info mezi průchodem testu
Na kartě zařízení by rád změnil "zařízení" na TNS a výrobní číslo TNS
PS tlak místo max přetlak
Velké L jako litry
Deník našel, vytvořil ale při vrácení se mu nechtělo znovu vyhledávat zařízení
Snažil se deník přidat přes tlačítko
To že jsou pole zablokované si také nevšiml

Scénář 6

Šel do seznamu ale hledal tam kde filtrovat
Na shz byl datum anglicky.
Serial nr přepsat na výrobní číslo

Scénář 7 Ok

Scénář 8

Objednávku tvoří pro jednotlivé firmy zvlášť, rád by měl export přímo do excelu, kde už může jednoduše filtrovat. Nebo filtraci v aplikaci s následným exportem do excelu.

V průběhu testu se hodně doptával na věci okolo, nesnažil se scénářem projít sám.

Poznámky na změny přímo u témat které spravuje

TNS

Nucené zobrazí může pryč
Kanban i export možnost se dívat přímo na měsíc podle výběru
Provozní rev změnit na provozní nasl
Aby šlo zjistit počet aktivních zařízení

Respondent 5

Dotazník před testováním

Pohlaví: Muž

Věk: 20

Znalost domény revizí: nízká

Role v systému: kontrola

Úroveň práce s počítačem: pokročilý

Čím trávíte v systému nejvíce času? kontrola

Dotazník po testování

Pokud nevíte co odpovědět, zkuste se zamyslet jak se to dělalo ve staré aplikaci, jestli je to pro vás lepší nově, nebo se vám pracovalo lépe po staru.

1. Jak hodnotíte možnost zobrazení všech zařízení pro jednotlivé kategorie.
 - Z hlediska přehlednosti velký pokrok oproti staré verzi.
2. Jak se vám líbí způsob vyhledávání podle vybavení a zařízení na kartě zařízení?
 - Vyhledávání podle vybavení je oproti staré verzi o dost přehlednější.
3. Jak hodnotíte stránku se všemi informacemi ke stroji (karta stroje)?
 - Tuto možnost hodnotím velmi kladně, určitě ušetří spoustu času.
4. Jak hodnotíte způsob filtrování na stránce TNS (seznam TNS)? Případně, máte nějaké návrhy na změny?
 - Filtrování je oproti staré verzi přehlednější a snažší na používání.
5. Jak hodnotíte způsob editace a vytváření záznamů o zařízeních pomocí formulářů? Existuje více způsobů jak editovat záznamy. Například možnost editace přímo v řádku tabulky (ta v tomto případě, kde je dat hodně nevyhovuje).
 - Nenapadá mě nic jiného
6. Jak hodnotíte možnost otevření editace přímo ze seznamu zařízení, která ve starém systému nebyla.
 - Super
7. Chybí vám něco v novém systému?
 - Nic mi nechybělo
8. Pracovalo se vám v novém systému lépe?
 - Ano
9. Přejde vám v systému něco navíc (zbytečné)?
 - Ne
10. Co se vám na novém systému líbí?
 - Celkové zpracování, přehlednost a jednoduchost pro nové uživatele.
11. Máte nějaké další připomínky nebo návrhy?
 - Ne
12. Potřebujete nějaké roční exporty nebo jiné informace, které pravidelně nebo nepravidelně často získáváte nestandardním způsobem ze staré aplikace. (prosím uveďte, které jsou pravidelné operace)
 - Nepoužívám

Poznámky z průběhu testování

Scénář 1

Neví kde má hledat, šel na TNS, použil ctrl+F - našel protokol tlakové láhve

Po radě našel kartu stroje, protokoly viděl ihned

Scénář 2

Kst kontakty našel po upozornění na horní menu

Přidání našel hned

Ptal se na mezery v zadávání telefonu

Používá ctrl+F

Scénář 3

Ihned našel, trochu hledal co kam zapsat ze scénáře

Scénář 4 Ok

Scénář 5

Hledal jak se dostat na kartu , otevřel si protokol

Po připomenutí že musí deník vytvořit našel kde

Rozhodil ho neprázdný formulář po použití šipky zpět (to se stalo poprvé, u jiných testů se vždy formulář vyprázdnil)

Pak jsem musel upozornit na zámek polí

Scénář 6

Byl na seznamu, šel na kartu

Datum má všude anglicky (důvod neznámý - prohlížeč má česky, windows anglicky ale s českým datumem)

Scénář 7 Ok

Scénář 8

Navíc deaktivoval zařízení co bylo na odstaveném stroji - OK

Respondent 6

Dotazník před testováním

Pohlaví: Muž

Věk: 21

Znalost domény revizí: střední

Role v systému: zadávání revizí / kontrola

Úroveň práce s počítačem: expert

Čím trávíte v systému nejvíce času? Se systémem pracuji málokdy

Dotazník po testování

Pokud nevíte co odpovědět, zkuste se zamyslet jak se to dělalo ve staré aplikaci, jestli je to pro vás lepší nově, nebo se vám pracovalo lépe po staru.

1. Jak hodnotíte možnost zobrazení všech zařízení pro jednotlivé kategorie.
 - Kladně
2. Jak se vám líbí způsob vyhledávání podle vybavení a zařízení na kartě zařízení?
 - Tuto možnost jsem nevyužil, ale je to ideální.
3. Jak hodnotíte stránku se všemi informacemi ke stroji (karta stroje)?
 - Krásný přehled informací o provedených revizích
4. Jak hodnotíte způsob filtrování na stránce TNS (seznam TNS)? Případně, máte nějaké návrhy na změny?
 - Lepší provedení mě nenapadá
5. Jak hodnotíte způsob editace a vytváření záznamů o zařízeních pomocí formulářů? Existuje více způsobů jak editovat záznamy. Například možnost editace přímo v řádku tabulky (ta v tomto případě, kde je dat hodně nevyhovuje).
 - Rozhodně jednodušší než cv předešlém programu
6. Jak hodnotíte možnost otevření editace přímo ze seznamu zařízení, která ve starém systému nebyla.
 - Tato možnost se mi líbí, urychluje čas
7. Chybí vám něco v novém systému?
 - Ne
8. Pracovalo se vám v novém systému lépe?
 - Ano
9. Přijde vám v systému něco navíc (zbytečné)?
 - Spíše ne, jsou věci které osobně nevyužiji ale myslím si že ostatní pracovníci ano
10. Co se vám na novém systému líbí?
 - Jednoduchost, přehlednost
11. Máte nějaké další připomínky nebo návrhy?
 - Možnost odepsání nedostatků naráz
12. Potřebujete nějaké roční exporty nebo jiné informace, které pravidelně nebo nepravidelně často získáváte nestandardním způsobem ze staré aplikace. (prosím uveďte, které jsou pravidelné operace)
 - Nepotřebuji

Poznámky z průběhu testování

Scénář 1

Ok možná titulky přes odkazy na stažení protokolů

Scénář 2

Po upozornění na horní menu OK

Scénář 3 Ok

Scénář 4

Ok, klikal do tabulky

Navrhl tlačítko na vyřešení všech nedostatků naráz, nebo by to mohlo odemknout OK checkboxy přímo v tabulce

Scénář 5

Menu otevřel ale nevšiml si deníků

V seznamu všech našel přes ctrl+F

Pak se do deníků dostal přes formulář

Hledal přidání nádoby do deníku ona stránce deníků, pak to udělal opět přes seznam.

Tlačítka odemknout pole si všiml v průběhu

Scénář 6 Ok, úpravy dělal v seznamu

Scénář 7 Ok

Scénář 8 Ok - expert s prací v excelu

Respondent 7

Dotazník před testováním

Pohlaví: Muž

Věk: 46

Znalost domény vysoká

Role v systému: zadávání revizí, objednávání revizí, kontrola, úprava údajů

Úroveň práce s počítačem: pokročilý

Čím trávíte v systému nejvíce času? objednávání revizí, ukládání revizních zpráv

Dotazník po testování

Pokud nevíte co odpovědět, zkuste se zamyslet jak se to dělalo ve staré aplikaci, jestli je to pro vás lepší nově, nebo se vám pracovalo lépe po staru.

1. Jak hodnotíte možnost zobrazení všech zařízení pro jednotlivé kategorie.
 - OK, přehledné.
2. Jak se vám líbí způsob vyhledávání podle vybavení a zařízení na kartě zařízení?
 - Dobré, ale chce to sjednotit pro jednotlivé revize umístění políčka vybavení
3. Jak hodnotíte stránku se všemi informacemi ke stroji (karta stroje)?
 - Výborné, je přehledně zobrazen stav všech revizí na stroj
4. Jak hodnotíte způsob filtrování na stránce TNS (seznam TNS)? Případně, máte nějaké návrhy na změny?
 - Nepoužívám
5. Jak hodnotíte způsob editace a vytváření záznamů o zařízeních pomocí formulářů? Existuje více způsobů jak editovat záznamy. Například možnost editace přímo v řádku tabulky (ta v tomto případě, kde je dat hodně nevyhovuje).
 - Bylo by vhodné alespoň pro CheckBox, jestli bylo objednáno
6. Jak hodnotíte možnost otevření editace přímo ze seznamu zařízení, která ve starém systému nebyla.
 - Výborné, to jsem postrádal
7. Chybí vám něco v novém systému?
 - Není možnost ve formuláři při zadávání výběr z platných čísel vybavení
 - Filtrování jak v databázi MS Access
 - U freonů datum založení revizí a datum deaktivace zařízení/revize
8. Pracovalo se vám v novém systému lépe?
 - Ano, rychlá odezva systému
9. Přejde vám v systému něco navíc (zbytečné)?
 - Asi nic
10. Co se vám na novém systému líbí?
 - Přehlednost, intuitivnost
11. Máte nějaké další připomínky nebo návrhy?
 - Ukládání PDF, filtrace položek, generování e-mailů
12. Potřebujete nějaké roční exporty nebo jiné informace, které pravidelně nebo nepravidelně často získáváte nestandardním způsobem ze staré aplikace. (prosím uveďte, které jsou pravidelné operace)
 - Ano, potřebuji vědět, kolik kg freonů nám ročně přibylo a kolik ubylo

Poznámky z průběhu testování

Scénář 1 OK

Scénář 2 OK

Scénář 3 OK, rád by automaticky vybral kategorii z filtru

Scénář 4 OK

Scénář 5

Šel na seznam, hned do karty, deník vytvořil, pak přidal. Chtěl by vyhledávat v deník selectu podle názvu

Scénář 6 OK

Scénář 7 OK

Scénář 8 OK

Poznámky na změny přímo u témat které spravuje

U freonů přidat datum kdy se záznam vytvořil a datum deaktivace

Na freony není potřeba info o objednavce (zatím přidat, pak se uvidí, jestli se fakt nebude používat)

Do návodu zapsat kde se upravuje počítání intervalu pro freony

Formuláře

Možnost zadávání jen existujících čísel vybavení

Sjednotit řazení polí ve všech kategoriích

Obj poznámka nechat ve formuláři otevřenou pro změny rovnou

Vylepšení - do budoucna

Z formuláře možnost vytvořit nedostatek

Rád by měl možnosti uložení PDF - na správné místo, pod správným jménem

Export pro objednání (rovnou email pro freony)

Nalezené buggy

Nedostatky výchozí filtr si pamatuje filtrované vše, i po prvním načtení filtruje vše

Jednoduché, logické změny

V seznamu zvýraznit vybavení se zrušeným statusem

V seznamu nezvýrazňovat datумы u neaktivních

Respondent 8

Dotazník před testováním

Pohlaví: Muž

Věk: 44

Znalost domény revizí: střední

Role v systému: zadávání revizí, objednávání revizí, kontrola

Úroveň práce s počítačem: středně znalý

Čím trávíte v systému nejvíce času? Aktualizace dat

Dotazník po testování

Pokud nevíte co odpovědět, zkuste se zamyslet jak se to dělalo ve staré aplikaci, jestli je to pro vás lepší nově, nebo se vám pracovalo lépe po staru.

1. Jak hodnotíte možnost zobrazení všech zařízení pro jednotlivé kategorie.
 - Kladně
2. Jak se vám líbí způsob vyhledávání podle vybavení a zařízení na kartě zařízení?
 - Líbí se.
3. Jak hodnotíte stránku se všemi informacemi ke stroji (karta stroje)?
 - Je tam vše podstatné, na jednom místě
4. Jak hodnotíte způsob filtrování na stránce TNS (seznam TNS)? Případně, máte nějaké návrhy na změny?
 - Dobré, určitě se využije.
5. Jak hodnotíte způsob editace a vytváření záznamů o zařízeních pomocí formulářů? Existuje více způsobů jak editovat záznamy. Například možnost editace přímo v řádku tabulky (ta v tomto případě, kde je dat hodně nevyhovuje).
 - Nic jiného mě nenapadá.
6. Jak hodnotíte možnost otevření editace přímo ze seznamu zařízení, která ve starém systému nebyla.
 - Líbí se, urychluje práci.
7. Chybí vám něco v novém systému?
 - Ne.
8. Pracovalo se vám v novém systému lépe?
 - Je potřeba si zvyknout, ale vypadá to dobře.
9. Přijde vám v systému něco navíc (zbytečné)?
 - Ne.
10. Co se vám na novém systému líbí?
 - Celkově je aplikace kompaktní a dá se v ní lépe vyhledávat.
11. Máte nějaké další připomínky nebo návrhy?
 - Líbila se mi stejná informace o objednavce jako u plynových zařízení i u odstředivek.
12. Potřebujete nějaké roční exporty nebo jiné informace, které pravidelně nebo nepravidelně často získáváte nestandardním způsobem ze staré aplikace. (prosím uveďte, které jsou pravidelné operace)
 - Nepotřebuji.

Poznámky z průběhu testování

Scénář 1

Chtěl hledat v kategoriích, Protokoly našel

Scénář 2

Po chvílce našel, přidat našel hned, edit ok

Scénář 3

Přes draeger nedostatky, přidání ok

Scénář 4 Ok, klikal v tabulce

Scénář 5

Hledal v extra, hned TNS přes menu do deníků, přidal ok, hledal jak přidat nádobu do deníku na stránce deníků, po otevření karty, tlačítkem na deníky zpět na stránku deníků, tápal, ale našel

Scénář 6 přes kartu, OK, zkusil I seznam

Scénář 7 Ok

Scénář 8 Ok

Bibliografie

1. STÁTNÍ ÚŘAD INSPEKCE PRÁCE. *Vyhrazená technická zařízení (tzv. VTZ)* [online]. 2023. [cit. 2023-07-01]. Dostupné z: https://www.suip.cz/informace-z-bozp-a-vtz/-/asset_publisher/Nkpp0dEZcc66/content/vyhrazena-technicka-zarizeni.
2. VÝZKUMNÝ ÚSTAV BEZPEČNOSTI PRÁCE, V. V. I. *Vyhrazená technická zařízení* [online]. 2023. [cit. 2023-07-01]. Dostupné z: <https://zsbozp.vubp.cz/vyhrazena-technicka-zarizeni>.
3. GEOFFREY, Elliott. *Global Business Information Technology*. Pearson Education, 2004.
4. PRESSMAN, Roger S. *Software Engineering: A Practitioner's Approach*. McGraw-Hill Education, 2014.
5. SCHWABER, Ken; SUTHERLAND, Jeff. *The Scrum Guide*. Scrum.org, 2017.
6. SOMMERVILLE, Ian. *Software Engineering*. Addison-Wesley, 2011.
7. BOEHM, Barry W. A Spiral Model of Software Development and Enhancement. *ACM SIGSOFT Software Engineering Notes*. 1988, roč. 11, č. 4, s. 14–24.
8. POPPENDIECK, Mary; POPPENDIECK, Tom. *Lean Software Development: An Agile Toolkit*. Addison-Wesley, 2003.
9. KRUCHTEN, Philippe. *The Rational Unified Process: An Introduction*. Addison-Wesley, 2000.
10. HUMBLE, Jez; FARLEY, David. *Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation*. Addison-Wesley, 2010.
11. ANON. *Vodopádový model (Waterfall Model)* [online]. 2015. [cit. 2023-08-08]. Dostupné z: <https://managementmania.com/cs/vodopadovy-model-waterfall-model>.
12. JIŘÍ, Mlejnek. Přednáška 3. *Analýza a sběr požadavků - případy užití* [online]. 2020 [cit. 2023-07-01]. Dostupné z: https://moodle-vyuka.cvut.cz/pluginfile.php/308910/course/section/46035/3_RequirementsEngineering.pdf.
13. Software Engineering. *Classification of Software Requirements*. [Online]. 2018 [cit. 2023-07-01]. Dostupné z: <https://www.geeksforgeeks.org/software-engineering-classification-of-software-requirements>.
14. YOUNG, Ralph R. *The Requirements Engineering Handbook*. Norwood, MA, USA: Artech House, 2003.
15. ROBERTSON, S.; ROBERTSON, J. *Mastering the Requirements Process: Getting Requirements Right*. Pearson Education, 2012. ISBN 9780132942843. Dostupné také z: <https://books.google.de/books?id=yE91LgrpaHsC>.

16. WIEGERS, Karl E.; BEATTY, Joy. *Software Requirements*. 3rd. Redmond, WA, USA: Microsoft Press, 2013.
17. BARBORA, Obstova. Uživatelské rozhraní. [B.r.]. Dostupné také z: https://wikisofia.cz/wiki/U%C5%BEivatelsk%C3%A9_rozhran%C3%AD. [online] 2013 [Cit. 2023-07-05].
18. JOSEF, Pavlíček. User Interface Design. [B.r.]. Dostupné také z: https://docs.google.com/presentation/d/10IjuumeQ0b-t3YZkqb8J7BNos8K-usoYgXj1xmQIkuA/edit#slide=id.g99ecd62e4f_2_0. [online] 2021 [Cit. 2023-07-05].
19. AGIONET S.R.O. *FACMAN* [online]. 2022. [cit. 2023-06-22]. Dostupné z: <https://www.facman.cz>.
20. ASW-CZ S.R.O. *Evis PRO* [online]. 2023. [cit. 2023-06-22]. Dostupné z: <https://www.evispro.cz/evispro>.
21. ACONTE, S.R.O. *Online Revize* [online]. 2017. [cit. 2023-06-22]. Dostupné z: <https://online-revize.cz/>.
22. CIVOP S.R.O. *Revize a kontroly technických zařízení* [online]. 2023. [cit. 2023-06-22]. Dostupné z: <https://www.civop.cz/revize-technicky-zarizeni/>.
23. SUN SOLUTIONS S.R.O. *EcoSun CMMS* [online]. 2017. [cit. 2023-06-22]. Dostupné z: <https://www.ecosun.one/cz/cmms>.
24. FORM SOLUTION S.R.O. *evidujsirevizi.cz* [online]. 2023. [cit. 2023-06-22]. Dostupné z: <https://www.evidujsirevizi.cz/uvod>.
25. INSIO SOFTWARE S.R.O. *INSIO Servisní systém* [online]. 2023. [cit. 2023-06-22]. Dostupné z: <https://insio.cz/aplikace/servisni-system/>.
26. Q-LANYS S.R.O. *Revize zařízení, strojů a budov* [online]. 2023. [cit. 2023-06-22]. Dostupné z: <https://qlanys.cz/cz/revize-zarizeni-a-budov>.
27. LÍDA, Dubinská. *Co je wireframe webu, proč ho potřebujeme a jak ho vytvořit*. [B.r.]. Dostupné také z: <https://www.rascasone.com/cs/blog/co-je-wireframe-predstavujeme-5-duvodu-proc-je-pro-klienty-drateny-model-dulezity>. [online] 2021 [Cit. 2023-07-22].
28. FIGMA. Figma - Online UI Design and Prototyping Tool. 2023. Dostupné také z: <https://www.figma.com/>. [online] 2021 [Cit. 2023-07-25].
29. FOUNDATION, Nette. Nette framework. [B.r.]. Dostupné také z: <https://nette.org/cs/>. [online] 2008 [Cit. 2023-12-20].
30. BORSKÝ, Jiří. NI-ADP Architektonické a návrhové vzory. [B.r.]. Dostupné také z: <https://courses.fit.cvut.cz/NI-ADP/materials/architectures/MVC.html>. [online] 2023 [Cit. 2023-12-20].
31. FOUNDATION, Nette. Presenters. [B.r.]. Dostupné také z: <https://doc.nette.org/cs/application/presenters>. [online] 2008 [Cit. 2023-12-20].
32. DUCKETT, Jon. *HTML and CSS: Design and Build Websites*. Wiley, 2011. ISBN 978-1118008188.
33. FLANAGAN, David. *JavaScript: The Definitive Guide*. O'Reilly Media, 2011.
34. MICROSOFT. *Visual Studio Code - Code Editing. Redefined*. Dostupné také z: <https://code.visualstudio.com/>. [online] 2023 [Cit. 2023-12-20].
35. ATLASSIAN. *Sourcetree - Free Git GUI for Mac and Windows*. Dostupné také z: <https://www.sourcetreeapp.com/>. [online] 2023 [Cit. 2023-12-20].
36. COMMUNITY, Git Development. *Git - Fast Version Control System*. Dostupné také z: <https://git-scm.com/>. [online] 2023 [Cit. 2023-12-20].

37. FOUNDATION, Apache Software. *Apache HTTP Server Documentation*. Dostupné také z: <https://httpd.apache.org/>. [online] 2023 [Cit. 2023-12-20].
38. RADEK, Kitner. Co je testování softwaru? [B.r.]. Dostupné také z: https://kitner.cz/testovani_softwaru/co-je-testovani_softwaru/. [online] 2021 [Cit. 2023-12-08].
39. MICHAL, Voják. Jak dělat uživatelské testování. [B.r.]. Dostupné také z: <https://designdev.cz/jak-delat-uzivatelske-testovani>. [online] 2020 [Cit. 2023-12-08].
40. JAKOB, Nielsen. Why You Only Need to Test with 5 Users. *Nielsen Norman Group*. [B.r.]. Dostupné také z: <https://www.nngroup.com/articles/why-you-only-need-to-test-with-5-users/>. [online] 2000 [Cit. 2023-12-08].
41. MICHAL, Voják. Příprava testu použitelnosti. [B.r.]. Dostupné také z: <https://designdev.cz/priprava-testu-pouzitelnosti>. [online] 2020 [Cit. 2023-12-08].
42. MICHAL, Voják. Průběh testu použitelnosti. [B.r.]. Dostupné také z: <https://designdev.cz/prubeh-testu-pouzitelnosti>. [online] 2020 [Cit. 2023-12-08].
43. MICROSOFT. Microsoft Teams. [B.r.]. Dostupné také z: <https://www.microsoft.com/cs-cz/microsoft-teams/group-chat-software>. [online] 2023 [Cit. 2023-12-08].
44. SAS, LEARNPULSE. Screenpresso - Image and Video screen capture. [B.r.]. Dostupné také z: <https://www.screenpresso.com/>. [online] 2023 [Cit. 2023-12-08].
45. INC., Docker. Docker Docs. [B.r.]. Dostupné také z: <https://docs.docker.com/>. [online] 2023 [Cit. 2023-12-10].
46. INC., Docker. Docker Hub. [B.r.]. Dostupné také z: https://hub.docker.com/_/php. [online] 2023 [Cit. 2023-12-10].

Obsah přiloženého média

readme.txt.....	stručný popis obsahu média
src	
├─ impl.....	zdrojové kódy implementace
├─ thesis.....	zdrojová forma práce ve formátu L ^A T _E X
text.....	text práce
├─ thesis.pdf.....	text práce ve formátu PDF
videos.....	videa z testování