

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER SYSTEMS

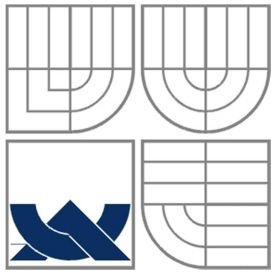
TECHNICKÁ PODPORA MANAGEMENTU ZMĚN

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

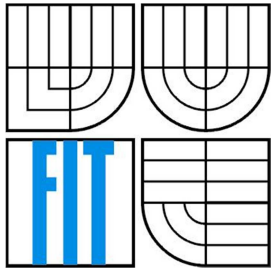
AUTOR PRÁCE
AUTHOR

Bc. MICHAL VANĚK

BRNO 2013



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER SYSTEMS

TECHNICKÁ PODPORA MANAGEMENTU ZMĚN TECHNICAL ASSISTANCE OF CHANGE MANAGEMENT SUPPORT

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR

Bc. MICHAL VANĚK

VEDOUCÍ PRÁCE
SUPERVISOR

doc. RNDr. JITKA KRESLÍKOVÁ, CSc.

BRNO 2013

Abstrakt

Tato diplomová práce popisuje zlepšení nástroje technické podpory managementu změn v reálné společnosti poskytovatele IT služeb. Hlavním cílem je analýza a implementace nástroje prezentující možná zlepšení. Práce zahrnuje úplný vývojový proces systému včetně sběru požadavků, analýzy a návrhu systému, jeho implementace. Sběr požadavků je rozšířen o analýzu problémů ve společnosti a hledání řešení, potřebných změn pro dosažení cílů společnosti. Cílem implementovaného systému je ověřit přínos nalezených řešení. Práce obnášela důkladné seznámení se s procesem managementu změn v dané společnosti a doporučeními, které na ně kladou nejlepší praktiky uvedené v knize ITIL.

V první části je popsána oblast managementu změn, knihy ITIL a vybrané praktiky. V druhé části je zaznamenána analýza současného stavu managementu změn a nástroje a jsou navrženy možnosti zlepšení v rámci nástroje. Tyto požadavky jsou dále analyzovány a zpracovány do specifikace požadavků, následně je navrženo a implementováno jejich řešení. V závěru je popsáno ověření přínosu pomocí experimentu nad implementovaným řešením, následuje shrnutí celé práce.

Abstract

The project inspects and discusses improvements of the technical assistance of Change Management support within IT service provider. The primary objective of the project is to analyse and implement a new software tool to display possible improvements. This project involves whole software development process including requirements collection, system analysis, design, implementation.

In the first part, the IT service management, especially Change Management area and ITIL methodology is described. In the second part, the analysis of current state of the process implementation and software tool is discussed and proposed opportunities for improvement. In the end, the results are summarized and possibilities of the future improvements are proposed.

Klíčová slova

Management změn, řízení změn, řízení IT služeb, ITSM, ITIL, nejlepší praktiky, neustálé zlepšování, MVP, Nette Framework

Keywords

Change management, IT Service management, ITSM, ITIL, best practices, continual improvement, MPV, Nette Framework.

Citace

Vaněk Michal: Technická podpora managementu změn, diplomová práce, Brno, FIT VUT v Brně, 2013

Technická podpora managementu změn

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně pod vedením doc. RNDr. Jitky Kreslíkové, CSc. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Michal Vaněk
29.5.2013

Poděkování

Děkuji doc. RNDr. Jitce Kreslíkové, CSc. za odborné vedení, konzultace, rady a připomínky, které umožnili dokončení mé diplomové práce.

© Michal Vaněk, 2013

Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.

Obsah

Obsah.....	1
1 Úvod.....	3
1.1 Cíle práce.....	3
1.2 Rozsah práce.....	3
1.3 Terminologie.....	3
1.4 Utajení některých údajů a fiktivní názvy.....	4
1.5 Rozložení tohoto dokumentu.....	4
1.6 Návaznost na jiné práce.....	4
2 Standardy řízení IT služeb.....	5
2.1 ITSM.....	5
2.1.1 Informační technologie.....	5
2.1.2 Řízení informatiky.....	5
2.1.3 IT Služba.....	6
2.1.4 Proces.....	6
2.1.5 Funkce.....	6
2.1.6 Role.....	7
2.2 ITIL.....	7
2.2.1 Nejlepší praktiky.....	7
2.2.2 Charakteristické znaky ITIL.....	8
2.2.3 Životní cyklus služeb.....	9
2.2.4 Knihy ITIL.....	10
2.3 Management změn.....	13
2.4 Proces zlepšení v 7 krocích.....	13
2.5 Lean Six Sigma.....	17
2.5.1 DMAIC.....	18
2.6 ITSM projekt.....	18
2.6.1 Rozsah ITSM.....	19
2.6.2 Jednotlivé kroky ITSM projektu.....	19
3 Analýza požadavků.....	20
3.1 Zadavatel.....	20
3.2 Motivace a způsob řešení.....	21
3.2.1 Podnět k projektu.....	21
3.2.2 Postup.....	21
3.2.3 Cíl, rozsah a výstupy projektu.....	21

3.2.4	Zainterесované strany	22
3.3	Zmapování současného stavu	22
3.3.1	Nástroj.....	24
3.3.2	Role v procesu	25
3.3.3	Podprocesy.....	26
3.3.4	Pojmy procesu	29
3.3.5	Management konfigurací	31
3.4	Hledání příčin a návrh řešení.....	31
3.4.1	Prevence neúspěšných změn.....	31
3.4.2	Úspora času.....	34
3.4.3	Další související problémy.....	38
3.4.4	Shrnutí a návrh experimentu.....	38
4	Specifikace požadavků a návrh.....	40
4.1	Neformální specifikace.....	40
4.2	Specifikace případů užití	40
4.2.1	Aktéři a role	42
4.2.2	Popis případů užití	43
4.3	Další požadavky a dodatky	49
4.4	Nefunkční požadavky	51
4.5	Akceptační kritéria.....	52
4.6	Návrh architektury	52
4.6.1	Architektura Model-View-Presenter.....	52
4.7	Návrh databáze	53
4.8	Návrh grafického rozhraní	55
5	Implementace	56
5.1	Použité programovací jazyky	56
5.1.1	Nette Framework	57
5.1.2	Nette\Database	57
5.1.3	jQuery	58
5.2	Použité nástroje.....	58
5.3	Implementace s Nette Framework	58
5.4	Implementace vybraných požadavků.....	59
5.4.1	Vyhledávání změn	59
5.4.2	Úpravy změny.....	59
6	Experiment a výsledky	60
7	Závěr	62

1 Úvod

Informační technologie v dnešní době hrají velkou roli v podpoře podnikových aktivit mnoha firem. Konkurenční výhoda těchto firem je často závislá právě na kvalitě IT služeb.

Garance kvality IT služeb, jejich zabezpečení včetně kvalitní a efektivní podpory vyžaduje řízení IT služeb neboli tzv. IT Service Management (ITSM).

Následnou potřebu ověřených praktik v oblasti ITSM i jejich neustálý rozvoj pomáhá řešit například IT Infrastructure Library (ITIL). ITIL je mezinárodně uznávaný souhrn doporučených praktik postavený na dlouholetém sbírání zkušeností z různých společností o různé velikosti, v různých zemích a s různými IT systémy.

Management změn, kterým se budu blíže zabývat v této práci, je jednou z podstatných částí řízení IT služeb. Její hlavní podstatou je minimalizace rizika a dopadu prováděné změny na poskytované služby.

Samotná realizace řízení IT služeb často vyžaduje technickou podporu, aby mohla splnit rostoucí nároky na efektivitu a kvalitu.

1.1 Cíle práce

Cílem práce je poskytnout náhled do inovace části ITSM v reálné společnosti, konkrétně managementu změn. Dílčími cíli jsou analýza současného stavu a analýza potřebných změn, výběr části změn a jejich přínosu pomocí implementace takto pozměněné technické podpory managementu změn. Tato práce chce také poukázat na provázanost procesů, nástrojů technické podpory a lidí v ITSM projektech.

1.2 Rozsah práce

Obsahem této práce je analýza a specifikace požadavků na implementaci technické podpory managementu změn v kontextu zavedených procesů reálné společnosti. Tato analýza vychází ze současné situace ve společnosti – současných procesů, současných nástrojů technické podpory a potřeby změn u těchto nástrojů. Na základě těchto požadavků je navrženo řešení a to je následně implementováno. Tato implementace má za cíl ověření očekávaného dopadu na procesy a potvrzení přínosu těchto změn. Tyto výsledky pak budou použity k opodstatnění těchto změn v produkčním nástroji.

1.3 Terminologie

Problémem terminologie v oblasti IT služeb byla a stále částečně je častá nejednotnost významů jednotlivých termínů a fakt, že většina literatury zabývající se IT službami je v anglickém jazyce. Bez přesných definic termínů by mohlo být mnoho míst této práce zavádějících a použití českých výrazů

bez zmínky o jejich anglickém ekvivalentu by ztížilo logické navázání na publikace v anglickém jazyce.

A právě jeden z přínosů ITIL je jednoznačná terminologie v oblasti řízení IT služeb. Pro tento katalog pojmů existuje i oficiální česká variace, která obsahuje anglický a český ekvivalent a popis v obou jazycích. Část tohoto katalogu zahrnující pojmy použité v této práci naleznete jako přílohu 1. Další pojmy budou vysvětleny v samotném textu práce.

Mnohé pojmy jsou i u nás známější pod anglickým pojmem než-li pod jeho českým ekvivalentem. A zadávající společnost požadovala aplikaci a dokumentaci k ní v anglickém jazyce. Z těchto důvodů jsou v práci občas uvedeny anglické ekvivalenty v závorce za českým výrazem.

Anglické texty z citovaných zdrojů byly volně přeloženy autorem této práce.

1.4 Utajení některých údajů a fiktivní názvy

Vzhledem k požadavku firmy na utajení některých citlivých dat zde není uveden skutečný název firmy. V práci je místo skutečného jména firmy použito smyšlené jméno „TechServices“.

1.5 Rozložení tohoto dokumentu

První kapitola je úvodem.

Druhá kapitola je teoretickou částí práce. Tato kapitola popisuje teoretické podklady použité v této práci – standardy řízení IT služeb, management změn, popisuje také postupy používané při zlepšování a projektech v ITSM.

Třetí kapitola je první praktickou částí, jejím obsahem je celková analýza problému. Zahrnuje stanovení cíle, rozsahu a výstupů projektu, zmapování současného stavu – procesů, nástrojů technické podpory a praktik – ve firmě a následně hledá změny, které by naplnily stanovené cíle. Pokrývá tak sběr a analýzu požadavků na nástroj technické podpory managementu změn.

Čtvrtá kapitola je druhou praktickou částí a obsahuje specifikaci požadavků z předchozí kapitoly a návrh implementace nástroje.

Pátá kapitola popisuje několik bodů z implementace tohoto nástroje – použité nástroje, části aplikace.

Šestá kapitola popisuje experiment ověřující přínos navržených změn v nástroji a jeho výsledky.

Sedmá kapitola je závěrečným zhodnocením této práce.

1.6 Návaznost na jiné práce

Tato práce nenavazuje na semestrální projekt ani jinou práci.

2 Standardy řízení IT služeb

Management změn v rámci této práce je „jen“ jednou z částí řízení IT služeb (IT Service Management, zkratka ITSM). Tato kapitola popisuje základy ITSM, management změn a postupy a metody používané při zlepšování. Na začátku této kapitoly jsou popsány základní pojmy IT služeb, v další části je zběžně představen ITIL a jeho jednotlivé knihy, které zahrnují i management změn. V závěru kapitoly je představena problematika projektů v ITSM.

2.1 ITSM

V oblasti řízení podnikové informatiky nestačí již dělat „standardní správu informačních technologií“, je potřeba definovat a popsat služby, které podniková informatika poskytuje zaměstnancům, naučit zaměstnance s těmito službami pracovat v kontextu jejich každodenních činností a následně tyto služby kontinuálně řídit, a to jak na operativní, tak i taktické a strategické úrovni. IT Service Management (ITSM) je disciplína, která se této oblasti věnuje. ITSM se tedy zkráceně řečeno zabývá řízením IT služeb a infrastruktury organizace a jejich přínosu pro podnik.

V současnosti ITSM metody zahrnují specifické cesty k dosažení a optimalizaci analýzy, plánování a implementaci nejlepších praktik (best practices) popsaných v metodice ITIL (viz kapitola 2.2 ITIL) [1].

2.1.1 Informační technologie

Pojem informační technologie (zkratka IT) je dle ITIL verze 3 definován jako „technologie pro uložení, komunikaci a zpracování informací, jež typicky zahrnuje počítače, telekomunikace, aplikace a další software“ [2]. Pojem IT dle ITIL verze 3 tedy zahrnuje i komunikačních technologie. Mimo ITIL je někdy snaha zahrnutí komunikačních aspektů explicitně vyjádřit tak, že se mluví o informační a komunikační technologii (zkratka ICT) [3].

2.1.2 Řízení informatiky

Řízení informatiky je součástí celkového řízení podniku. Představuje všechny činnosti spojené s určováním základní koncepce, strategie informatiky, s plánováním a poskytováním informačních služeb nebo se zajišťováním potřebných zdrojů [4].

IT strategii je potřeba sladit s celkovou strategií a cíli podniku. Podniková informatika musí podle strategie podniku definovat svoje cíle a přínosy, které dosažením ICT cílů získá. Takto vytvořena informační a komunikační strategie je integrována do celkové strategie podniku. Řešení informatiky je v každém podniku unikátní a proto je potřeba hledat optimální řešení pro daný podnik v daných podmínkách [5].

Nároky na řízení informatiky podniku vyplývají z ekonomických, obchodních a provozních potřeb a cílů firmy a současně nových možností informačních a komunikačních technologií [4].

2.1.3 IT Služba

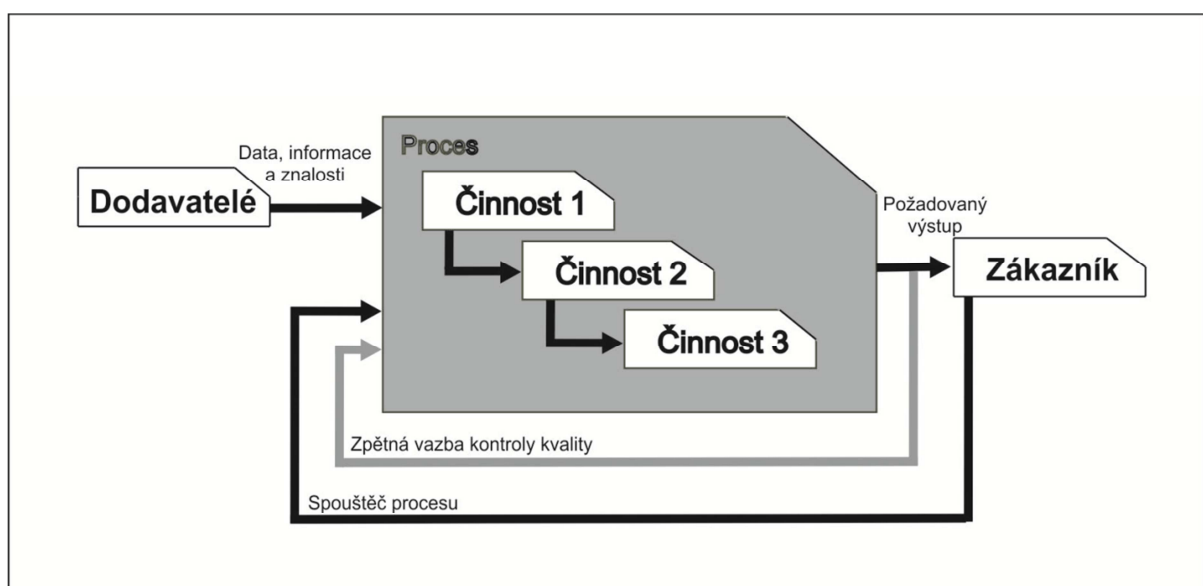
ITIL definuje službu IT jako speciální případ služby založené na použití informačních technologií [2]. Služba je dle ITIL prostředek poskytování hodnoty zákazníkovi, prostřednictvím výstupů, kterých chce zákazník dosáhnout, aniž by vlastnil specifické náklady a rizika spojená s dodávkou služby.

2.1.4 Proces

Proces je dle ITIL [2] strukturovaná množina činností navržená pro dosažení určitého specifického cíle. Proces má definované vstupy a přetváří je na definované výstupy. Proces může obsahovat role, odpovědnosti, nástroje a manažerské kontrolní mechanismy požadované pro spolehlivou dodávku výstupů.

Každý proces by měl být zdokumentovaný, měřitelný a kontrolován. Kontrolováný proces se stane opakovatelným a říditelným. Každý proces by měl mít svého vlastníka, který by měl být zodpovědný za proces a jeho zlepšování a za to, že proces bude plnit své cíle [6].

Na obrázku 2.1 je zobrazen jednoduchý proces zachycující základní souvislosti.



Obrázek 2.1: Jednoduchý proces, vstupy, výstupy, zdroj [6]

2.1.5 Funkce

Funkce je tým nebo skupina lidí a jejich nástroje, které používají k provádění jednoho nebo více procesů nebo aktivit. Funkce je organizační jednotkou určenou k provádění určité práce a zodpovědnou za určité výstupy. Je vybavena schopnostmi a zdroji potřebnými k jejich výkonu a výstupům [6].

2.1.6 Role

Role je sada odpovědností, aktivit a pravomocí vztažených k jednotlivci nebo týmu. Jeden tým nebo jednotlivec může mít i více než jednu roli. Rozsah rolí je definován v příslušném procesu [ITIL v3].

2.2 ITIL

„ITIL je veřejně dostupný rámec, jenž popisuje nejlepší praktiky ve správě IT služeb. Poskytuje rámec pro zvládnutí IT v organizaci, pojednává komplexně o službách a zaměřuje se na neustálé měření a zlepšování kvality dodávaných služeb IT, a to jak z pohledu businessu, tak z pohledu zákazníka.“ ITIL může pomoci všem IT organizacím bez ohledu na velikost nebo používané technologie a není ani byrokratický ani neohrabaný, pokud je použit v souladu s obchodními požadavky organizace [6].

Některými z přínosů implementace metodiky ITIL [6] jsou:

- Zvýšení uživatelské a zákaznické spokojenosti s IT.
- Zvýšení dostupnosti služeb vedoucí k zvýšení tržeb a zisků.
- Snížení nákladů plynoucích z omezení opakované práce a lepšího přidělování zdrojů.
- Snížení času potřebného k uvedení nového výrobku nebo služby na trh.
- Zjednodušení rozhodování a optimalizace rizik.

ITSM Portál [3] upozorňuje, že při práci s ITIL je třeba mít na paměti, že se jedná o sbírku nejlepších zkušeností (best practices) z oblasti řízení služeb IT. Nejedná se však ani o metodiku, ani o normu.

ITIL je rámec ITSM, nikoli metodika. ITIL nepopisuje jednotlivé prvky do detailu, ale dává rámcové návody, nepopisuje jak řešit podporu služeb ve specifických podmínkách. ITIL proto zmiňuje princip „*přijmi a přizpůsob*“ (adopt and adapt).

Podle ITIL nelze systém řízení služeb IT objektivně auditovat. ITIL doporučuje, nepředpisuje.

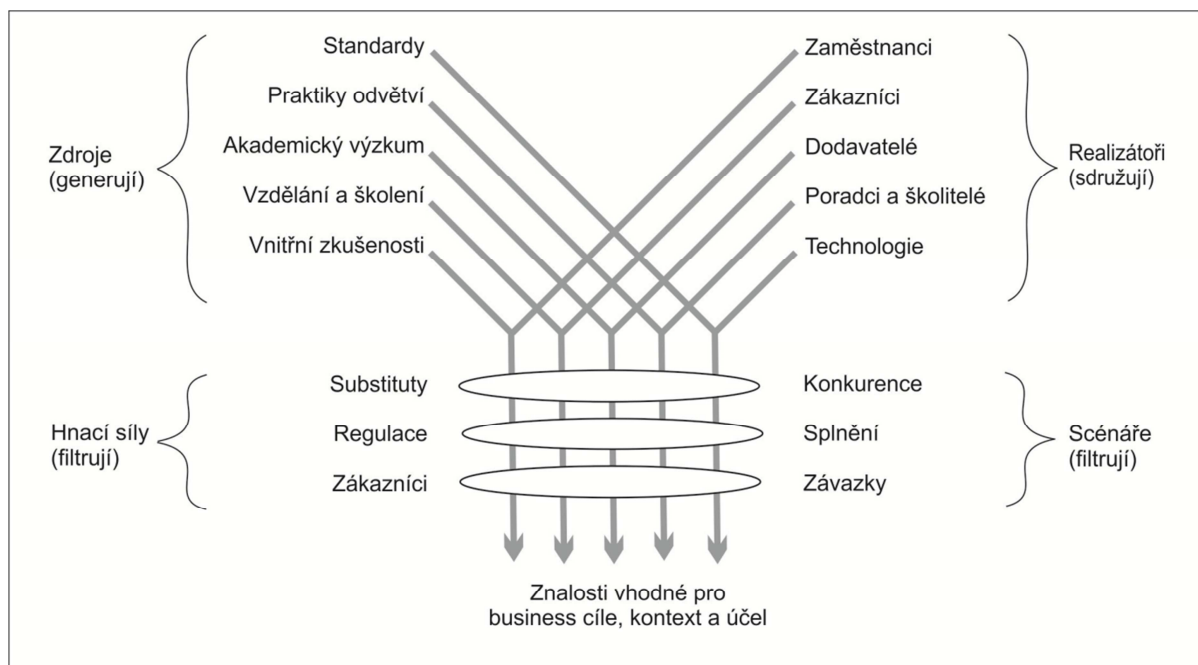
2.2.1 Nejlepší praktiky

Nejlepší praktiky jsou dle metodiky ITIL [6] sadou technik, metod, procesů, postupů a aktivit založených na úspěšném použití mnoha lidmi nebo organizacemi v dlouhém časovém období. Základní myšlenka je taková, že je časově i finančně výhodnější okopírovat to, co už někdo jiný vymyslel, než vymýšlet vlastní řešení pro stejný problém.

Tato přejatá řešení by mohla být často i kvalitnější, protože vyšší kvality u mnoha řešení lze dosáhnout až postupným vylepšováním. A nejlepší praktiky jsou právě výtvořem mnoha lidí nebo organizací v dlouhém časovém období.

Na druhou stranu zde musí zákonitě docházet ke zpoždění, protože i ta současná „nejlepší řešení“ byla do nejlepších praktik zařazena, až se osvědčila v praxi [3].

Vznik nejlepších praktik včetně zdrojů, vlivu realizátorů a filtrů zachycuje obrázek 2.2.



Obrázek 2.2: Vznik nejlepších praktik, zdroj [7]

2.2.2 Charakteristické znaky ITIL

Dle [8] jsou pro ITIL charakteristické tyto znaky:

- **Procesní řízení**

ITIL přináší moderní, procesně orientovaný přístup k řízení IT služeb (na rozdíl od tradičního funkčně-liniového řízení). Proces je logický sled činností transformujících nějaký vstup na nějaký výstup, přičemž plnění jednotlivých činností v procesu je zajišťováno rolími s jasně definovanými odpovědnostmi. Celý proces je řízen, monitorován, měřen, vyhodnocován a neustále vylepšován, což je odpovědností vlastníka procesu.

- **Zákaznický orientovaný přístup**

Tento rys vyplývá přímo ze samotné podstaty ITSM; všechny procesy se navrhují s ohledem na potřeby zákazníka, tzn. každá aktivita, každý úkon v každém procesu musí přinášet nějakou přidanou hodnotu pro zákazníka - pokud ne, pak je taková činnost nadbytečná.

- **Jednoznačná terminologie**

Jednoznačná terminologie je někdy málo doceňovanou nebo úplně opomíjenou charakteristikou ITIL, ale jen do té doby, než budeme poprvé v praxi řešit nedorozumění plynoucí z toho, že někdo používá stejný termín v jiném významu, než očekáváme.

- **Nezávislost na platformě**

Rámec ITSM procesů podle ITIL je nezávislý na jakékoliv platformě. Dokonce je možné ITIL použít i pro navržení procesů v jakékoliv firmě, která podniká ve službách i mimo IT.

- **Veřejně dostupná**

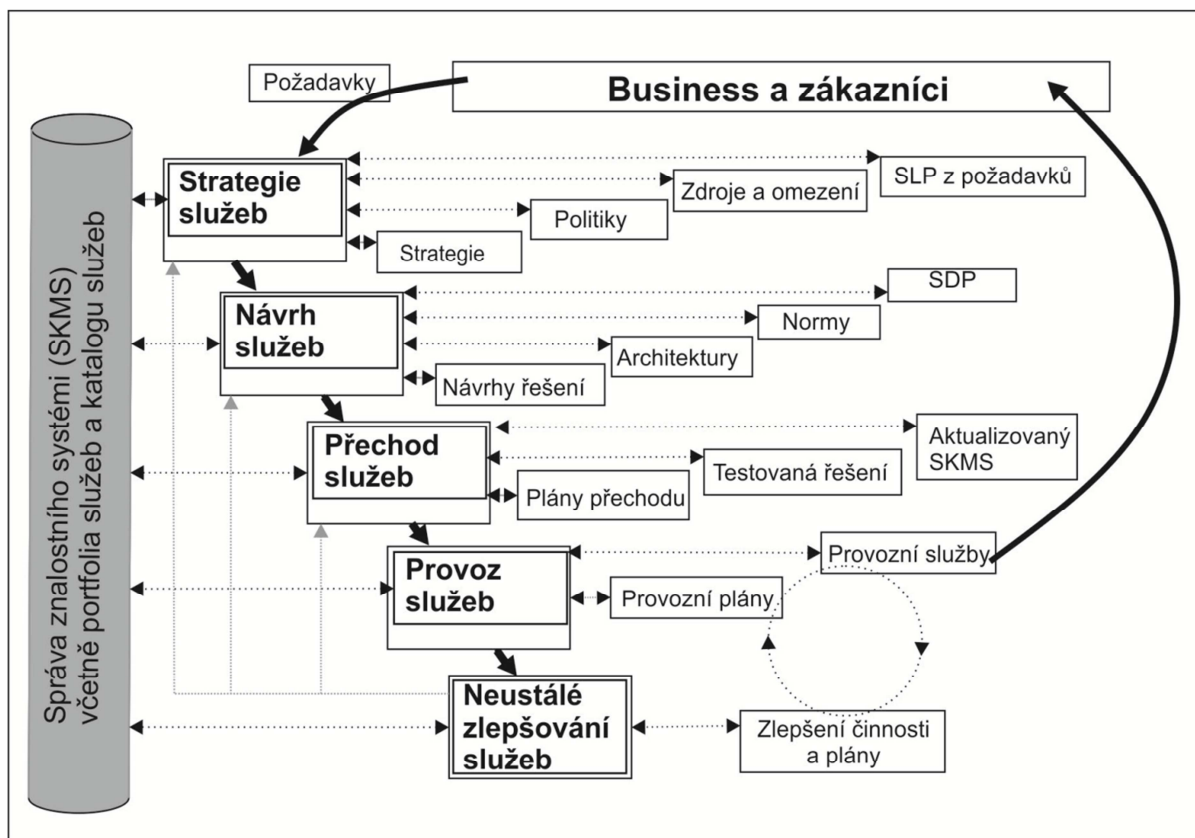
Knihovna je volně dostupná, což znamená, že každý si může knihy ITIL koupit a procesy ITSM podle ITIL ve svém podniku implementovat, aniž by musel platit jakékoliv další licenční poplatky. Tato skutečnost přispěla k rychlému celosvětovému rozšíření ITIL.

2.2.3 Životní cyklus služeb

Podle ITIL vychází podoba každé služby ze strategie (Service Strategy), která definuje důvody její existence, následně služba prochází tvorbou návrhu (Service Design), jenž je následně zrealizován, služba je nasazena do provozu (Service Transition) a na každodenní bázi provozována (Service Operation). Ve všech fázích životního cyklu služby pak dochází k neustálému zlepšování (Continual Service Improvement) všech aspektů služby.

V ITIL zcela úmyslně schází fáze vývoje služby (Service Development), protože na životní cyklus služby je zde nahlíženo z pohledu zákazníka [3].

Podrobnější životní cyklus služeb včetně vstupů a výstupů je uveden na obrázku 2.3.



Obrázek 2.3: Podstatné souvislosti, vstupy a výstupy fází životního cyklu služeb, zdroj: [9]

vysvětlivky k obrázku: SLP = Service Level Package (Sada úrovní služby), SDP = Service Design Package (Sada pro návrh služby), SKMS = Service Knowledge Management System (Systém správy znalostí o službě)

2.2.4 Knihy ITIL

Rámec ITIL je vzhledem ke své obsáhlosti rozdělen do několika knih.

Strategie služeb (Service Strategy)

Kniha Service Strategy ukazuje, jak přetvořit správu služeb v strategické aktivum nezbytné k dosažení strategických cílů. Pomáhá porozumět hodnotě služeb a objasňuje vztahy mezi službami, systémy a procesy, které řídí organizaci, a jejími business modely, strategiemi a cíli.

Návrh služeb (Service Design)

Kniha Service Design se zabývá návrhem a vývojem služeb a procesů správy služeb. Obsahuje principy návrhu a metody pro přetváření strategických cílů v portfolio služeb. Pod návrh služeb nespadá pouze tvorba nových služeb, patří sem také změny a vylepšení služeb potřebných ke zvýšení nebo udržení hodnoty pro zákazníky. Kniha se také zabývá kontinuitou služeb, dosahováním dohodnutých úrovní služeb a shodou se standardy a předpisy.

Klíčové procesy a činnosti knihy Service Design jsou správa katalogu služeb (Service Catalogue Management), správa úrovně služeb (Service Level Management), správa kapacit

(Capacity Management), správa dostupnosti (Availability Management), správa kontinuity služeb (IT Service Continuity Management), správa bezpečnosti informací (Information Security Management) a správa dodavatelů (Supplier Management).

Přechod služeb (Service Transition)

Kniha Service Transition poskytuje průvodce vývojem a zlepšováním služeb před jejich předáním do provozu. Děje se tak po příjmu balíčku návrhu služby z fáze návrhu služby. Tato kniha se nezabývá jen implementací služeb, ale i všech ostatních aspektů služby. Je třeba zajistit, aby služba pracovala

i v určitých extrémních nebo abnormálních podmínkách (v rámci mezí) a aby v případě poruch nebo chyb byla dostupná podpora, chyby byly odstraněny a byl minimalizován jejich dopad na business. Pokud se po fázi návrhu služby změni požadavky businessu na službu, pak jsou tyto změny provedeny ve fázi přechodu služeb.

Kniha Service Transition uvádí 7 procesů: správa znalostí (Knowledge Management), plánování a podpora přechodu (Transition Planning and Support), management změn (Change Management), správa aktiv a konfigurace (Service Asset and Configuration Management), správa verzí a nasazení (Release and Deployment Management), ověření a testování služby (Service Validation and Testing) a vyhodnocení (Evaluation). Z těchto sedmi se blíže podíváme na management změn v kapitole 2.3.

Provoz služeb (Service Operation)

Účelem provozu služeb je koordinovat a provádět aktivity a procesy potřebné k dodání a řízení služeb zákazníkům na dohodnuté úrovni. Provoz služeb je také zodpovědný za technologie a infrastrukturu, které podporují dodávku těchto služeb. Pouze v této fázi životního cyklu služby dodávají viditelnou hodnotu businessu, a pokud nejsou správně prováděny, kontrolovány a řízeny, pak nezáleží, jak dobře byly navrhnuty a implementovány, hodnotu nepřinesou žádnou nebo jen malou.

Mezi klíčové procesy provozu služeb patří správa událostí (Event Management), správa incidentů (Incident Management), provádění požadavků (Request Fulfillment), správa přístupů (Access Management) a správa problémů (Problem Management). Aktivity při provozu služeb, které nejsou součástí výše popsaných procesů, jsou monitorování a kontrola (detekce stavu služeb a konfiguračních položek a vyvolání příslušné nápravné akce), centrum dohledů (ústřední koordinační bod pro monitorování a správu služeb) a správa infrastruktury (HW, SW, databází atd.).

Neustálé zlepšování služeb (Continual Service Improvement)

Kniha Continual Service Improvement (zkratka CSI) provádí tvorbou a udržováním hodnoty pro zákazníka skrze lepší návrh, zavedení a provoz služeb. Kombinuje principy, praktiky a metody

managementu kvality a managementu změn a používá je na jednotlivé etapy cyklu, aktuální služby, procesy, činnosti a technologie.

Klíčovým procesem CSI je Proces zlepšení v 7 krocích (7-Step Improvement process). Tento proces je blíže popsán v kapitole 2.4. Cyklus neustálého zlepšování je obsažen i v principu Plan-Do-Check-Act podle CSI, ten je uveden níže. Pro CSI je také podstatné měření služeb, které mu poskytují důležité informace o stavu procesu. Níže jsou tyto modely a techniky blíže popsány.

PDCA model

Plan-Do-Check-Act model je základním principem CSI. Jeho cílem je neustálé zlepšování.

Ve fázi „Plánuj“ jsou stanoveny cíle a související metriky. Je provedena analýza současného stavu a porovnána se stavem, kde bychom chtěli být. Jsou definovány kroky k překlenutí tohoto rozdílu.

Fáze „Udělej“ zahrnuje vývoj a implementaci projektu k překlenutí rozdílu mezi současným a požadovaným stavem. Jsou implementována vylepšení procesů managementu služeb.

Ve fázi „Zkontroluj“ jsou porovnána implementovaná vylepšení s metrikami z fáze „Plánuj“. Zda se podařilo dosáhnout požadovaného stavu, záleží na stanovených tolerancích.

Fáze „Uskutečni“ vyžaduje finální vylepšení stávajících procesů. Pokud nebyl zcela dosažen požadovaný stav, je třeba zajistit zdroje pro další kolo PDCA cyklu. Rozhodnutí z této fáze jsou vstupem pro další kolo.

Měření služby (Service Measurement) se používá k měření a reportování aktuálního výkonu služby oproti stanoveným cílům. K měření se používá různých metrik.

Z procesních metrik to jsou kritické faktory úspěchu (Critical Success Factor, zkratka CSF), klíčové ukazatele výkonnosti (Key Performance Indicator, zkratka KPI) a metriky aktivit. Technologické metriky jsou používány například k měření výkonnosti nebo dostupnosti aplikací nebo jejich komponent. Metriky služby měří výsledky služby z pohledu koncového uživatele. Procesní metriky, ve formě kritických faktorů úspěchu a klíčových ukazatelů výkonnosti, určují celkové zdraví procesu. Klíčový ukazatel výkonnosti je metrika používaná k měření úspěšnosti dosažení kritických faktorů úspěchu. Klíčový ukazatel výkonnosti může pomoci při hledání odpovědi na otázky ohledně kvality, výkonu, hodnoty a souladu procesu.

Technické pozorování (Technical Observation) je technika užívaná ke zlepšení služby, zkoumání problémů a správu dostupnosti. Personál technické podpory se schází a monitoruje chování a výkonnost služby IT a následně vytváří technická doporučení ke zlepšení

Analýza trendů (Trend Analysis) je analýzou dat za účelem identifikace změn chování v čase.

2.3 Management změn

Management změn zajišťuje, aby všechny změny byly zaznamenány, vyhodnoceny, autorizovány, byla jim určena priorita, byly testovány, implementovány, dokumentovány a zkontrolovány. Všechny změny by měly být zaznamenány v systému správy konfigurací (Configuration Management System). To zajistí snížení počtu výskytu chyb u nových nebo měněných služeb a rychlejší a přesnější implementaci změn.

„Změna služby je přidání, modifikace nebo odstranění autorizované, plánované nebo podporované služby nebo její komponenty a dokumentace, která k ní přísluší.“ [9] Požadavky na změnu mohou mít formu oficiálních žádostí o změnu, dokumentu o zahájení projektu atd.

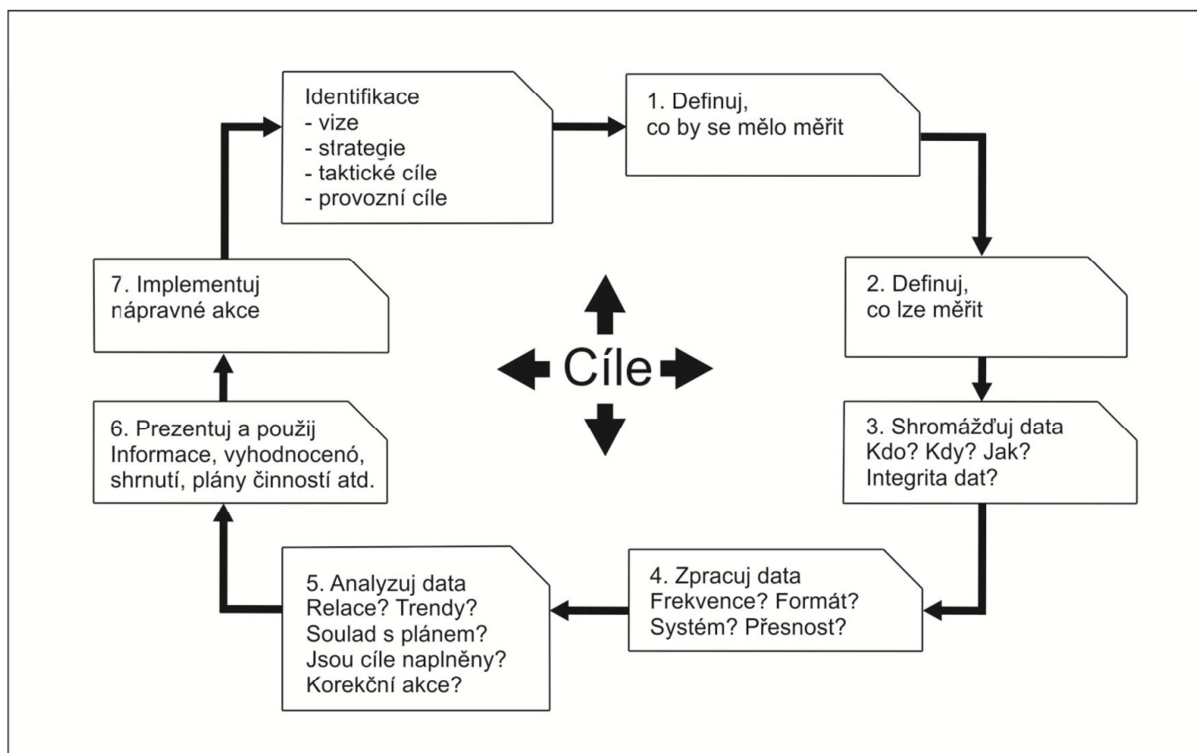
Systém správy konfigurací

Systém správy konfigurací souvisí s managementem aktiv a konfigurací (Service Asset and Configuration Management, zkratka SACM). SACM má za úkol identifikovat, kontrolovat a zodpovídat za aktiva služeb a konfigurační položky, chránit a zajišťovat jejich integritu v celém životním cyklu služby. To zajišťuje tak, že dovoluje užít jen povolené komponenty a dělat jen povolené změny. Cílem je definovat a kontrolovat komponenty služeb a infrastruktury a udržovat přesné informace o konfiguraci služeb a infrastruktury na historické, plánované a aktuální bázi. To dovoluje podporovat ostatní procesy Správy Služeb dodáváním přesných informací o konfiguraci.

Všechny potřebné údaje jsou zaznamenány právě v systému správy konfigurací. Tyto údaje pak pomáhají lidem například autorizovat změny nebo řešit incidenty rychleji, snižují počet problémů a reklamací způsobených nevhodnou konfigurací atd. Pokud konfigurační položkou bude například aplikační software, pak jeho záznam bude obsahovat dodavatele SW, jeho cenu, datum nákupu, datum vypršení licence, smlouvu, případně SLA atd.

2.4 Proces zlepšení v 7 krocích

Proces zlepšení v 7 krocích (7-Step Improvement process) je procesem CSI. Kroky procesu prováděné v cyklu (viz obrázek 2.4) jsou základem k neustálému zlepšování. Tento proces zahrnuje kroky pro sběr smysluplných dat, jejich analýzu (identifikaci trendů a problémů), prezentaci takto získaných informací managementu za účelem určení priorit a schválení implementace nápravných změn. Pro podporu zlepšovacího procesu je důležité tento proces integrovat do každého stavu životního cyklu služeb včetně podpůrných procesů [7].



Obrázek 2.4: 7-Step Improvement Process, zdroj [7]

První dva kroky procesu jsou spojené se strategickými, taktickými i operativními cíli v rámci služeb a procesů řízení služeb, také u technologií i CSI aktivit. Z těch se vychází v tom, co by se mělo měřit. Častou chybou je, že se tyto první kroky ignorují a informace se začínají sbírat, aniž bychom věděli, k čemu tyto informace budou a jak budou použity [7].

1. Definuj, co by se mělo měřit – zde se vychází z otázky „co je podstatné pro business?“ Odpovědi lze najít u zákazníka, business managementu, IT managementu. Vhodným zdrojem může být katalog služeb.

Hledáme a spojujeme tyto aspekty:

- Firemní vize, mise a cíle.
- IT vize, mise a cíle.
- Kritické faktory úspěchu.
- Požadované hodnoty poskytovaných služeb.
- Popis práce IT oddělení.

2. Definuj, co lze měřit – ne všechno, co by bylo vhodné měřit, také měřit lze. Některá data lze získat z aktuálních nástrojů či reportů, některá budou vyžadovat nové nástroje či nové reporty.

Vhodné je začít seznamem aktuálních nástrojů, které jsou zdrojem dat a zjistit, co každý nástroj měří. Dále také prohlédnout procesy, procedury a pracovní instrukce a zjistit, která data se reportují.

Mezi potřebami a aktuálními možnostmi lze provést rozdílovou analýzu (gap analysis) a tak určit požadavky pro nové nástroje, konfigurace, reporty. S těmito požadavky je obvykle nutné jít za business managementem firmy, IT managementem a zákazníkem.

3. Shromážďuj (sbírej) data – pro sběr dat je potřeba určit:

- Kdo je odpovědný za sběr.
- Jak budou data sbírána.
- Kdy a jak často.
- Jaká budou kritéria hodnotící integritu dat.

Kromě samotných průběžných čistých hodnot parametrů služeb, procesů, atd. nás zajímají i výjimečné stavy těchto hodnot. Zachycení těchto stavů se dle ITIL nazývá „monitorování“. Pro výjimečné stavy je obvykle potřeba hledat řešení, dlouhodobé přesahy volají po rozsáhlejší změně.

Můžeme rozeznávat tři typy metrik:

- Technologické - spojené s komponentami, aplikacemi (např. dostupnost, výkon).
- Metriky procesu - jsou ve formě CSF (kritického faktoru úspěchu) nebo KPI (klíčového indikátoru výkonnosti) a metrik činností, určují zdraví procesu a signalizují místa pro zlepšení.
- Metriky služby - zahrnují službu v celém jejím průběhu (end-to-end), jsou souhrnem několika ostatních metrik.

ITIL zmiňuje tuto proceduru zavedení sběru dat:

1. Určí požadavky na sběr dat a monitorování.
2. Určí četnost, s kterou mají být sbírány.
3. Určit požadavky na nástroje.
4. Vytvoř procedury na sběr a monitorování.
5. Vytvoř a komunikuj plán sběru a monitorování.
6. Aktualizuj plány dostupnosti a kapacity.
7. Začni samotný sběr dat.

4. Zpracuj data – získaná data je potřeba konvertovat do požadovaného formátu pro danou cílovou skupinu. Snažíme se data logicky spojovat. V úvahu při zpracování je nutné brát:

1. Frekvenci zpracování dat.
2. Požadovaný formát výstupu (jak je analýza provedena, jak jsou informace použity).
3. Jaké nástroje můžeme použít pro zpracování dat.
4. Jak ověříme přesnost dat.

5. Analyzuj data – bez analýzy jsou metriky zbytečné. Chybou by bylo považovat výstup sběru dat v podobě grafu za konečný výsledek. Říká se, že „obrázek vydá za tisíce slov“, ale v analýze je právě otázkou: „Jakých tisíc slov?“ Data byla zpracována v předchozím kroku do informací. Z těchto informací můžeme získávat znalosti o událostech, které mají vliv na organizaci.

Informace můžeme analyzovat hledáním odpovědí na tyto otázky:

- Jsou v nich nějaké jasné trendy?
- Jsou tyto trendy negativní nebo pozitivní?
- Co způsobuje tyto trendy?
- Jsou tyto trendy očekávané?
- Je tento průběh/trend podle plánu?
- Jsou potřebné nějaké změny?
- Dosahujeme požadovaných hodnot?
- Je potřeba nápravných akcí?
- Je za tím nějaký problém (příčina)?
- Jaká je cena „zalepení mezery“?

Výsledky porovnáváme s měřitelnými cíli. Pokud jich nebylo dosaženo, měly bychom vznést potřebu změny. Při provádění změny je potřeba dále sbírat data, která budou sloužit jako ukazatel, že bylo dosaženo očekávaného zlepšení.

6. Prezentuj a použij znalosti – získané výstupy z analýzy je potřeba dále prezentovat těm, kteří je využijí. Problémem ale může být rozdíl mezi potřebami cílové skupiny a tím, co prezentujeme. Report proto musíme přizpůsobit cílové skupině a jejím potřebám. Lze uvažovat tyto základní cílové skupiny:

- Business management - nejvíce zajímá, zda IT dodalo služby, které slíbilo a pokud ne, jakou korekci plánuje.
- Senior IT – jsou obvykle zaměřeni na CSF a KPI, spokojenost zákazníka, aktuální stav versus plán, cenu a návratnost.
- Interní IT – nejčastěji sledují KPI a měřítka činností, ty jim pomáhají plánovat, koordinovat a hledat příležitosti na (přírůstková) zlepšení.

7. Implementuj nápravné akce – použij získané znalosti na optimalizaci, vylepšení, korekci služby.

2.5 Lean Six Sigma

Lean Six Sigma také souvisí s vylepšováním. I když byla původně „jen“ spojením metodologií Lean a Six Sigma, je v současnosti již spíše komplexním systémem řízení. Přesto jsou právě tyto dvě metodologie s jejich rozdílnými záměry i cestou základem tohoto řízení, které společnou synergií dokážou naplnit potřeby mnoha podniků. [10]

Obě metody nejlépe vystihuje tabulka 2.1.

Tabulka 2.1: Hlavní znaky a porovnání Lean a Six Sigma, zdroj [10]

	Lean	Six Sigma
Záměr	Efektivní vytvoření hodnoty, která je definována na základě znalosti požadavku zákazníka	Efektivní zajištění kvality, která je vymezena kritickými vlastnostmi předmětu podle definice zákazníka.
Cesta	Odstranění plýtvání.	Snížení variability.
Nejvýraznější přínos	Zkrácení doby trvání procesu.	Zvýšená uniformita výstupů procesu.
Organizace cyklu projektu	Cyklický a iterativní PDCA, Naplánuj-Udělej-Zkontroluj-Zasáhni.	DMAIC (Definuj-Měř-Analyzuj-Zlepši-Kontroluj).
Klíčové metody	Mapování a měření procesních toků. Optimalizace procesních toků.	Měření výskytů a četností. Analýzy příčin a důsledků.

2.5.1 DMAIC

Model DMAIC z anglického „Define-Measure-Analyze-Improve-Control“, česky „Definuj-Měř-Analyzuj-Zlepši-Kontroluj“, je modelem zlepšování metodologie Six Sigma. Tento model zlepšování zahrnuje těchto 5 fází [10], [11].

Definuj (Define)

- Určení cíle, rozsah projektu, vymezení výstupů
- Určit, co je potřeba zlepšit, identifikovat problémy
- Určit, co se bude sledovat, kdy a jak měřit
- Při všech bodech se řídit potřebami zákazníka (naslouchat hlasu zákazníka)
- Využívají se Paretovy analýzy, různé diagramy a jednoduché statistické nástroje.

Měř (Measure)

- Zmapovat proces
- Sbírat data

Analyzuj (Analyze)

- Identifikuj příčiny problému, potvrď jejich přítomnost
- Testuj hypotézy, regresní analýza, navrhni experiment
- Formuluj hypotézy, interpretuj výsledky
- Využívají se Ishikawův diagram, metody testování statistických hypotéz, DOE (návrh a vyhodnocení experimentů) a metody regresní analýzy

Zlepši (Improve)

- Navrhni, ověř a implementuj řešení

Kontroluj (Control)

- Zhodnoť výsledky předcházejících fází
- Zdokumentuj metody zlepšení
- Sleduj proces a vyhodnoť výsledky
- Využívají se regulační diagramy

2.6 ITSM projekt

ITSM projekt definujeme jako projekt, jehož účelem je dosáhnout konkrétního zlepšení buď v oblasti řízení služeb nebo ve správě infrastruktury IT a/nebo zdrojů IT. V ITSM projektu je potřeba správně rozlišovat cíle a prostředky k jejich dosažení.

Cílem by mělo být zlepšení nějakého aspektu IT prostředí, například zrychlit realizace změn, snížit průměrný počet výpadků. Definicí cílů by neměly být například funkční a ne-funkční požadavky na technickou podporu, ale parametry budoucího stavu.

Prostředkem může být automatizace procesních činností prostřednictvím technického nástroje (podpory), implementace nového procesu včetně změny nástroje technické podpory, zdokumentování pracovního postupu a nového proškolení lidí a podobně [3].

2.6.1 Rozsah ITSM

ITSM jako manažerská disciplína zahrnuje tři navzájem propojené a na sobě závislé oblasti. Jsou jimi procesy, nástroje a lidé [3].

Lidé – všichni zaměstnanci podniku, kteří přicházejí do kontaktu se službami IT.

Nástroje – nástroje usnadňující řízení služeb a infrastruktury IT a nástroje automatizující rutinní činnosti a spolupráci.

Procesy – veškeré organizačně-procesní prvky systému řízení služeb IT, zejména definice pojmů, vymezení aktivit, rolí a jejich odpovědností, definice vstupů a výstupů aktivit a procesů, definice komunikačních kanálů, metrik, reporting a dokumentace celého systému.

Pokud je jedna z těchto oblastí v projektu opomenuta, zvyšuje se riziko, že nebude dosaženo cílů tohoto projektu.

2.6.2 Jednotlivé kroky ITSM projektu

ITSM portál [3] zmiňuje čtyři důležité kroky, které by měly být součástí ITSM projektů.

1. Definice cíle a prostředků k jeho dosažení, přičemž je potřeba se vyvarovat jejich vzájemné záměny.
2. Definovat výstupy, které má projekt dodat, přičemž by měly zahrnovat všechny tři aspekty ITSM, tj. lidi, nástroje i procesy.
3. Naplánovat činnosti a zdroje, přičemž by mělo být zřejmé, které úkoly se týkají kterého aspektu.
4. Samotná realizace.

Projekt a jeho cíl vzniká z předchozích analýz, auditů, strategických plánování, požadavků businessu a manažerských rozhodnutí. Předpokladem mnoha z těchto prvků je sběr metrik. Ty slouží často také pro určení měřitelného cíle projektu a tak i k ověření jeho dosažení. Procesy sběru metrik je součástí knihy ITIL Continual Service Improvement, která byla popsána v předchozí podkapitole.

3 Analýza požadavků

Cílem první kapitoly praktické části práce je analýza problému a návrh řešení, z kterých vychází požadavky na nástroj technické podpory managementu změn. Kapitola zachycuje část sběru a analýzy požadavků.

V první části kapitoly popisují zhruba firmu, v které je tento projekt realizován a snažím se blíže odpovědět na otázku – co je náplní této práce.

V další části kapitoly je zmapován současný stav ve firmě, je popsán proces a to, jak jsou aktuálně automatizovány jeho části.

Poslední část popisuje problém a hledání příčin a jeho řešení – navržení změn, které povedou ke zlepšení v požadovaném směru a jsou v souladu s cíli zadavatele.

Požadavky na technickou podporu zdokumentované v následující kapitole pak vycházejí ze spojení aktuálního stavu a navržených změn.

3.1 Zadavatel

Na úvod je vhodné se zhruba seznámit se firmou a oblastí, které se projekt týká. Firma TechServices působí mezinárodně v outsourcingu IT služeb. Firma TechServices má více poboček po celém světě a větší počet zákazníků. Mnohá řešení byla tvořena na míru zákazníkovi.

Před několika lety firma TechServices jako svou strategii zvolila cestu k více sdíleným řešením a prostředím pro více svých zákazníků, ke standardizaci všech svých procesů a nástrojů. A to tak, aby mohla lépe poskytovat stálou kvalitu služeb zákazníkovi, splnit jeho potřeby a uspokojit i své investory.

Firma je procesně orientovaná. Ve svých procesech vychází zvláště z ITUP¹, který vychází z ITIL v3 a COBIT². Firma má také zavedené procesy pro neustále zlepšování, proto má k dispozici již delší historii metrik v oblasti managementu služeb i výsledků jejich analýz. V rámci svých metod v inovacích vychází také z metodologií Lean Six Sigma³. V projektech firma interně doporučuje využívat metod a praktik dle PMBOK Guide⁴.

Tým managementu změn, technické týmy ve firmě a zákazník v minulosti využívali více různých nástrojů, které byly dvěma lety v rámci strategie standardizace nahrazeny jedním strategickým nástrojem. Tento nástroj je vyvíjen partnerskou firmou.

¹ IBM® Tivoli® Unified Process (ITUP) poskytuje detailní dokumentaci procesů používaných v managementu IT služeb. Je produktem společnosti International Business Machines Corporation (IBM Corp.).

² COBIT je framework vytvořený mezinárodní asociací ISACA pro správu a řízení informatiky (IT Governance).

³ Lean Six Sigma je komplexním systémem řízení spojující metodologie Lean a Six Sigma, více v kapitole 2.6.

⁴ PMBOK Guide (A Guide to the Project Management Body of Knowledge) je metodika a příručka pro projektové řízení vyvíjena neziskovou organizací zaměřující se na projektové řízení PMI (Project Management Institute).

3.2 Motivace a způsob řešení

3.2.1 Podnět k projektu

Podnět k tomuto projektu vychází ze dvou souvisejících zdrojů.

První podnět vychází z neustálého zlepšování služeb a souvisejících procesů. Tento podnět se zvláště týká rozhodnutí vedení o zvýšení tlaku a podpory svých aktivit ke snížení počtu neúspěšných změn na minimum. Neúspěšné změny ohrožují služby a tím se výrazně dotýkají zákazníka a cílů firmy.

Druhým jsou výtky uživatelů na práci s aktuálním nástrojem technické podpory managementu změn.

Tyto dva podněty byly spojeny do jednoho projektu.

3.2.2 Postup

Při plánování postupu jsem vycházel z doporučeného obecného postupu pro ITSM projekty (viz kapitola 2.6) a modelu DMAIC (viz kapitola 2.7) a adaptoval jej na potřeby tohoto projektu.

V prvním kroku (Define) jsem společně se zainteresovanými stranami (tým managementu změn, osoby odpovědné za inovace) určil cíle, rozsah a výstupy projektu.

V druhém kroku (Measure) jsem popsalsoučasný stav ve firmě – zmapoval proces, organizační strukturu ve vztahu k tomuto procesu a nástroj technické podpory, dále jsem se seznámil s metrikami, které jsou sbírány s připomínkami uživatelů.

Ve třetím kroku (Analyze) jsem identifikoval možné příčiny problému.

Ve čtvrtém kroku (Improve) jsem navrhl řešení – změny, kde hlavním předmětem změny byl nástroj technické podpory. Pro ověření očekávaného dopadu jsem navrhl ověření experimentem.

Dalším krokem bylo provedení experimentu a zhodnocení výsledků.

3.2.3 Cíl, rozsah a výstupy projektu

Při stanovení cílů, rozsahu i výstupů jsem se držel doporučení ohledně ITSM projektů (viz kapitola 2.5) a to v rozlišování cíle a prostředku k jeho dosažení a toho, aby výstupy i rozsah projektu zahrnovaly všechny aspekty ITSM – lidi, proces a nástroje.

Cílem tohoto projektu je přinést „úsporu času“ – zkrácení doby trvání některých činností v procesu managementu změn. Podmínkou je, aby nebyl negativně dotčen výstup. Hlavním nástrojem k dosažení tohoto cíle mají být změny v nástroji technické podpory. Změny v procesu mají být minimální. Tento cíl vychází z podnětu výtky uživatelů na práci s aktuálním nástrojem.

Druhým cílem je nalezení zlepšení, které by mohlo vést ke snížení počtu neúspěšných změn.

Výstupem projektu bude návrh na změny vedoucí ke zlepšení v rámci stanovených cílů. Výstupem tohoto projektu konkrétněji bude identifikování míst pro zlepšení ve stávajícím nástroji

formulovaných jako požadavky na technickou podporu. Součástí výstupu bude i popis nutných změn v procesu a personálu. Pokud to bude možné, měl by být přínos těchto změn podpořen výsledky experimentu provedeného cestou zjednodušené implementace technické podpory dle těchto požadavků.

Potřeba a způsob tohoto experimentu je daná postojem zainteresovaných stran. Aktuální nástroj je strategickým nástrojem firmy, který bez silného business důvodu nebude nahrazen, ale v případě potřeby může být rozvíjen. Odůvodnění změn je potřeba podložit. S vykázaným přínosem, který je součástí výstupu, budou tyto požadavky na změny vzneseny k odpovědnému vedení a ty pak případně k dodavateli tohoto strategického nástroje.

Ve firmě lze často identifikovat mnoho míst pro zlepšení, ale firma si nemůže dovolit je implementovat všechny. U každého zlepšení je proto potřeba zvážit jeho přínos, cenu a na základě aktuální situace i prioritu [7].

3.2.4 Zainteresované strany

Hlavními účastníky procesu a uživateli nástroje jsou týmy technické podpory, uživatelé zastupující zákazníka, tým managementu změn a týmy managementu kvality výstupu i procesu. Existuje velké množství dalších uživatelů, výše zmíněné skupiny jsou však nejčastějšími uživateli nástroje technické podpory managementu změn. Jejich spolupráce je potřebná k hledání míst ke zlepšení, požadavků i výsledků testování.

Každý podstatný krok v projektu (požadavky, návrh) by měl být projednáván s vlastníkem procesu managementu změn, kterým je právě tým managementu změn.

Tým, který aktuálně provozuje a poskytuje služby nástroje technické podpory, je důležitý pro získání přístupu pro fázi testování na reálných datech v praxi a také jako poskytovatel zdroje metrik k managementu změn.

3.3 Zmapování současného stavu

Pro zmapování procesu managementu změn ve firmě mi jako zdroj sloužila oficiální firemní dokumentace procesu a rozhovory s účastníky procesu. Tato dokumentace k procesu odkazuje na proces managementu změn tak, jak je definován v ITUP verze 7.1.4 [12]. Tuto definici procesu pak rozšiřuje a upravuje v několika bodech.

V rámci tohoto procesu jsem zjišťoval, jakým způsobem jsou jeho jednotlivé činnosti automatizovány nástrojem technické podpory a která podpora podle uživatelů případně schází.

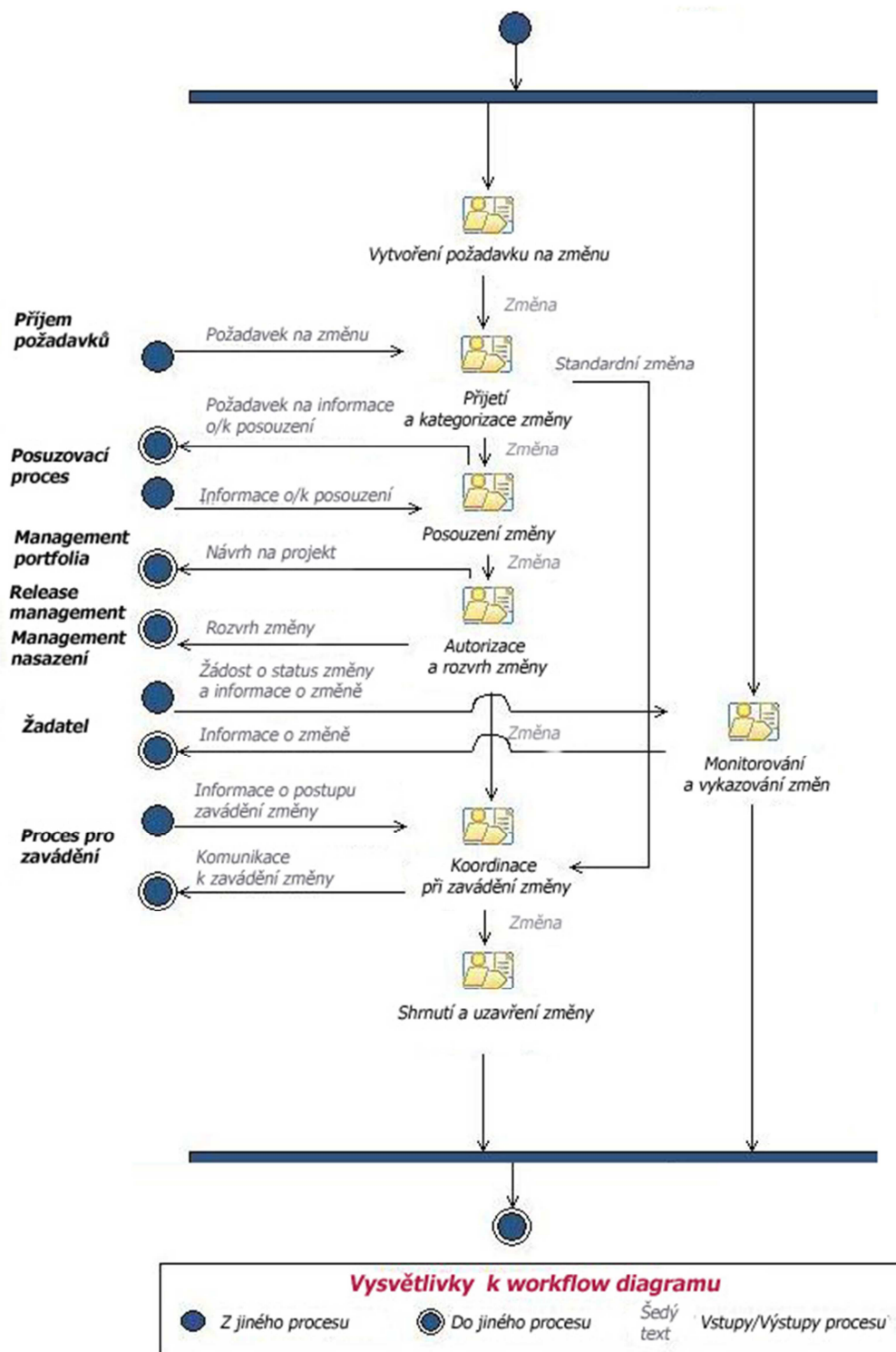
Proces managementu změn zahrnuje tyto podprocesy, kterými se budu blíže zabývat:

- Vytvoření požadavku na změnu
- Přijetí a kategorizace změny
- Posouzení změny

- Autorizace a rozvrh změny
- Koordinace při zavádění změny
- Shrnutí a uzavření změny
- Monitorování a vykazování změn

Diagram tohoto procesu je uveden na obrázku 3.1. Tento diagram je přeloženou a upravenou verzí diagramu uvedeného v ITUP verze 7.1.4.

Následuje zachycení současného stavu ze tří pohledů – úloha nástroje, role v procesu a samotné podprocesy. Jsou zachyceny některé podstatné požadavky a připomínky k aktuálnímu řešení.



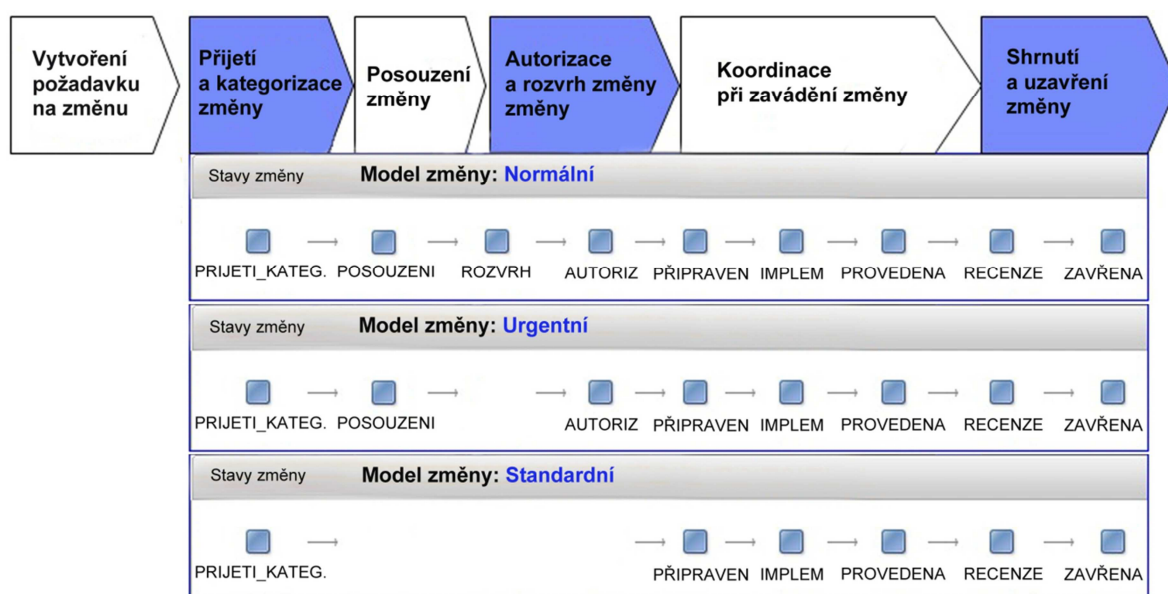
Obrázek 3.1: Diagram procesu managementu změn, zdroj: ITUP verze 7.1.4 [12], upraveno dle dokumentace firmy TechServices

3.3.1 Nástroj

Základní úlohou nástroje technické podpory je vést dokument o změně skrze celý proces managementu změn. Nástroj umožňuje, požaduje a zaznamenává některé potřebné činnosti, které by

měly být dle procesu managementu změn vykonány (a zaznamenány v dokumentu). Pokrytí činností – míra automatizace je dána konkrétní implementací.

Dokument prochází v aktuálním nástroji několika stavy, které jsou spojeny s důležitými milníky v procesu. Napojení těchto stavů na jednotlivé procesy vyjadřuje obrázek 3.2. Jedná se o stavy: přijetí a kategorizace, posouzení, rozvrh, autorizace, připravena, implementace, provedena, recenze, zavřena. Každý stav určuje, které položky dokumentu změny lze upravovat. Přejít mezi stavy vyžaduje splnění podmínek nebo přímou akci, která vychází z procesu. Například zmocněná osoba schválí změnu a stav se změní z „autorizace“ na „připravena“, v naplánovaný čas realizace změny se pak stav se změní na „implementace“.



Obrázek 3.2: Namapování stavů změny v nástroji technické podpory na proces, zdroj: dokumentace firmy TechServices

3.3.2 Role v procesu

Žadatel (Requester) je osobou, která podává požadavek na změnu formou oficiálního dokumentu (vytvořeného ve zvoleném nástroji). Požadavek na změnu může zaslat i zákazník nebo jiný oprávněný uživatel mailem, telefonicky nebo jiným dohodnutým způsobem odpovědné osobě. Tato osoba je v rámci procesu v roli žadatele, i přestože podává žádost někoho dalšího.

Aktuální nástroj umožňuje zaznamenat „*autora změny*“, „*žadatele*“ a „*za koho je žádost podána*“. Autor změny je neměnný a je to uživatel, který záznam o změně vytvořil. V rámci informací o žadateli uvádí i informaci, jakého zákazníka daná osoba představuje. Žadatel zde může zvolit jednoho z více zákazníků, pokud jich zastupuje více.

Vlastník změny (Change Owner) je osobou, která provází a dohlíží na změnu od jejího počátku do konce. Jeho odpovědností je kontaktovat zainteresované strany podle potřeb a směřovat změnu k jejímu zdárnému konci.

V nástroji má změně přiřazen vlastník. Tato položka je povinná pro stav „posouzení“, od tohoto stavu ji nelze měnit. Tím pádem může nastat problém, pokud je vlastník změny nepřítomen, protože nemůže být nikým ve své roli zastoupen.

Posuzovatel změny (Change Assessor) je odpovědný za včasné posouzení žádosti o změnu a podání schvalovacího nebo zamítacího vyjádření za danou oblast jeho působnosti. Posuzovatel je obvykle zástupce zúčastněné nebo dotčené strany. Výčet odhadců ve změně by měl představovat všechny zúčastněné a dotčené strany včetně manažera změn.

Nástroj zde nerozlišuje mezi několika rolami v procesu – analytikem změny, vlastníkem služby, finančním analytikem a business analytikem, ale rozlišuje mezi typem posudku (finanční, technický, atd.), který je této osobě nebo týmu přiřazen. Správnou volbu ponechává na vlastníkovvi změny.

Realizátor změny (Change Implementer) provádí schválenou změnu. Realizátor změny provádí přiřazený úkol, který je mu ve změně přiřazen.

V nástroji se jedná o osobu nebo tým, který má přiřazen úkol.

Manažer změn (Change Manager) je ve firmě představován celým týmem managementu změn. Tento tým je odpovědný za celkovou kvalitu změn procesu managementu změn. Tento tým je koordinátorem v procesu a kontaktním bodem pro změny jak pro zákazníka, tak pro vnitřní organizaci. V konkrétní změně také zajišťuje správnou kategorizaci, a svolává schůzky, na kterých se rozhoduje o schválení (autorizaci) změny.

Zmocněná osoba (Change Authority) je osoba, který opravňuje provedení změny. V praxi firmy je tato role zastoupena manažerem změn.

3.3.3 Podprocesy

Níže popisují jednotlivé procesy a jejich činnosti, také jejich implementace v aktuálním nástroji a nalezené problémy tohoto řešení nebo rozšíření. Na základě toho zmiňují některé požadavky, které byly získány nebo odvozeny (sběr požadavků) a dále zvažovány (analýza požadavků). Výsledné požadavky, které jsou výstupem analýzy požadavků, jsem dále zpracoval do specifikace požadavků, která byla schválena zainteresovanými stranami a je uvedena v následující kapitole 5. Z této části jsem vynechal rozsáhlé seznamy a popisy položek dokumentu. Ty jsou zachyceny až ve specifikaci požadavků. Tahle kapitola se snaží zachytit průběh sběru a analýzy požadavků.

Vytvoření požadavku na změnu

Tento proces zahrnuje činnosti v tomto pořadí: vytvoření požadavku na změnu, definování postupu při změně a odeslání požadavku na změnu.

Nástroj umožňuje **vytvořit změnu** – nový dokument. Dokument lze uložit až po vyplnění povinných položek (do stavu „uložená“). Nástroj umožňuje **vyplnit položky dokumentu a odeslat požadavek** (do stavu „přijetí a kategorizace“).

Tyto položky však neodpovídají zcela procesu. V procesu jsou uvedeny povinné položky pro vytvoření požadavku a následně povinné položky při jeho odeslání. Tento spor jsem probral týmem managementu změn. Závěrem je důležitý požadavek, aby obě skupiny povinných položek byly kontrolovány při odeslání požadavku (do stavu „přijetí a kategorizace“). Všechny tyto položky jsou potřebné pro následující činnosti. Povinné položky při vytvoření mají dle mého názoru smysl pro komunikaci a snížení počtu „prázdných“ nebo „neaktivních změn“ ve stavu uložená. Tento požadavek má však malou prioritu.

Přijetí a kategorizace změny

Podstatnou činností procesu je **filtrování, ověřování a přijetí změny manažerem změn**, na základě této činnosti rozhodne o **kategorii změny a přijetí nebo zamítnutí změny**. V průběhu tohoto procesu může manažer vyžadovat další informace.

V tomto případě by měl nástroj poskytovat manažerovi změn všechny potřebné informace pro jeho rozhodnutí. Přesto může nastat situace, kdy jsou uvedené informace nedostatečné svým obsahem, to zahrnuje činnost „**získání dalších informací**“. Aktuální nástroj však nezajišťuje vhodnou formu předání požadavku na informace. Tato činnost probíhá mimo nástroj, nástroj poskytuje kontaktní údaje (jméno a adresu elektronické pošty). Firma má jiné nástroje pro komunikaci, přesto požadavek na zprávy a požadavky v rámci nástroje byl zaznamenán. Později se mu věnuji v kapitole 3.4.1.

Manažer změn na základě těchto informací určí typ změny, odhadne riziko (viz kapitola 3.3.3) a na základě něho určí typ změny a rozhodne o přijetí nebo zamítnutí požadavku. V případě přijetí se jedná se tyto typy změn:

- Normální změna.
- Předem schválená či standardní změna.
- Urgentní změna.

Určením typu změny v nástroji určí i její následný **model (workflow)**, který určuje posloupnost stavů a tím pádem i činnosti, které mají být provedeny. Tento model stavů systému odpovídá procesu (viz obrázek 3.2). Pokud je změna typu normální nebo urgentní, je dalším stavem „posouzení změny“. Pokud má kategorii standardní, je následujícím stavem „implementace“. V případě zamítnutí je změna automaticky zavřena (stav „zavřena“).

Posouzení změny

Proces i nástroj mají 4 hlavní po sobě následující činnosti: **předběžné posouzení změny vlastníkem změny, technické a business posouzení** přiřazenými posuzovateli a následné **schválení výsledků** vlastníkem změny, kterým změnu posune dál v procesu (dle workflow). Posudky slouží jako podklad pro rozhodnutí o schválení nebo zamítnutí změny ve fázi „autorizace“.

Výsledkem prvních tří činností je posudek změny od každé přiřazené osoby nebo skupiny. V rámci každého posouzení lze vytvořit a přiřadit další posudek (případně jiné odebrat), určit dopad, odhadovanou cenu a lidské zdroje, napsat slovní vyjádření a poznámku.

V předběžném posouzení vlastník určí kromě vlastního vyjádření a hodnot hlavně potřebné posudky a přiřadí je osobám. Určí i oblast působnosti daného posudku z několika kategorií, které spadají do technické nebo business oblastí. Systém tak zajišťuje i pořadí posudků dle procesu, kdy technické posudky mají předcházet těm s business kategorií.

Poslední činností, kterou systém umožňuje provést, pokud neexistuje žádný nepotvrzený posudek, je souhrnné schválení vlastníkem změny. Změna je tak posunuta do dalšího stavu, pokud se jedná o změnu s kategorií normální, je posunuta do stavu „plánování“, pokud má kategorii „urgentní“ je posunuta do stavu „autorizace“.

Obvykle dává poslední business posudek manažer změn, tím zajišťuje správnost. Systém automaticky tento posudek nekládá. Z toho vychází další požadavek, podle kterého by nástroj mohl automaticky považovat tento posudek za povinný.

Autorizace a rozvrh změny

Činnosti tohoto procesu jsou v nástroji rozděleny do dvou samostatných částí. První je proveden rozvrh změn a až následně autorizace. Toto rozdělení má být dle týmu managementu změn zachováno.

V rámci činností rozvrhu určuje čas počátku a konce provádění změny. Také může v nástroji vytvořit několik úkolů, které jsou součástí změny a přiřadit je osobám nebo skupinám. Každému úkolu může také nastavit vlastní čas počátku a konce. Ten musí být v rámci času provedení změny. Při potvrzení plánu systém zkontroluje určení časů změny. Následně změna přejde do stavu „autorizace“.

V průběhu autorizace může osoba s dostatečným oprávněním (zmocněná osoba) schválit změnu k provedení. Systém umožňuje 4 úrovně, nejčastěji je však využívána jedna, kterou má každý manažer změn. A prakticky ji provádí v systému vždy právě manažer změn. Mimo systém manažer změn organizuje pravidelné schůzky, na kterých jsou důležité změny probírány. Pokud to změna vyžaduje, je rozhodnutí dáno na těchto schůzkách a manažer změn jej promítne do systému. Pro organizaci těchto schůzek potřebuje manažer změn exportovat data o změnách do vhodného formátu.

Koordinace při zavádění změny

Součástí tohoto procesu je zaznamenávání stavu implementace realizátory změny a monitorování změny vlastníkem změny. V nástroji je toto provedeno skrze stavy úkolů, které by měl realizátor změnit při dokončení úkolu, neúspěšném provedení atd.

Shrnutí a uzavření změny

Vlastník změny má v rámci činností tohoto procesu zhodnotit změnu a zavřít ji po té, co byla provedena úspěšně nebo i neúspěšně. Zavřít změnu lze kdykoliv v jejím průběhu. Při zavírání změny lze uvést stav, s kterým byla změna ukončena a v případě neúspěšné změny i důvod neúspěchu. Důvod neúspěchu je zadán slovně a výběrem kódu. Kód slouží pro sledování statistik příčin neúspěšných změn.

Monitorování a vykazování změn

Tento proces probíhá souběžně, jsou zde činnosti kontroly stavu, sledování postupů, identifikace důležitých záznamů, vytváření reportů atd. V rámci této práce nejsou požadovány, jsou ale podstatné při produkčním zavedení nástroje managementu změn. Týká se to například požadavku na exportování nebo rozhraní do nástroje pro sběr a analýzu metrik.

3.3.4 Pojmy procesu

Zde jsou blíže uvedené některé pojmy popisovaného procesu.

Riziko změny

Riziko změny je ve firmě definováno vztahem (1):

$$\text{Riziko změny (Risk)} = \text{Dopad (Impact)} \times \text{Pravděpodobnost (Probability)} \quad (1)$$

Riziko je událost, která může nastat a v případě, že nastane, má negativní dopad na služby. Dopad (Impact) udává míru škody, kterou daná událost způsobí. Velikost dopadu se ale liší a to jak v rámci na technologie, obchodu (business) nebo jiného kritického faktoru. Pravděpodobnost (Probability) je pravděpodobnost, že tato událost nastane. Může být vyjádřitelná v procentech, ale pro potřeby určování rizika se užívá kategorií.

Kategorie změny je spojená s rizikem změny. Tabulka 3.1 zobrazuje jak je ve firmě aktuálně odvozena kategorie změny na základě dopadu a pravděpodobnosti rizika. Tato tabulka je daná zkušenostmi a dohodou se zákazníkem, může být tedy v budoucnu změněna.

Tabulka 3.1: Určení kategorie změny dle rizika (dopad x pravděpodobnost), zdroj: dokumentace firmy TechServices

		Dopad			
		1 – Kritický	2 – Velký	3 – Střední	4 – Malý
Pravděpodobnost	1 – Velká	Kategorie 1	Kategorie 1	Kategorie 2	Kategorie 3
	2 – Střední	Kategorie 1	Kategorie 2	Kategorie 3	Kategorie 4
	3 – Malá	Kategorie 2	Kategorie 2	Kategorie 4	Kategorie 4

Vyhodnocení rizika je založeno na kombinaci vyhodnocení míry dopadu a pravděpodobnosti rizika.

Modely změn

Typ změny je dán její schopností splnit daná kritéria pro změny (procesu změn). V managementu změn firmy jsou uváděny základní tři typy změn a modely s nimi spojené:

1. Normální
2. Urgentní
3. Standardní

Časy povinného předstihu

Žadatel a vlastník změny jsou odpovědní vést změnu skrze proces a zajistit, že změna odpovídá definovanému procesu a také, že je dodržena požadovaná kvalita. Při vedení změny musí dodržet lokálně definované časy pro podání změny v předstihu před provedením změny. To má zajistit dostatek času na korektní posouzení nebo obdobně v případě schválení. Změny, jejichž průchod procesem je rychlejší, jsou považovány za výjimku a taková výjimka musí být schválena požadovanou autoritou.

Aktuální cílové časy, v kterých musí být změna kompletní a přístupná pro zhodnocení jsou zobrazeny v tabulce 3.2.

Tabulka 3.2: Cílové časy podle kategorie změny, zdroj: dokumentace firmy TechServices

Kategorie	1	2	3	4	5
Předstih podání	14 dní	8 dní	2 dny	1 den	0 dní
Předstih schválení	4 dny	2 dny	1 den	Není	Není

Časy povinného předstihu jsou spojeny s kategorií změny a tedy rizikem spojeným se změnou. Cílem je, aby byl vždy dostatečný čas na posouzení všech částí změny. Časy povinného předstihu pro schválení určují čas, v kterém musí být změna již schválena před samotným provedením změny. Tento čas je stanoven za účelem dostatečné přípravy k provedení změny.

System hlídá tyto časy a v případě jejich porušení vyžaduje k pokračování změny akci (schválení výjimky) od manažera změn.

3.3.5 Management konfigurací

Ve firmě je u konkrétního zákazníka, v rámci kterého jsem projekt prováděl, tento proces zaveden omezeně. Pro současný nástroj managementu změn jsou využity třídy konfiguračních položek, ale schází databáze jednotlivých konfiguračních položek. Navržení databáze konfigurací spadá pod cíle této práce, ale zavedení konfigurační databáze je již součástí jiného očekávaného projektu. Tato oblast by také přesahovala časové možnosti této práce.

3.4 Hledání příčin a návrh řešení

Mým dalším krokem po seznámení se s procesem, aktuálním nástrojem a účastníky procesu bylo určení problémů, které mohou mít vliv na dosažení stanovených cílů. K těmto problémům jsem pak hledal jejich příčiny a tyto znalosti pak použil pro návrh řešení.

V této části jsem sledoval dva hlavní cíle, které byly stanoveny v předchozím kroku, a to „prevence neúspěšných změn“ a „úspory času“. Každý z těchto cílů vyžadoval trochu jiný přístup, přesto měly některé společné příčiny a tím pádem i řešení.

Využil jsem analýzy dokumentace (zvláště výsledků předchozích „identifikací příčin“ sledovaných problémů, které firma eviduje, firemních školení procesu) a rozhovorů s účastníky procesu. V případě některých hypotéz o souvislostech jsem vycházel z názorů firemních expertů na danou oblast.

3.4.1 Prevence neúspěšných změn

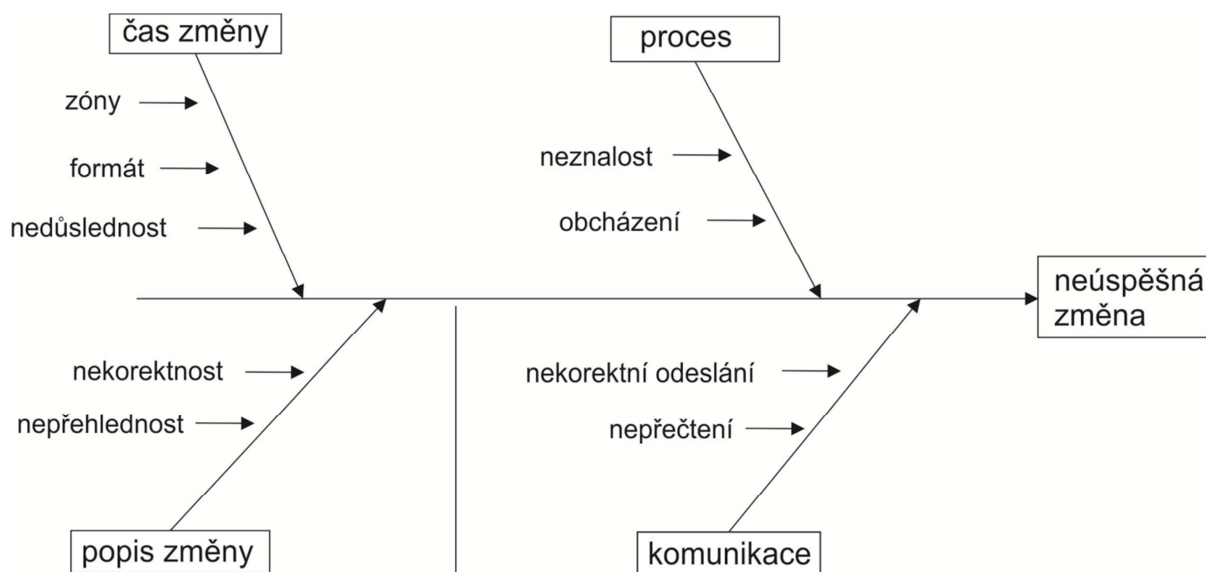
Neúspěšná změna je změna, která měla negativní dopad na služby jiný než schválený, nebo změna, která nedosáhla schválených výstupů ve schválený čas. Za neúspěšnou změnu se také považuje změna, která byla provedena mimo proces.

Velkou pomocí mi tu při hledání příčin neúspěšných změn byl fakt, že firma již dlouhodobě sleduje a řeší každou neúspěšnou změnu, aby jejich výskyt dále minimalizovala.

Vlastník změny je povinen pro každou neúspěšnou změnu identifikovat a zdokumentovat její příčinu, stanovit a případně provést preventivní akce. Doporučenou technikou je tu „pětkrát proč.“ Kvalita těchto identifikací a příčin je hlídána.

Toto mi poskytlo hlavní zdroj pro určení příčin včetně jejich četností. Další diskuzí s týmem managementu změn jsem hledal další souvislosti a příčiny. A následně ve spolupráci se všemi účastníky procesu jsem hledal možná řešení. Tato řešení jsem srovnal s aktuálním nástrojem.

Na obrázku 3.3 je uveden Ishikawův diagram⁵ („rybí kost“), který jsem při práci použil. Diagram zachycuje přehled dílčích oblastí a příčin.



Obrázek 3.3: Hledání příčin neúspěšných změn, Ishikawův diagram, zdroj: autor

Čas změny – nejasnosti v časech

Změna provedená v jiném čase, než by měla být provedena, může mít velký dopad na služby. Provedení změny ve správném čase je kritickým faktorem úspěšné změny. Pod „správným“ časem zde rozumíme čas s minimalizovaným rizikem dopadu na služby, ne vždy se jedná právě o „schválený“ čas. K chybné interpretaci času může dojít i v průběhu schvalování změny.

Nejasnosti v časech mohou vycházet z různých časových formátů nebo zón a nebo nedůsledné kontroly při vytváření, hodnocení, schvalování nebo provádění změny. Uživatel může například pochybit v interpretaci času „6:00“ pokud není explicitně uvedeno, zda je daný čas ve formátu 12H (a.m./p.m.) nebo 24H.

Navržené řešení: Systém by měl proto zobrazovat časy s explicitně uvedeným formátem i zónou. Všechny podstatné časy musí být jasně zobrazeny v průběhu kategorizace, hodnocení, provádění i schvalování, protože hodnotitel, schvalující i provádějící změny jsou odpovědní za kontrolu a důsledné přečtení těchto údajů. Aktuální nástroj tento problém řeší výše uvedeným způsobem.

⁵ Ishikawův diagram - Technika, která pomáhá týmu identifikovat všechny možné příčiny problému. Původně navržena Kaoru Ishikawou.

Komunikace – nedostatky komunikace

Nedostatek komunikace má často za následek neúspěšné provedení změny, ať už jejím nekorektním provedením, provedením nesplňujícím očekávání nebo její kompletním neprovedením. Míst potřebné komunikace v procesu, a tím pádem i rizikových míst v procesu, je více.

Nejkritičtější je získávání posudků. Vlastník změny je zde odpovědný kontaktovat posuzovatele a zajistit včasné posouzení změny. Posuzovatel je odpovědný pravidelně sám kontrolovat svou frontu požadavků na posouzení. Přesto často dochází k tomu, že posouzení není dodáno včas.

Příčina je až příliš často v komplikovaném vyhledání správné osoby v systému, která má poskytnout posudek. I pokud vlastník zná skupinu odhadce, neví kdo je z této skupiny odpovědný. Dochází pak k tomu, že jsou kontaktovány osoby, které nemají odpovědnost ani oprávnění tento posudek dávat. Toto je způsobeno tím, že rozdělení těchto skupin a oprávnění v systému není dostatečné pro potřeby procesů. Například z důvodu získání přístupu k datům skupiny (statistiky) jsou v systému členy skupin i osoby, které nejsou členy skupiny i mimo systém a tedy nenesou danou roli odhadce v procesu. To vede k problematickému způsobu hledání odpovědné osoby odpovědné za schválení.

I pokud se vlastník rozhodne kontaktovat všechny uživatele skupiny, bude to obtížný úkol. V systému je sice povinně uvedená mailové adresa, ale kopírování každé jednotlivé adresy ze skupiny odhadce může být zdlouhavé. Některé skupiny mají více než dvacet členů.

Řešení: Zavedení rolí v rámci jedné skupiny podle rolí procesu. A komunikace pomocí zpráv uvnitř systému, upozornění na požadovanou akci i na zprávy v systému, zasílání zpráv na mailovou adresu přímo ze systému. Tento problém aktuální nástroj neřeší dostatečně.

Popis změny – nedodržení popisu a nekorektní popis

Komplexní změna může mít dlouhý popis s různou relevantností k jednotlivým úkolům. Úkoly pak mají vlastní popis. Při zobrazení úkolu není současně zobrazen popis změny. Pro prohlédnutí popisů úkolů je potřeba otevřít každý úkol zvlášť. Tyto fakty znesnadňují přístup ke kompletnímu přehledu o popisu, vedou k různé kvalitě popisu a často vynechané kontrole.

Řešení: Při zobrazení úkolu zobrazit i popis změny. A umožnit zobrazení přehledu všech úkolů včetně popisů bez otvírání jednotlivých úkolů. V tomto bodě má aktuální nástroj tyto zmíněné nedostatky.

Proces – nedodržení procesu

Hlavním problémem je tu provádění neautorizovaných změn. Jednou z důvodů vyhnutí se provedení změny dle procesu jsou příliš náročné činnosti v procesu změny. Nástroj by měl mnohé činnosti usnadnit.

Druhou příčinou je neznalost procesu. Zde by nástroj mohl také přinést usnadnění v podobě průvodce a nápovědy.

Oba tyto body jsou náplní druhého cíle „úspory času“, který je popsán níže.

3.4.2 Úspora času

Podnět k tomuto cíli přišel od uživatelů ve formě požadavku na změny v podpůrném nástroji, které by jim usnadnily provádění činností procesu managementu změn. Proč je tento cíl ale nastaven jako úspora času?

Protože business cíle jsou důvodem pro změny. I pokud by mým vlastním cílem bylo zlepšení podmínek pro uživatele, tak business odůvodnění ke změnám má ve firmě větší podporu. Odpovědná osoba za změny ve firmě mi doporučila, abych našel odpověď na otázku „Jak jsou tyto požadavky spojeny se sledovanými metrikami procesu, jeho výstupu a tím pádem i business cíli?“

Kritérii úspěchu se stává zlepšení v daném procesu nebo jeho výstupu, které můžeme sledovat pomocí metrik.

Výtky ze strany vlastníků změn, odhadců, žadatelů a realizátorů změn byly tyto:

- Nepochopení „co se od nich požaduje“ u některých činnostech.
- Obtížné dohledání některých informací.
- Problém s kontaktováním odpovědných osob (zvláště získávání posudků).
- Zdlouhavé vytváření změny (požadavku na změnu).
- Nutnost přeplánovat změnu znovu, protože nebyly získány všechny posudky včas.

Ze strany manažera změn (týmu managementu změn) je to pak prakticky stížnost na následky těchto problémů ostatních rolí a tím jsou časté dotazy, nekorektně vyplněné změny a včas nezískaná posouzení, které musí manažer změn řešit.

Po debatě s týmem managementu změn jsem určil tyto dva parametry procesu, které jsou v souvislosti s těmito výtkami účastníků procesu a mají vliv na business:

1. Čas strávený činnostmi v procesu. Tato hodnota má vliv na úsporu času.
2. Procento změn, které nebyly ani schváleny ani zamítnuty, ale byly zrušeny v důsledku chybějícího posouzení, ale zrušené, nebo změny špatně vytvořené, které byly zrušené

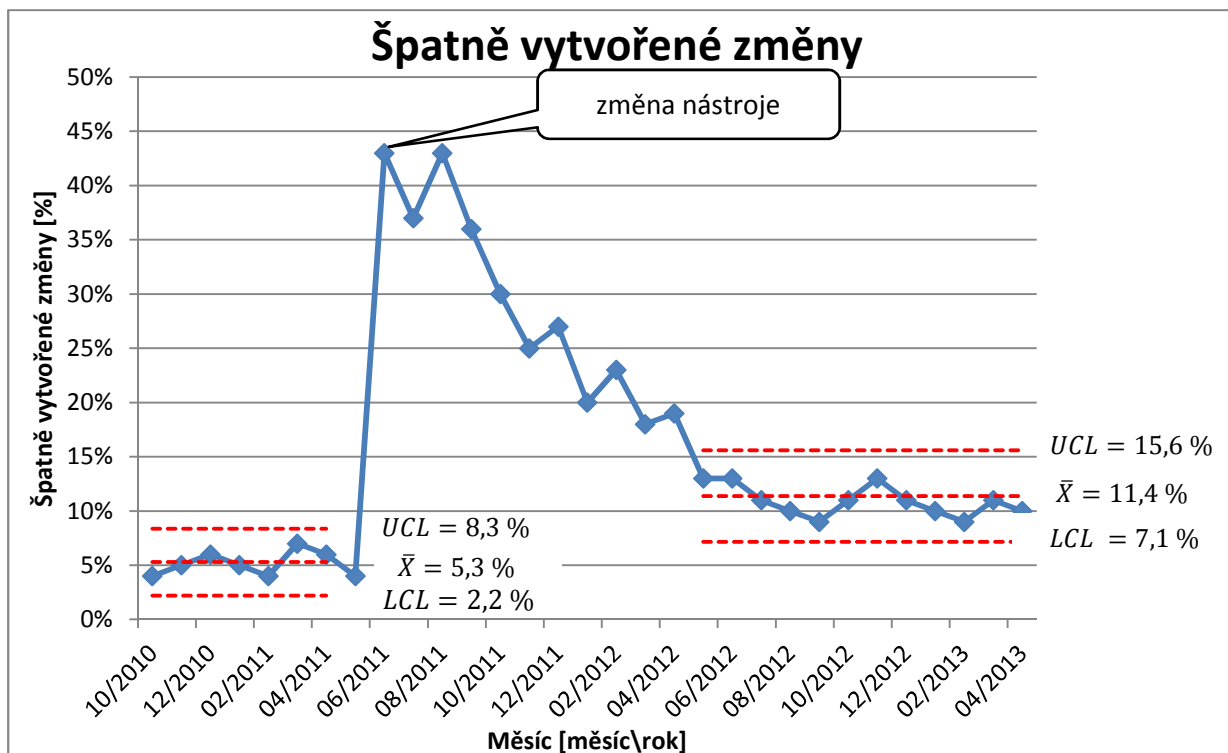
předtím, než došly k akceptování. Nesprávné vytvoření změn ohrožuje naplnění požadavků zákazníka, protože požadovaná změna nemusí být provedena včas.

V rámci prvního bodu „času stráveného činnostmi“ nejsou ve firmě sbírány relevantní metriky. Problém je dle mého úsudku v náročnosti sbírání těchto metrik tak, aby měly skutečně vypovídající hodnotu. Zde jsem absenci těchto metrik řešil pomocí několika měření před a po změně. Tyto pak sloužily k vyhodnocení vlivu provedených změn na tento cíl. Postup a výsledky jsou uvedeny ve výsledcích v kapitole 6.

Druhý bod se týkal špatně vytvořených změn. Relevantní metrikou je tu procento špatně vytvořených změn. Procento špatně vytvořených změn je metrika odvoditelná z počtu špatně vytvořených změn a počtu všech otevřených změn. Pro procento špatně vytvořené změny existuje ve firmě pravidelný sběr metrik v týdenním intervalu nebo je lze odvodit z historie všech záznamů změn v nástrojích, jak uvádí vzorec (2).

$$\text{špatně vytvořené změny [\%]} = \frac{\text{špatně vytvořené změny}}{\text{všechny vytvořené změny}} \quad (2)$$

Graf zobrazující trendy je jedním z nástrojů užívaných i ve firmě TechServices k zobrazení vývoje měření v čase. Tento graf pomáhá porozumět tomu, jak se mění výkonnost. Pomocí těchto grafů jsem provedl několik analýz týkajících se dat o počtu špatně vytvořených změn několik analýz. Nejvýmluvnější je graf zachycující vývoj špatně vytvořených změn za období zhruba posledních dvou let (viz obrázek 3.4). Uvedený graf je shrnutím více analýz, nedodržuje všechna pravidla postupu, tak jak jsou používány ve firmě TechServices, přesto nejlépe poslouží pro zachycení vývoje a událostí.



Obrázek 3.4: Graf špatně vytvořených změny za období posledních dvou let, zdroj: autorem práce zpracovaná historická data záznamů o změnách firmy TechServices

Do tohoto grafu jsem červeně zakreslil kontrolní hladiny (limity) standardního chování. UCL je horní limit, LCL je spodní limit a \bar{X} je středová hodnota. Limity jsou zde dány posunem od středové hodnoty o trojnásobek standardní odchylky v populaci. Porušení těchto limitů značí posun v chování procesu nebo snížení jeho stability [10].

Odhalená změna nebo trend vyžaduje hledání jeho příčiny a případnou korekci. Při analýze těchto změn a trendů jsem použil základní dvě otázky z ITIL [7] a to:

- „Je změna/trend dobrá nebo špatná?”
- Existuje nějaký důvod, je tato změna nebo trend očekávaný?

Analýza:

- V období před červnem 2011 má procento špatně vytvořených změn poměrně ustálené se středovou hodnotou 5,3 %.
- V červnu 2011 dochází k velkému překročení limitů, a to dlouhodobě. Tato změna je špatná, větší počet špatně vytvořených změn je spojen s horším výkonem procesu a ohrožení výstupů. Tato změna byla ale očekávaná. V červnu 2011 byl zaveden nový nástroj technické podpory a problém se zaškolením, navyknutím si na nový nástroj, odporem ke změnám byl očekáván.
- O pár měsíců později je po několik měsíců viditelný trend snižování procenta špatně vytvořených změn. Toto odpovídá očekávanému návratu po odeznění problému s navyknutím na nový nástroj.

- Zhruba od června 2012 se hodnota znovu ustálila, již nelze pozorovat trend snižování. Středová hodnota je 11,4 % za toto období.

Soudím tedy, že je možný vliv nového nástroje na zvýšenou středovou hodnotu i po odeznění problému s navyknutím. Nový nástroj nemusí být jedinou příčinou. Přesto tento závěr považuji za dostatečný, abych si odůvodnil další práci s tímto problémem.

Následně jsem hledal možná zlepšení. Protože podnět přišel od uživatelů, získal jsem od nich jejich konkrétní návrhy a diskutoval je dále se všemi účastníky procesu (uživateli nástroje).

Průvodce

Současný nástroj nenavádí dostatečně uživatele v rámci procesu, potřebných aktivit, účelu a možností. Toto vše vede k mnoha špatně navrženým změnám, nedostatečným kontrolám, špatným návykům, komplikacím při implementaci a tím pádem k většímu počtu neúspěšných změn a větší časové náročnosti pro technické týmy i týmy managementu změn.

Malá znalost procesu v týmech, složité pochopení procesu a jeho nepřesné provozování vede k obtížím začínajících uživatelů a ke špatným návykům. Uživatelům nemusí být jasné jejich odpovědnosti.

Řešení: Zavést průvodce do nástroje, který bude průběžně popisovat požadované činnosti.

Vzory změn

Vytváření několika podobných změn může být náročné na rutinu vyplňování neustále stejných položek.

Také u nového požadavku není často jasné, co má změna obsahovat za informace, které budou nutné pro její posouzení, pro realizátory atd. Proces toto řeší tím, že odpovědnost za získání těchto znalostí má vlastník, který by měl mít v této oblasti zkušenosti. Často se však stává, že žadatel změny se stává i vlastníkem místo strany, která by měla požadavek obdržet a změny v dané oblasti obvykle provádí nebo řídí. V tomto případě je problém z velké části na nekorektním provádění procesu jejími účastníky a tento problém částečně může řešit i výše zmíněný průvodce a zlepšení komunikace. Poskytnutý vzorový formát změny je dobrým prostředkem komunikace, která by měla usnadnit další získávání informací i urychlit vytvoření korektní změny.

V navrženém řešení by měla skupina mít možnost vést svou sadu vzorů, které budou viditelné ostatním uživatelům a ti na jejich základě budou vytvářet změny.

Při diskuzi možnosti využívání vzorů jsem se setkal s upozorněním na rizika spojené s jejich užíváním. Hlavní problém byl spatřen v dopadu neaktualizovaných vzorů. K tomuto problému jsem našel v dokumentaci firmy související povinnost, kterou je potřeba zahrnout do řešení a tím pádem i snížit zmíněné riziko. Z procesu managementu dokumentů vyplývá povinnost pravidelného

schválení platnosti všech aktivních dokumentů v daném intervalu. Schválení platnosti vlastníkem by mělo značit, že dokument zkontroloval a je v aktuálním stavu. V rámci systému se tato podmínka tedy vztahuje na tyto vzory změn. Z toho plyne požadavek na funkci hlídání stáří vzorů, jejich automatickou deaktivaci po vypršení intervalu, možnost schválení platnosti vzoru jeho vlastníkem, které prodlouží jeho platnost.

Tento návrh vyžaduje změny i v procesu, jinak se může minout účinkem. Proces by měl být rozšířen o činnosti „vytvoření vzorů změn“, „poskytnutí vzorů změn“, „schválení vzorů změn“ a „vytvoření požadavku na změnu“ rozšířit o možnost využití vzoru.

Upozornění na rizikové změny

Automatická upozornění sice mohou určit některé problémy, ale mohou také vést přenechávání odpovědnosti na systém a tím pádem ke snížení pozornosti všech zúčastněných.

3.4.3 Další související problémy

Při úvaze řešení souvisejících s nástrojem a automatizováním částí procesu jsem narazil ještě na jeden podstatný bod pro implementaci takového nástroje.

Riziko absence přístupu

Oprávněná osoba (skupina) nebude mít přístup k nástroji v potřebném čase pro aktivitu ohodnocení nebo schválení změny. Možným dopadem je opoždění změny až nenaplnění potřeb, za kterým měla být změna provedena. Trvalou absenci přístup není potřeba uvažovat.

Aktuální systém umožňuje obejít proces a přesunou změnu ručně, do kteréhokoliv stavu, avšak nezmění se tím vnitřní automatický tok aktivit a stavů v systému. Mnohé aktivity je tak potřeba provést bez podpory systému.

Řešení: Manažer změn může zprostředkovat kontaktování dané osoby (skupiny) a získat její vyjádření. Následně manažer změn může podat posouzení nebo autorizaci do systému v zastoupení za danou osobu (skupinu). Tato akce je zaznamenána. Zastoupený uživatel může být informován na adresu elektronické pošty.

3.4.4 Shrnutí a návrh experimentu

Byly navrženy tyto změny:

- Zavedení rolí ve skupině v nástroji technické podpory.
- Přidání možnosti zobrazení souhrnného přehledu úkolů změny včetně popisu těchto úkolů.
- Přidání možnosti zobrazení i popisu změny při zobrazení popisu úkolu.

- Přidání možnosti průvodce, který bude navádět uživatele požadavky jednotlivých činností.
- Vytvoření systému vzorů změn, tato změna vyžaduje pro svůj přínos změny v nástroji i v procesu a personální přiřazení role koordinátora změn ve skupině.

Přínos některých těchto bodů by bylo vhodné ověřit experimentem. Zadavatelem byl vybrán návrh na rozšíření nástroje o vzory změn. Zadavatelem byl vznesen požadavek na vytvoření upraveného nástroje, který umožní ověření přínosu tohoto návrhu a jeho prezentaci. Test měl být proveden bez produkčních dat, tedy pomocí připravených testů. Tento nástroj by měl primárně implementovat ty požadavky, které jsou nutné pro tento experiment.

4 Specifikace požadavků a návrh

V této kapitole jsou zdokumentovány požadavky na nástroj technické podpory managementu změn, které jsou výstupem analýzy požadavků. Přesněji řečeno jsou tyto požadavky konečnou podobou požadavků, které provázely analýzu požadavků ve více iteracích a byly v této podobě zdokumentovány a schváleny zainteresovanými stranami. Jedná se o detailní popis požadavků (na technické úrovni).

Jsou zde popsány požadavky na nástroj, který bude implementován a využit pro prezentaci a ověření přínosu navržených úprav pomocí experimentu. Tyto požadavky jsou podstatné pro dosažení cílů této práce. Jsou zde uvedené i požadavky, které nemusí být implementovány v rámci tohoto projektu, ale byly vzneseny a schváleny pro případný produkční nástroj. V rámci tohoto projektu však mají menší prioritu. U těchto požadavků je tento fakt explicitně uveden.

Požadavky určují, co má systém dělat, neurčují, jak to má dělat. Funkční požadavky, které definují očekávané služby nebo chování systému, jsou zachyceny pomocí případů užití a doplňujících informací a také popisu specifických funkcí. Následně je uveden soupis nefunkčních požadavků. Nefunkční požadavky specifikují vlastnosti nebo omezení, které musí systém splňovat. Na úvod je zde uvedena krátká část neformální specifikace.

4.1 Neformální specifikace

Nástroj pro technickou podporu managementu změn má za účel poskytnout některé funkce pro proces managementu změn dle procesu zadávající firmy. Mělo by se jednat o ty funkce, které jsou podstatné z hlediska efektivity práce, úspěšnosti celkového managementu služeb a spokojenosti zákazníků zadávající firmy.

Systém bude evidovat záznam o změně jako elektronický dokument a umožňovat činnosti spojené s tímto dokumentem tak, jak jsou popsány v procesu a nařízeních firmy.

Priorita těchto požadavků je dána v souladu se schválenými cíli práce a zaměřuje se na provedení experimentu s navrženými změnami.

4.2 Specifikace případů užití

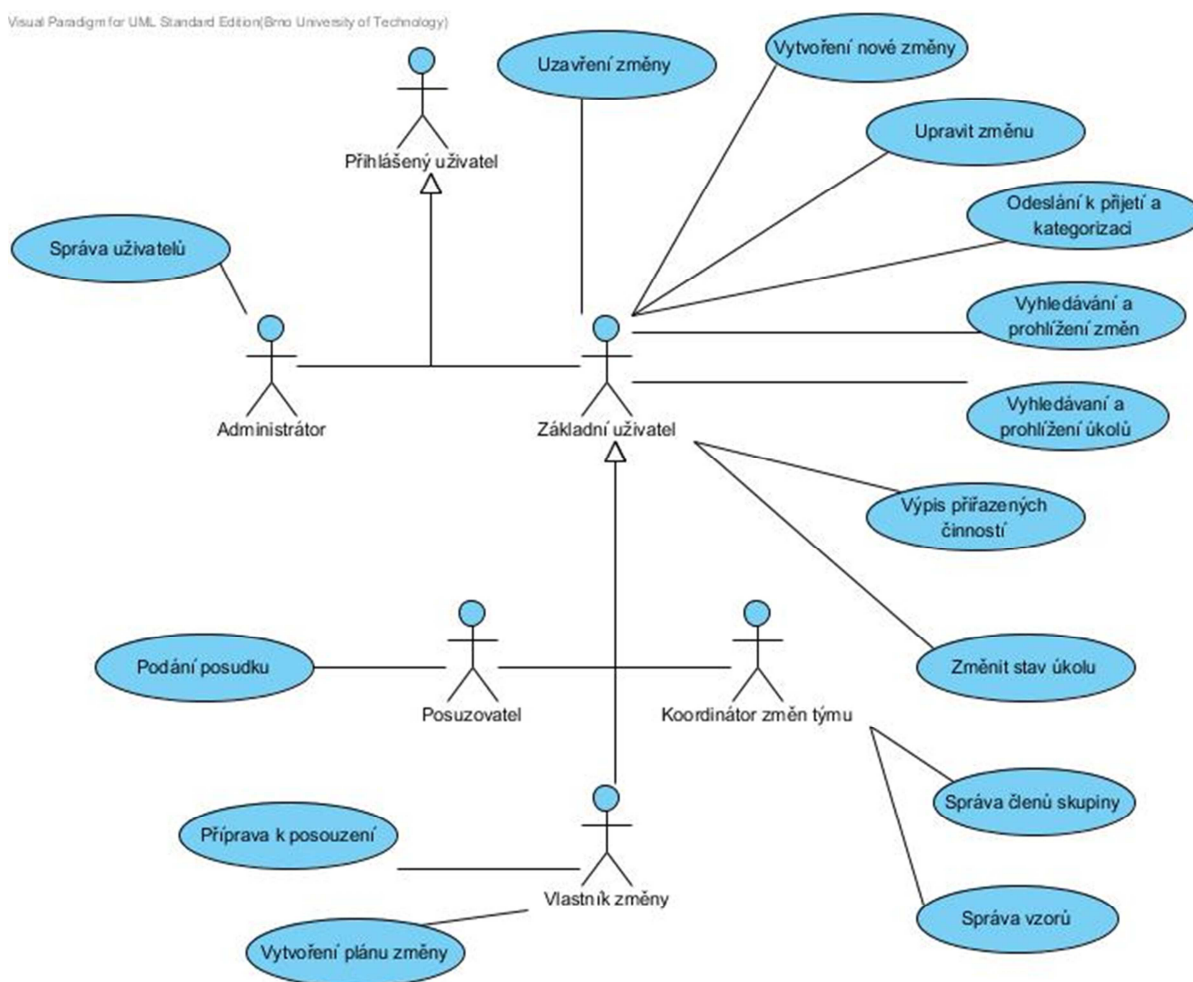
Vhodnou formou pro specifikaci požadavků jsou případy užití, které se využívají i v předchozích fázích pro komunikaci se zákazníkem. Případy užití popisují požadavky z pohledu jednotlivých uživatelů. Zde uvedené případy užití jsou přeloženou verzí oproti požadavkům vytvářeným i prezentovaným zákazníkovi. Původní případy, prezentované zákazníkovi, byly v anglickém jazyce.

Součástí specifikace případu užití je diagram případů užití a textový popis jednotlivých případů užití.

Diagram případů užití obsahuje tyto základní komponenty [13]:

- hranice systému – ohraničení okolo případů užití vyjadřují hranice modelovaného systému
- aktéři – vyjadřují role osob nebo předmětů používajících systém
- případy užití – činnosti, které systém umožňuje aktérům vykonávat
- relace – vyjadřují interakci mezi aktérem a případem užití, mezi případy užití navzájem nebo mezi aktéry

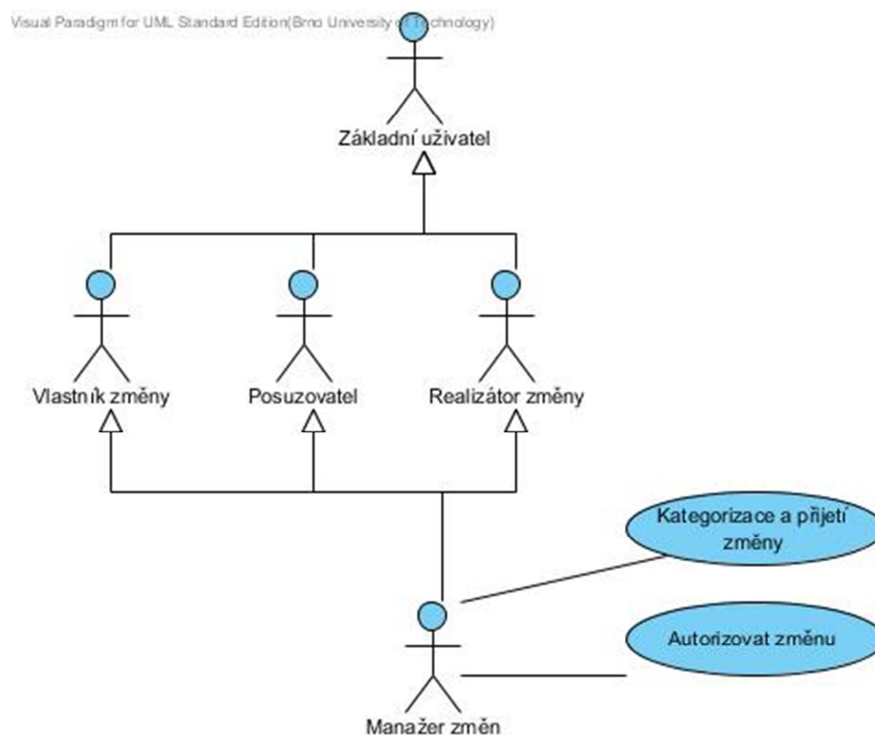
Diagramy případů užití této specifikace jsou uvedeny na obrázku 4.1 a obrázku 4.2. Pro větší přehlednost jsou vztahy rozděleny do více diagramů a nezachycují vztahy „include“. Modelované diagramy jsou v jazyce UML⁶ 2.0 (Unified Modeling Language) a jako podpůrný modelovací nástroj jsem použil nástroj Visual Paradigm⁷ for UML 10.0.



Obrázek 4.1: Digram případů užití, část zobrazující aktéry kromě manažera změn, zdroj: autor

⁶ UML, Unified Modeling Language je grafický jazyk pro vizualizaci, specifikaci, navrhování a dokumentaci programových systémů.

⁷ Visual Paradigm je balíček nástrojů pro vizuální návrh architektury aplikací a její vyjádření pomocí notace jazyka UML.



Obrázek 4.2: Digram případů užití, část zobrazující manažera změn, zdroj: autor

4.2.1 Aktéři a role

Některé role jsou obecné v celém systému (můžeme říci, že se dědí ke všem změnám v prostředí) a jiné role jsou získané, ty vznikají nebo zanikají akcí nad konkrétní změnou. Příkladem obecné je „Manažer změn“, který svoji roli plní v celém systému ke všem změnám jednoho zákazníka. Role „Vlastníka změny“ je vázaná ke konkrétní změně, tuhle roli může získat Základní uživatel. Níže jsou uvedeni jednotliví aktéři a role systému.

Přihlášený uživatel reprezentuje abstraktního aktéra.

Základní uživatel představuje aktéra, který se může vyskytovat ve více rolích v procesu, často současně v různých rolích k jedné změně nebo v různých rolích k různým změnám, ale i v žádné z těchto rolí.

Manažer změn (Change Manager) je odpovědný za celkovou kvalitu změn procesu managementu změn. Zajišťuje správnou kategorizaci a průběh procesu, může v systému zastoupit v činnostech některé jiné role.

Zmocněná osoba (Change Authority) představuje aktéra, který opravňuje provedení změny. Podle schválených požadavků může být v systému nahrazen manažerem změn⁸.

Administrátor představuje aktéra, který spravuje uživatele a jejich oprávnění v systému. Tento aktér má být (v produkčním prostředí) oddělen od ostatních uživatelů systému.

⁸ Role je zde zmíněná pro lepší referenci na proces. A také vzhledem k tomu, že tento požadavek byl sporný a lze očekávat jeho pozdější změnu.

Koordinátor změn týmu (Change Focal Point) představuje aktéra, který v systému vytváří vzory změn za jednotlivé skupiny.

Níže jsou uvedené získané role, které jsou platné pouze k dané změně, úkolu nebo posudku.

Žadatel (Requester) je uživatelem, který podal žádost na změnu.

Vlastník změny (Change Owner) reprezentuje uživatele, který provází a dohlíží na “změnu” od jejího počátku do konce. Kontaktuje zúčastněné (zainteresované) strany podle potřeb a směřuje změnu k jejímu zdárnému konci.

Posuzovatel (Change Assessor) je odpovědný za včasné ohodnocení žádosti o změnu a podání schvalovacího nebo zamítacího vyjádření za danou oblast jeho působnosti. Posuzovatel je obvykle zastupujícím zúčastněné nebo dotčené strany.

Realizátor změny (Change Implementer) provádí jemu přiřazený úkol. V systému může být nahrazen libovolným „základním uživatelem“.

4.2.2 Popis případů užití

Níže je uveden textový popis případů užití. U některých případů tu nejsou uvedeny následné podmínky, vedlejší toky a výjimky, přesto by měl stále zachycovat nejpodstatnější detaily pro specifikaci požadavků.

PU01 – Vytvoření nové změny

Primární aktér: Základní uživatel.

Akce pro spuštění: Uživatel zvolí „nová změna“.

Hlavní tok:

1. Systém nabídne uživateli vytvoření změny „bez vzoru“ nebo „dle vzoru“.
2. Pokud uživatel zvolí „bez vzoru“.
 - 2.1. Systém vytvoří prázdnou změnu.
3. Pokud uživatel zvolí „dle vzoru“.
 - 3.1. Systém uživateli umožní vyhledávání dle skupiny (Owner Group) a popisu vzoru (Summary).
 - 3.2. Uživatel vybere změnu.
 - 3.3. Systém vytvoří novou změnu dle vzoru (zkopírováním popisu (Summary), detailního popisu (details), položek úkolů a posudků).

Dodatek: Uživatel může alternativně vstoupit do 3. kroku akcí „vyhledávat vzory“. Krok 3.2 nemusí být umožněn, pokud nebyl nalezen žádný odpovídající změnu, uživatel může libovolněkrát opakovat krok 3.1.

PU02 – Vyhledávání a prohlížení změn

Primární aktér: Základní uživatel.

Akce pro spuštění: Uživatel zvolí „vyhledat“.

Hlavní tok:

1. Systém nabídne uživateli zadání filtrů pro vyhledávání změn.
2. Uživatel zadá položky do filtrů a potvrdí.
3. Systém zobrazí odpovídající záznamy změn v tabulce, kde každý řádek tvoří jednu změnu.
4. Uživatel může vyhledané položky seřadit.
5. Uživatel může zvolit jednu změnu ze seznamu.
6. Systém zobrazí uživateli vybranou změnu.

Dodatek: Filtry v kroku 1. zahrnují číslo změny (Change), krátký popis změny (Summary), kód zákazníka (Customer), stav změny (Status), naplánovaný začátek (Scheduled Start), naplánovaný konec (Scheduled End), identifikace vlastníka (Owner), jméno skupinu vlastníka (Owner Group), typ změny (Type), riziko změny (Risk), prioritu změny (Priority). V 2. kroku může uživatel zadat většinu položek textově, u plánovaného začátku a konce je zadáván den, měsíc a rok. Typ a stav změny může zvolit výběrem jedné z možných hodnot. Ve 3. kroku budou zobrazeny výsledky, které budou odpovídat všem zadaným položkám. Textové filtry budou vyhledávat podřetězce. Zobrazené položky pro vyhledávání jsou shodné s vyjmenovanými položkami filtrů.

Seznam položek zobrazené změny je uveden v kapitole 4.3.

PU03 – Upravit změnu

Primární aktér: Základní uživatel.

Akce pro spuštění:

Předpoklady: Uživatel je oprávněn upravovat tuto položku.

Hlavní tok:

1. Uživatel vyhledá a zobrazí změnu (<<include>> PU02 - Vyhledávání a prohlížení změn).
2. Uživatel u položky změny zvolí upravit, přidat nebo smazat.
3. Pokud se jedná o text.
 - 3.1. Systém umožní uživateli upravovat text a nabídne volbu potvrdit nebo zrušit úpravu.
 - 3.2. Uživatel upraví text a potvrdí.
 - 3.3. Systém uloží upravený text.
4. Pokud se jedná o položku výběru z pevně daného seznamu obsahujícího do 30 položek.
 - 4.1. Systém nabídne uživateli výběr z výpisu možných položek.
 - 4.2. Uživatel vybere a potvrdí.
 - 4.3. Systém nastaví položce zvolenou hodnotu.
5. Pokud se jedná o položku výběru z proměnlivého seznamu s běžně malým počtem položek.

- 5.1. Systém zobrazí data a umožní vyhledávání – nabídne zadání filtrů.
- 5.2. Pokud uživatel vyplní položky filtrů a potvrdí.
 - 5.2.1. Systém zobrazí upravená data dle zadaných filtrů.
- 5.3. Uživatel vybere položku.
- 5.4. Systém upraví původní položku podle zvolených dat.
6. Pokud se jedná o volnou položku a uživatel zvolil přidat.
 - 6.1. Systém přidá prázdnou položku (s výjimkou povinných hodnot).
7. Pokud se jedná o volnou položku a uživatel zvolil smazat.
 - 7.1. systém smaže danou položku.

Dodatek: Před každým uložením systém ověří oprávnění uživatele. V alternativním toku zobrazí upozornění uživateli, že na danou akci nemá oprávnění. Systém by také měl pokud možno, nezobrazovat uživateli možnost úpravy položky, pokud k ní není oprávněn. Oprávnění pro položky jsou popsána v kapitole 4.3.

PU04 – Odeslání k přijetí a kategorizaci

Primární aktér: Základní uživatel.

Akce pro spuštění: -

Předpoklady: Změna je ve stavu „uložená“.

Hlavní tok:

1. Uživatel upraví potřebné položky změny (<<include>> PU03 – Upravit změnu).
2. Systém ověří vyplnění povinných položek.
3. Pokud jsou po povinné položky vyplněny, systém změní stav změny na „přijetí a kategorizace“.
4. Pokud nejsou položky vyplněny, systém pokračuje alternativním tokem.

Alternativní tok:

1. Systém zobrazí upozornění na nevyplněné povinné položky.

Dodatek: Povinné položky pro tento případ užití by měly být vyznačeny při zobrazení změny. Povinné položky jsou: krátký popis (Summary), pravděpodobnost neúspěchu (Failure Probability), dopad (Impact), naléhavost (Urgency).

PU05 – Vyhledávání a prohlížení úkolů

Tento případ je obdobou případu vyhledávání a prohlížení změn s jedním rozdílem. Úkoly lze vyhledávat podle čísla úkolu (Task), vlastníka úkolu (Owner), plánovaného začátku a konce (Scheduled Start, Scheduled End), krátkého popisu (Summary), stavu (Status).

PU06 – Výpis přiřazených činností

Primární aktér: Základní uživatel.

Akce pro spuštění: Uživatel se přihlásí do systému nebo zobrazí hlavní stránku.

Hlavní tok:

1. Systém zobrazí seznamy:

1.1. Aktuální úkoly – seznam úkolů, které jsou uživateli přiřazeny a změna je ve stavu „realizace“.

1.2. Aktuální posudky – seznam posudků, které jsou uživateli přiřazeny a změna je ve stavu „posouzení“.

1.3. Seznam vlastních změn – seznam změn, které jsou otevřené a uživatel je jejich vlastníkem.

Dodatek: zobrazené položky budou obdobné jako u případů užití vyhledávání a zobrazení. Nebude pouze zobrazen vlastník a skupina vlastníka.

PU07 – Změnit stav úkolu

Primární aktér: Základní uživatel.

Akce pro spuštění:

Předpoklady: Změna, které je úkol součástí, není ve stavu zavřená. Realizátor nebo jeho skupina má přiřazen vybraný úkol.

Hlavní tok:

1. Uživatel vyhledá a zobrazí změnu (<<include>> PU02 - Vyhledávání a prohlížení změn) nebo vyhledá a zobrazí úkol (<<include>> PU05 - Vyhledávání a prohlížení úkolů).

2. Uživatel zvolí „změnit stav“ u některého z úkolů.

3. Uživatel změní položku podle toku případu „Upravit změnu“.

4. Pokud jsou všechny úkoly ve stavu „dokončeno“, změní stav změny také na „dokončená“.

PU08 – Autorizovat změnu

Primární aktér: Manažer změn.

Akce pro spuštění:

Předpoklady: Změna je ve stavu „autorizace“.

Hlavní tok:

1. Uživatel vyhledá a zobrazí změnu (<<include>> PU02 - Vyhledávání a prohlížení změn).

2. Uživatel zvolí „autorizovat“.

3. Systém změní stav změny na „implementace“.

PU09 – Příprava k posouzení

Primární aktér: Vlastník změny.

Akce pro spuštění:

Předpoklady: Uživatel je vlastníkem změny a změna je ve stavu „příprava posouzení“.

Hlavní tok:

1. Uživatel poskytne vlastní posudek (<<include>> PU03 - Upravit změnu).
2. Uživatel přidá potřebné posudky a přiřadí je (<<include>> PU03 - Upravit změnu).
3. Uživatel potvrdí přípravu k posouzení.
4. Pokud jsou vyplněné povinné položky, které jsou shodné s případem „PU04 – Odeslání k přijetí a kategorizaci“, systém změní stav změny na „technické posouzení“.

PU10 – Podání posudku

Primární aktér: Posuzovatel.

Akce pro spuštění:

Předpoklady: Uživatel má posudek přiřazený.

Hlavní tok:

1. Uživatel upraví svůj posudek ve změně (<<include>> PU03 - Upravit změnu).
2. Uživatel přidá potřebné posudky a přiřadí je (<<include>> PU03 - Upravit změnu).
3. Uživatel potvrdí přípravu k posouzení.
4. Pokud jsou všechny technické posudky vyplněny, změní systém stav změny na „business posouzení“.

Dodatek: Posudek zahrnuje kategorii (Category), dopad (Impact), závěr posudku v textové formě (Results), poznámky k implementaci (Implementation Notes), cenu (Cost), lidské zdroje (Effort). Uživatel může vyplnit libovolné položky. Upravovat je může i Základní uživatel. 4. krok však může provést pouze přiřazený posuzovatel. Systém by měl zobrazit posuzovatele jako vlastníka posudku (Owner) a/nebo skupinu posuzovatele (Owner Group).

PU11 – Vytvoření plánu změny

Primární aktér: Vlastník

Akce pro spuštění:

Předpoklady: Změna je ve stavu „plánování“.

Hlavní tok:

1. Uživatel určí čas začátku a konce změny (<<include>> PU03 - Upravit změnu).
2. Uživatel přidá potřebné úkoly a přiřadí je (<<include>> PU03 - Upravit změnu).
3. Uživatel potvrdí plán.
4. Pokud jsou vyplněné povinné položky, systém změní stav změny na „autorizace“.

Dodatek: Povinnými položkami je plánovaný start a konec změny (Scheduled Start, Scheduled End).

PU12 – Uzavření změny

Primární aktér: Základní uživatel

Akce pro spuštění:

Předpoklady: Změna není ve stavu „uzavřená“.

Hlavní tok:

1. Uživatel zvolí kód uzavření změny (<<include>> PU03 - Upravit změnu).
2. Systém změní stav změny na „uzavřená“.

Dodatek: Alternativně může uživatel v 2. kroku zvolit akci „zavřít změnu“.

PU13 – Správa členů skupiny

Primární aktér: Koordinátor změn týmu

Akce pro spuštění: Uživatel zvolí „Správa členů“

Předpoklady: Změna není ve stavu „uzavřená“.

Hlavní tok:

1. Systém zobrazí seznam všech členů skupin, ve kterých má uživatel roli „koordinátora“.
2. Systém uživateli umožní u každého uživatele přiřadit/odebrat role, jedná se o tyto možné role: „posuzovatel“, „realizátor“, „koordinátor“, „externí“. Uživatel nemusí mít přiřazenu žádnou roli.

PU14 – Správa vzorů

Primární aktér: Koordinátor změn týmu

Akce pro spuštění: Uživatel zvolí „Správa vzorů“

Předpoklady: -

Hlavní tok:

1. Systém zobrazí seznam všech vzorů skupin, ve kterých má uživatel roli „koordinátora“. Systém nabídne možnost vytvořit nový vzor, a u každé změny nabídne možnosti smazat vzor a validovat vzor.
2. Pokud uživatel zvolí nový vzor, systém vytvoří nový prázdný vzor.
3. Pokud uživatel zvolí smazat u některého vzoru, systém vybraný vzor smaže.
4. Pokud uživatel zvolí validovat vzor, je doba vypršení platnosti tohoto vzoru nastavena na 3 měsíce od aktuálního času a data.
5. Pokud uživatel zvolí upravit u některého vzoru, systém zobrazí vybraný vzor a umožní uživateli ho upravovat obdobně jako v (PU03 - Upravit změnu).

Dodatek: V 1. kroku jsou zobrazeny položky krátký popis (Summary), číslo vzoru (Template). V 5. kroku je umožněno měnit krátký popis (text), detailní popis (text), přidávat, mazat a měnit úkoly a posudky.

PU15 – Kategorizace a přijetí změny

Primární aktér: Manažer změn

Akce pro spuštění:

Předpoklady: -

Hlavní tok:

1. Uživatel určí typ změny (<<include>> PU03 - Upravit změnu).
2. Pokud uživatel zvolí „přijetí“ změny
 - 2.1. Pokud je typ změny vyplněn, systém změní stav změny podle modelu
 - 2.1.1. Urgentní – na stav „předběžný posudek“.
 - 2.1.2. Normální – na stav „předběžný posudek“.
 - 2.1.3. Standardní – na stav „implementace“.
3. Pokud uživatel zvolí „zamítnout“ změnu
 - 3.1. Systém změní stav změny na „zavřená“ s kódem zavření „zrušená“.

4.3 Další požadavky a dodatky

Položky dokumentu a oprávnění jejich úpravy

Na obsah dokumentu jsou kladeny přesné požadavky. Zde je uveden výčet požadovaných položek.

Položky jsou z hlediska práv rozděleny do několika skupin.

První skupina zahrnuje položky určující model, vlastníka, kategorii atd. Ty může Základní uživatel měnit až do „přijetí“ manažerem (tedy do stavu „přijetí a kategorizace“ včetně). Následně je může upravovat pouze manažer změn.

Druhá skupina zahrnuje položky popisující již změnu v podobě jak je schválená. Ty také může základní uživatel měnit, dokud není změna schválena (tedy do stavu „schválení“ včetně).

Třetí skupinou jsou položky, které lze měnit jen pomocí akcí v případech užití.

První skupinu tvoří:

- typ (Type) – výběr hodnoty z: normal, standard, emergency.
- kategorie (Category) – výběr hodnoty z: critical, major, medium, minor, BAU.
- čas povinného předstihu – nemá být měnitelný přímo, je určen dle kategorie (viz tabulka 3.2).
- vlastník a skupina vlastníka - elektronická adresa vlastníka, jeho jméno a jméno skupiny.

Druhou skupinu tvoří:

- Krátký popis (Summary) – text.
- Detailní popis (Details) – text.
- Popis testu před provedením změny (Pretest Details) – text.

- Dopad neprovedení změny (Effect of not implementing) – text.
- Ověřovací postup (Verification Plan) – text.
- Postup návratu (Remediation Plan) – text.
- Riziko (Risk) – výběr hodnoty z 1 až 5.
- Pravděpodobnost selhání (Failure Probability) – výběr hodnoty z 1 až 3.
- Dopad (Impact) – výběr hodnoty z 1 až 5.
- Naléhavost (Urgency) – výběr hodnoty z 1 až 5.
- Priorita (Priority) – výběr hodnoty z 1 až 4.

Neměnitelné položky:

- Číslo změny (Change Number) – automaticky přiřazené číslo.
- Autor (Created By) – uživatel, který změnu vytvořil.

Položky posudku (Assessment):

- Typ (Assessment Type) – výběr z několika technických a business typů.
- Dopad (Impact) – výběr hodnoty z 1 až 5.
- Výsledek (Result) – text.

Položky úkolu (Task):

- Číslo úkolu (Task Number).
- Popis úkolu (Description).
- Plánovaný začátek (Scheduled Start).
- Plánovaný konec (Scheduled End).
- Odhadovaná doba trvání (Estimated duration).
- Vlastník a skupina vlastníka (Owner, Owner Group).

Ostatní:

- Kód zavření – successful, canceled, closed, failed.
- Kód důvodu neúspěchu – výběr z několika možných chyb.
- Důvod neúspěchu – text.

Přílohy

System by měl umožnit vkládat soubory jako přílohy k jednotlivým změnám. Tento požadavek není podstatný pro provedení experimentu.

Časová pásma

Systém může být současně využíván ve více časových pásmech. Správné určení času včetně časového pásma je důležité, stejně tak je důležitá snadná orientace uživatelů. Systém by měl zobrazit v jakém časovém pásmu je čas uveden a také přizpůsobit zobrazení časů v systému aktuálnímu časovému pásmu konkrétního uživatele.

Tento požadavek není podstatný pro provedení experimentu.

Jazyk aplikace

Základním jazykem aplikace je angličtina, ale aplikace by měla otevřená snadným textovým i jazykovým úpravám. Do aplikace by mělo být možné vložit jiné znění všech textů, aplikace by měla být schopná udržovat více textových (jazykových) verzí, mezi kterými by mohl uživatel volit.

Tento požadavek není podstatný pro provedení experimentu.

Průvodce

Systém by měl nabídnout průvodce. Ten by měl dle stavu změny a role uživatele zobrazit adekvátní návod pro uživatele.

Logování všech činností uživatelů

Systém by měl zaznamenávat činnosti uživatelů. Z důvodů přehlednosti, by tak měl činit na více úrovních. První úrovní jsou činnosti procesu, druhou úrovní jsou všechny ostatní změny v dokumentu. Záznam by měl obsahovat čas, identifikaci uživatele, typ změny, původní a novou hodnotu.

4.4 Nefunkční požadavky

Tyto požadavky se týkají pouze testovaného nástroje. Vzhledem k účelu, kdy systém má sloužit jako podpora argumentů pro zlepšení, a ne jako samotný produkční systém, nejsou nefunkční požadavky na systém tak přísné na implementaci. Přesto je potřeba splnit níže uvedené požadavky pro dosažení jeho účelu.

Výkon

Systém by měl zvládnout souběžnou práci deseti uživatelů. Odpověď systému by měla být v rámci standardní webové aplikace.

Znovupoužitelnost

Uvedené funkce, myšlenky a části návrhu by měli být snadno použitelné ve strategickém nástroji firmy. Na zdrojový kód nástroje nejsou kladeny požadavky v rámci jejich opětovného použití.

Bezpečnost

Všechny operace systému budou zaznamenány v logu.

Autentizace a autorizace

Data systému jsou zobrazena pouze přihlášeným uživatelům. Přihlašovací údaje jsou ověřeny v lokální databázi aplikace pro případ experimentu bez produkčních dat.

Datové zdroje

Experiment nebyl schválen na produkčních datech, proto bude využito ručně vytvořených dat.

4.5 Akceptační kritéria

Kritéria, která musí být splněna, aby mohlo být dosaženo cílů projektu. Systém by měl implementovat všechny uvedené případy užití a požadavky, které jsou podstatné pro ověření přínosu navrhované změny. Jedná se o tyto případy užití:

1. Vytvoření nové změny.
2. Úprava změny.
3. Odeslání k přijetí a kategorizaci.
4. Vyhledávání a prohlížení změn.
5. Správa vzorů.

4.6 Návrh architektury

Architektura určuje celkovou strukturu a základní konstrukci části i celého systému. Architektura obvykle definuje rozdělení systému (aplikačních dat, procesů, datových toků) do logických celků, stanoví struktury těchto komponent a vzájemné vztahy a interakce mezi nimi. To systém zpřehledňuje, usnadňuje vývoj ve více lidech i testování. Tuto architekturu je potřeba zvolit v počátku vývoje, protože pozdější změny jsou velmi obtížné. Současně tato volba souvisí i s volbou jazyka, v mém případě frameworku, pomocí kterého bude systém implementován.

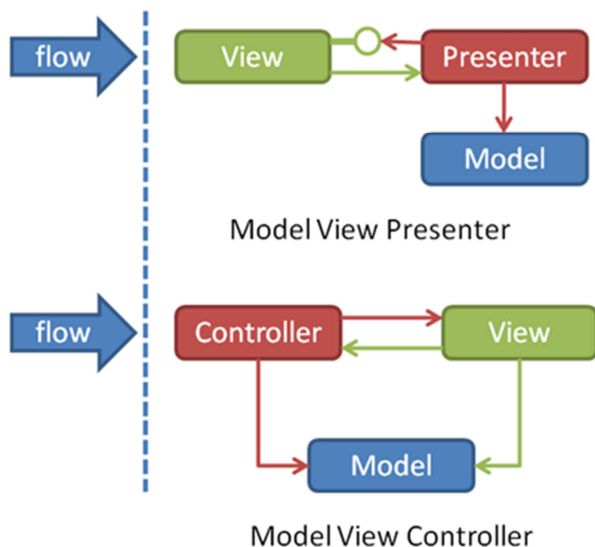
Z možných architektur jsem zvažoval architekturu MVC (Model-View-Controller) nebo MVP (Model-View-Presenter). Společně s výběrem frameworku Nette Framework jsem zvolil architekturu MVP.

4.6.1 Architektura Model-View-Presenter

Architektura MVP obsahuje tři vrstvy (komponenty):

- Model – zajišťuje přístup k datům a manipulaci s nimi.
- View – převádí data z modelu do vhodné podoby pro zobrazení uživateli.
- Presenter – reaguje na události od uživatele, stará se o změny v modelu nebo pohledu (View).

Dle autorů Nette Frameworku lze pro zjednodušení říci, že presenter v Nette je totéž, co controller v jiných frameworkcích [14]. Je tomu tak, nejspíš protože koncept MVC i MPV bývá v praxi realizován volnějším způsobem. Architekturu MVP i MVC zobrazuje obrázek 4.3. Hlavním rozdílem patrným i z uvedeného obrázku je, že pohled (view) nemusí o modelu vědět, komunikaci s modelem zajišťuje presenter [15].

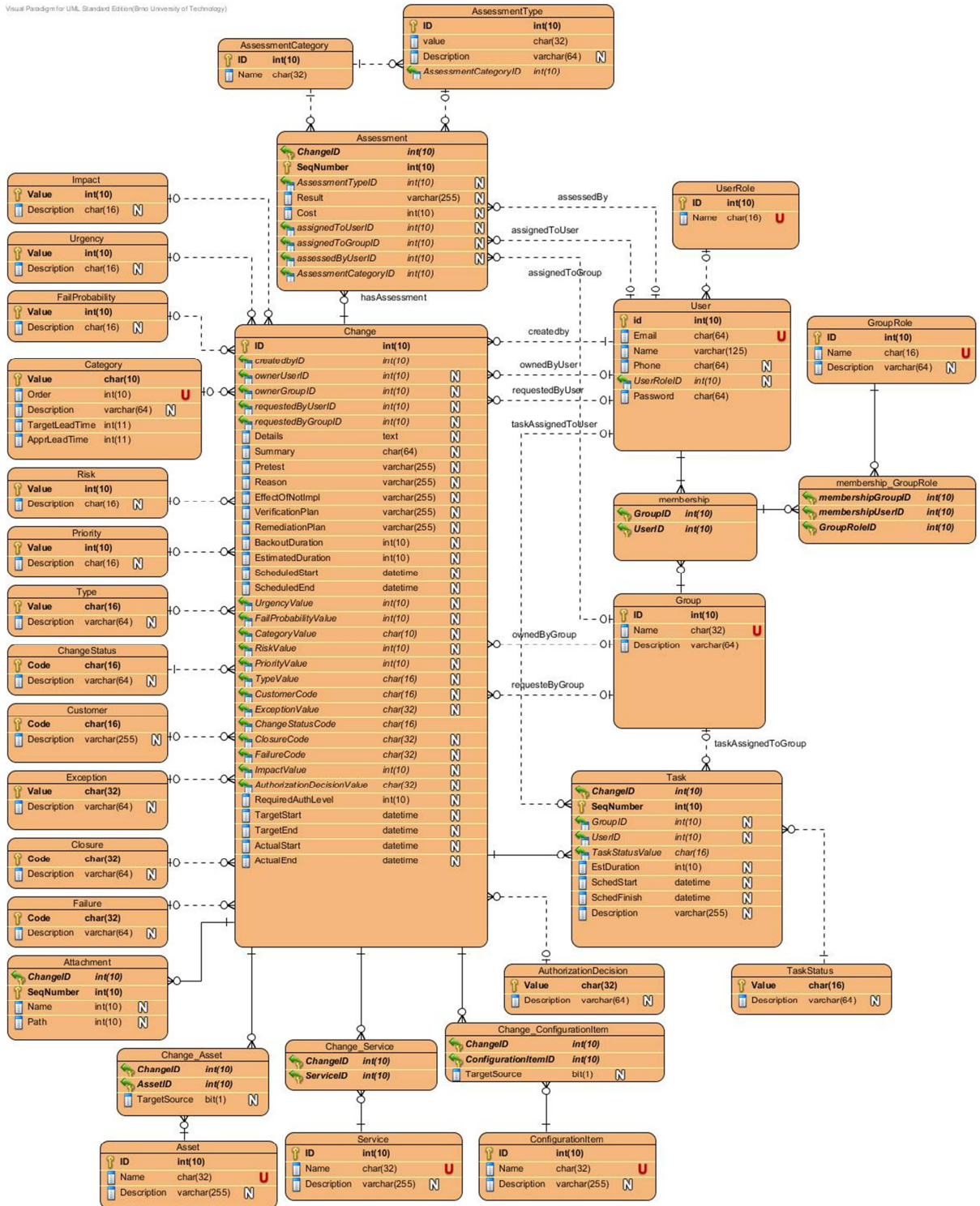


Obrázek 4.3: Rozdíly architektur MPV a MVC, zdroj: [15].

4.7 Návrh databáze

Návrh databáze popisuje, jak budou data aplikace uložena v databázi. Tento návrh má vliv na rychlost aplikace. U webové aplikace, jejíž podstatou je práce s daty, je návrh databáze opravdu důležitou částí.

Při návrhu databáze se staví na požadavcích na systém. Pro modelování návrhu databáze jsem využil ER diagramu, který se pro konceptuální datové modelování používá. ER diagramy nejsou možná vhodné pro složitější aplikace s komplexními daty, ale dle mého úsudku má zde navrhovaný systém stále spíše plochou jednoduchou strukturu (krátké záznamy s často pevnou délkou a atomickými hodnotami atributů). I zde jsem jako podpůrný modelovací nástroj použil nástroj Visual Paradigm for UML 10.0. Výsledný model databáze je zachycen na obrázku 4.4.



Obrázek 4.4: Návrh databáze aplikace, zdroj: autor

Základní entitou systému je změna (Change). Ta má pod sebou položky posudků (Assessment) a úkolů (Task) a přílohu (Attachment). Každá změna může mít přiřazeného uživatele (User) a skupinu (Group) ve vztazích vlastníka (owned by), autora (created by), žadatele (requested by). Každý úkol může mít také přiřazeného uživatele nebo skupinu, který ji bude provádět. Skupina zahrnuje několik uživatelů, ale i uživatel může být členem více skupin, to zachycuje entita členství (Membership).

Uživatel může mít ve skupině nějakou roli nebo více rolí (GrouRole). Podstatné jsou také entity služby (Service), aktiva (Asset) a konfigurační položky (ConfigurationItem), ty mohou být ke změně přiřazeny jako komponenty, které mají být změnou dotčeny. Ostatní entity zahrnují výčty „atributů“ změny, které lze u změny nastavit. Ty kromě samotné hodnoty obsahují i popis.

4.8 Návrh grafického rozhraní

V návrhu grafického rozhraní určujeme, jak bude prostředí aplikace vypadat a jak budou rozmístěny jednotlivé funkční prvky. Případně i jak budou jednotlivé stránky mezi sebou propojeny, aneb jak jimi bude uživatel moci procházet. Toto ovlivní intuitivnost a náročnost ovládání, přehlednost a dostupnost informací a funkcí.

Tento návrh by měl při nést také jistá zlepšení oproti stávajícímu nástroji. Při návrhu jsem vycházel ze stávajícího nástroje, obvyklého rozložení aplikací, určených požadavků (zvláště z případů užití) i předchozí analýzy procesu.

Základní rozložení stránky: V levém horním rohu bude název aplikace. V pravém horním rohu bude identifikace přihlášeného uživatele a ikona pro odhlášení ze systému. Na levé straně bude hlavní nabídka akcí (menu). Zvolená akce určí obsah aktivního okna. Na horní hraně aktivního okna bude nabídka akcí vztažených k dané činnosti.

Hlavní stránka po přihlášení nebo volbě „domů“ (Home) bude v aktivním okně obsahovat požadavky na uživatele – zprávy, úkoly, posudky, které má provést a vlastní změny obecně. Toto by mělo informovat uživatele po jeho přihlášení o všech činnostech, které jsou po něm požadovány. Tyto informace si také může zobrazit sám návratem na domovskou stránku.

Menu bude nabízet hlavní akce uživatelů – vyhledávání změn, vyhledávání úkolů, vyhledávání vzorů, vytvoření nové změny. V případě koordinátora změn i vytvoření vzoru.

Prohlížení změny je prováděno v rámci aktivního okna. Ze stávajícího nástroje jsem přebral rozložení položek změny do tabulátorů. Tohle bylo i podstatné, aby se uživatelé při „přechodu“ zorientovali. Důležitou změnou bylo umístění položek zavření změny, protože původní rozložení dělalo uživatelům potíže. Tabulátory jsou tyto: „General“, „Assessment“, „Schedule“, „Authorize“, „Review and Close“. V každém tabulátoru jsou zahrnuty položky blízké nějakému stavu změny a akce. Opakují se hlavní identifikační položky změny jako je její popis, číslo, dopad, pravděpodobnost neúspěchu a riziko.

Vyhledávání změny nebo úkolů je zobrazeno jako formulář pro zadání filtrů, podle kterých se bude vyhledávat, navázaných na tabulku, která bude zobrazovat výsledky.

5 Implementace

Tato kapitola popisuje několik bodů týkajících se implementace navrženého systému a je rozdělena do několika částí. Prvním krokem implementace je volba prostředků pro implementaci včetně jejich zajištění. Volba nástroje se prolíná s fází návrhu, konkrétně s návrhem architektury. První část kapitoly popisuje nástroje včetně důvodů, proč byly zvoleny. Druhá část popisuje princip řešení některých částí aplikace.

Určení požadavků, návrh i implementaci jsem provedl ve dvou iteracích. První implementace poskytla dobrý základ pro upřesnění požadavků. Na základě nich jsem provedl odpovídající změny v návrhu a upravil v implementaci do konečné podoby.

5.1 Použité programovací jazyky

Jedním z požadavků bylo, aby aplikace byla dostupná online. Tento požadavek sice neměl vysokou důležitost pro provedení experimentu, přesto jsem se rozhodl nástroj implementovat jako webovou aplikaci. To umožní i snadnější provádění experimentu v rámci firmy.

Pro implementaci webové aplikace existuje několik možností, co se týče logiky na straně serveru i obsluhy klientské části. Aktuální aplikace je v jazyce Java, avšak vzhledem k tomu, že aplikace měla být dodána v krátké době a s omezenými zdroji, kdy aplikace měla být implementována pouze mnou, zvolil jsem pro implementaci serverové části jazyk PHP⁹ a Nette Framework. Nette Framework poskytuje základní kostru pro aplikace v jazyce PHP, načítání tříd dle potřeb, vlastní vrstvu pro přístup k databázi, eliminaci několika bezpečnostních děr, ladící nástroje a několik dalších výhod.

Aby byla aplikace o něco blíže chování aktuálnímu nástroj, využil jsem pro obsluhu klientské strany jazyk Javascript¹⁰ a knihovnu jQuery¹¹.

Při implementaci jsem bral v úvahu to, že standardním nástrojem pro prohlížení webových stránek ve firmě je Firefox.

Pro uložení dat jsem použil databázový systém MySQL. Přístup k databázi je realizován s pomocí Nette\Database – vrstvy, která je součástí samotného Nette Framework.

Pro návrh rozhraní jsem použil jazyk XHTML¹² a CSS¹³.

⁹ PHP je skriptovací programovací jazyk určený především pro programování dynamických internetových stránek.

¹⁰ JavaScript je klientský skript (je vykonán na straně klienta), který se používá v internetových stránkách.

¹¹ jQuery je javascriptová knihovna, která usnadňuje vytváření interakce mezi JavaScriptem a HTML.

¹² XHTML je značkovací jazyk pro tvorbu hypertextových dokumentů v prostředí WWW vyvinutý W3C.

¹³ CSS je jazyk pro popis způsobu zobrazení stránek napsaných v jazycích HTML, XHTML nebo XML.

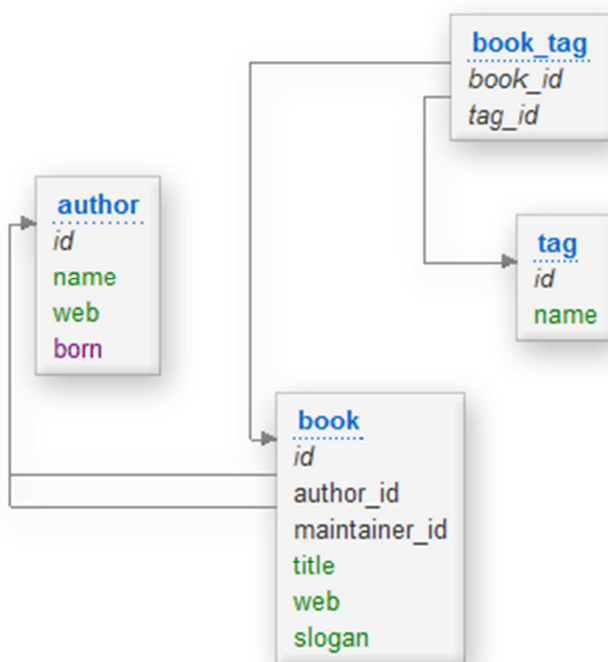
5.1.1 Nette Framework

Nette Framework byl vyvinutý českým vývojářem Davidem Grudlem. Framework je postaven na jazyce PHP a slouží pro tvorbu webových aplikací. Tento Framework využívá softwarovou architekturu Model-View-Presenter, která byla popsána v kapitole 4.7. Tento framework je nabízený pod licencí GNU GPL¹⁴ a BSD¹⁵ [14].

5.1.2 Nette\Database

Nette\Database je vrstva pro práci s databázemi. Jednou z jeho výhod, kterých jsem při vývoji využil, jsou informace zobrazené v panelu pro ladění. Tento panel se zobrazí v režimu ladění v Nette Framework. Při využití Nette\Database zobrazuje informace o vykonaných SQL dotazech, době jejich vyřízení, počtu vrácených řádků.

Další vlastností Nette\Database je vlastní nástroj pro vytváření SQL dotazů podle potřeb. Tento nástroj je založen na knihovně notORM¹⁶. Funkci tohoto nástroje naznačuje následující příklad převzatý z dokumentace k Nette Framework (viz obrázek 5.1) [14]:



Obrázek 5.1: Příklad databáze pro ukázkou funkce Nette\Database, zdroj: [14]

¹⁴ GNU General Public License (GNU GPL) je licence pro svobodný software.

¹⁵ Licence BSD licence pro svobodný software.

¹⁶ notORM je PHP knihovna pro snadný přístup k datům v databázi, <http://www.notorm.com/>

A zde je uveden kód, který by byl použit pro výpis knih (tabulka `book`) včetně údajů o autorovi (tabulka `author`):

```
foreach ($database->table('book')->order('title')->limit(5) as $book)
{
    echo $book->title, ' (' , $book->author->name, ')';
    // ekvivalentní s $book['title'], ' (' , $book['author']['name'], ')';
}
```

Snadné a efektivní použití tohoto nástroje v některých případech předpokládá dodržení konvence názvů tabulek, primárních a cizích klíčů. Cizí klíč by měl být tvořen spojením referenční tabulky a jeho klíče (například `author_id`, který je uveden předchozím příkladem). Tuto konvenci jsem ve své implementaci nemohl dodržet, protože z návrhu databáze mě plyne více referenčních klíčů mezi stejnými tabulkami a oba klíče nemohou mít stejný název.

5.1.3 jQuery

jQuery je framework napsaný v jazyce Javascript. Svými funkcemi ulehčuje vytváření skriptů v tomto jazyce. V průběhu implementace jsem využil hlavně funkcí pro vyhledávání elementů DOM a práce s nimi. Vyhledávají se pomocí CSS nebo Xpath¹⁷. Základem je volání funkce `jQuery`, parametry určují, co a jak tato funkce zpracuje.

5.2 Použité nástroje

Pro usnadnění implementace jsem využil při programování vývojové prostředí NetBeans IDE. Tento nástroj podporuje několik jazyků (PHP, JavaFX, C/C++, JavaScript atd.) a frameworků. Pro podporu Nette Framework existuje modul „NetBeans PHP Nette Framework“.

5.3 Implementace s Nette Framework

Při implementaci jsem dodržel adresářovou strukturu doporučenou Nette Framework. Hlavní části aplikace jsou tvořeny soubory v adresářích `app`, `libs` a `www`. Vlastní aplikační logika serveru je umístěna v adresáři `app`. V adresáři pro knihovny `libs` jsou uloženy znovupoužitelné komponenty. Jsou zde umístěny všechny komponenty Nette Framework. Oba tyto adresáře jsou z bezpečnostních důvodů nedostupné online. Jediný přístupný adresář je `www`. V něm je soubor `index.php`, který předává řízení do aplikace (tj. adresáře `app`) zaváděcímu souboru `bootstrap.php`.

Součástí Nette Framework je systémový DI kontejner. V něm se nachází všechny služby a parametry pro běh aplikace. Většina parametrů aplikace je uložena v souboru `config.neon`

¹⁷ XPath (XML Path Language) je počítačový jazyk, pomocí kterého lze adresovat části XML dokumentu.

v adresáři `app/config`. Při zpracování HTTP požadavku `bootstrap.php` předá požadavek s pomocí nadefinovaného přesměrování odpovídající části aplikace.

5.4 Implementace vybraných požadavků

V této kapitole popíšu způsob, jakým jsem implementoval některé základní funkce aplikace (specifikované případy užití).

5.4.1 Vyhledávání změn

Při implementaci vyhledávání změn jsem využil komponentu `Nextras\Datagrid` nabízenou pod licenci MIT. Tato komponenta spojuje klasické formuláře využitě pro zadání filtrů a tabulky pro zobrazení dat na straně uživatele s funkcí, která dané filtry zpracuje a komponentě poskytne data, která mají být zobrazena. Vzhled a vlastnosti položek v tabulce jsou dány šablonou, kde jsou definované.

Vlastní definovanou funkcí jsem tak určil zpracování filtrů a výběr dat. Tím jsem dosáhl požadovaného chování. Konkrétně při vyhledávání změn nejsou bez zadaných filtrů zobrazena žádná data, protože by docházelo k načítání velkého množství změn. Jiné chování jsem implementoval u zobrazení úkolů, posudků a při vyhledávání uživatelů.

5.4.2 Úpravy změny

Úprava změny je implementována s pomocí funkcí `jQuery`, které na straně klienta zobrazují editovatelné pole (pole pro zadání hodnoty uživatelem). Pokud tato hodnota byla změněna, je zaslána na server ve formátu `JSON`¹⁸. Server tuto zprávu zpracuje bez znovunačtení (invalidace) celé stránky.

Součástí zpracování je ověření oprávnění daného uživatele k provedení této změny.

¹⁸ JavaScript Object Notation (JSON) je datový formát, určený pro přenos dat

6 Experiment a výsledky

Tato kapitola blíže popisuje provedený experiment v rámci ověřování přínosu navržených změn.

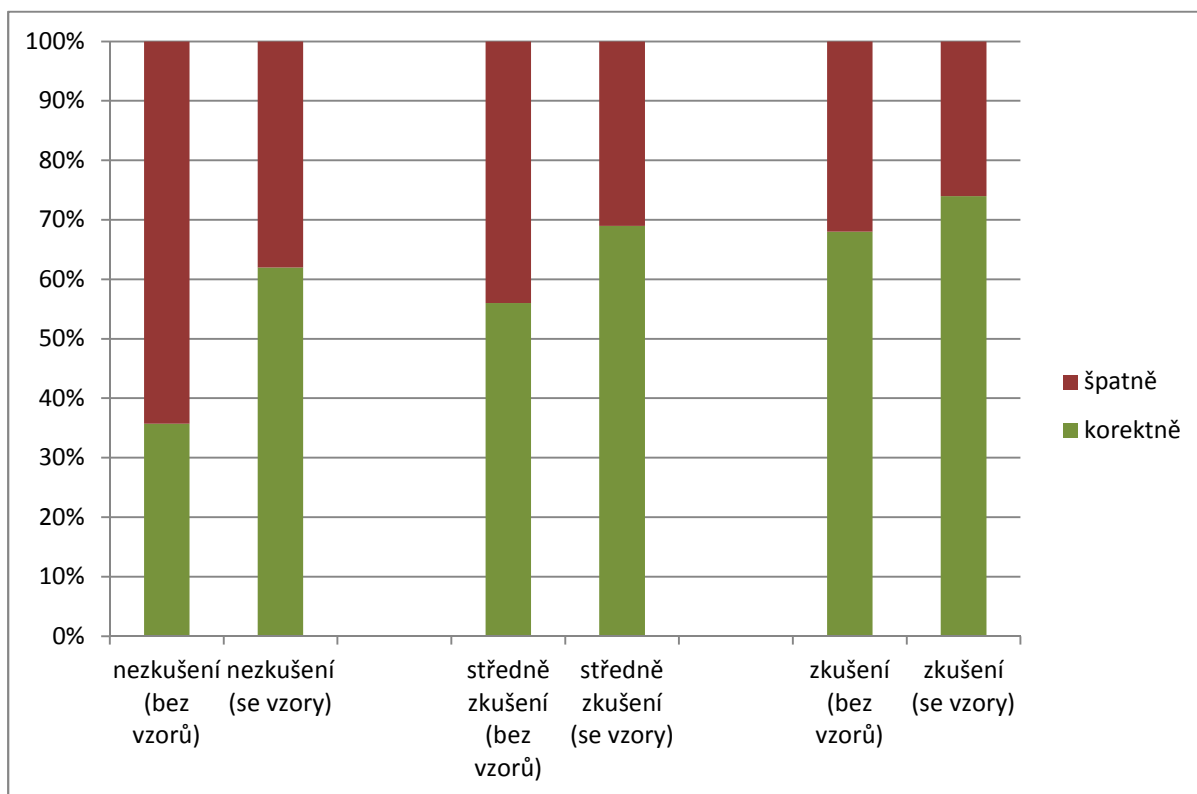
Nejpodstatnějším bodem zájmu zadavatele byla korektnost při vytváření změn, vytvoření požadavku na změnu a trvání této činnosti. V této souvislosti byl navržen a implementován upravený systém, který umožňoval využívání vzorů. Součástí tohoto řešení bylo i pravidelné ověřování platnosti vzorů, které mohlo být odpovědí na argumenty proti rizikům z neudržovaných vzorů (více v kapitole 3.4.2).

Zadáním tedy bylo srovnání činnosti vytváření změn v upravené variantě nástroje s funkcí využívání vzorů a stejné činnosti v původním nástroji. A následně i prezentace řešení včetně ověřování platnosti.

Při návrhu experimentu jsem uvažoval vliv zkušeností uživatelů s vytvářením změn i s původním nástrojem, kvality vytvořeného vzoru. Vzhledem k velkému vlivu zkušeností uživatelů jsem je plánoval rozdělit do tří skupin (nezkušený, středně zkušený a zkušený). Za vypovídající hodnotu o „míře zkušenosti“ považuji počet již vytvořených požadavků na změnu. Hranice jsem stanovil neformálně, opíral jsem se o zkušenosti z předchozí analýzy a rozhovorů s účastníky. Hranice jsem určil pro nezkušeného na „méně než 5“ a pro zkušeného na „více než 30“ vytvořených změn v původním nástroji. Tyto skupiny považuji za směrodatné. Střední skupina měla pro mne nejisté rozložení „míry zkušenosti“.

Pro experiment jsem získal podporu vedení dvou oddělení ve firmě. Celkem mi bylo poskytnuto 12 osob, každá na zhruba 2 hodiny času. Dle dotazů bylo 5 osob nezkušených, 4 zkušené a 3 středně zkušené. Vytvořil jsem 20 příkladů na vytváření požadavku na změnu z různých oblastí jejich oddělení tak, že jsem si vyžádal reálně provedené změny, vynechal z nich informace, které tyto týmy obvykle nemají, a vytvořil z nich textová zadání. Tato textová zadání pak byla střídavě vytvářena v obou nástrojích. Žadatelé se mohli svobodně ptát na doplňující informace. Změna pak byla kontrolována jedním z manažerů změn. Byl zaznamenáván čas trvání vytvoření úspěšné změny a počet chyb ve změně.

V naměřených hodnotách existovaly velké výkyvy u různých příkladů. V celkových číslech mělo použití vzorů pozitivní vliv na korektnost vytvoření požadavku i na trvání této činnosti. Bylo ušetřeno zhruba 0,5 hodiny času jednoho člověka na jednu změnu. Jedná se o průměrnou hodnotu ze všech 12 osob. Tabulka zachycující zlepšení v rámci procent korektně vytvořených změn z celkových změn zachycuje obrázek 6.1.



Obrázek 6.1: Korektně a špatně vytvořené požadavky na změnu

I zde došlo ke zlepšení, zvláště u nezkušených uživatelů.

Závěrem lze říci, že přínos této navržené změny byl potvrzen, otázkou je ale vypovídající hodnota tohoto experimentu o míře zlepšení.

Takto nastavený experiment není sice relevantní pro skutečné dokazování, byl ale dostatečný pro cíle v rámci získání další podpory. Projekt pokračuje nad rámec diplomové práce a v plánu je zajištění podpory pro „rozhraní“ do aktuálního nástroje a provedení dlouhodobějšího sběru metrik., avšak toto vzhledem ke komplikacím s podporou některých zainteresovaných stran ve firmě nemohlo být zahrnuto včas do této práce. Přesto tyto výsledky ukazují na „správný směr“.

7 Závěr

Diplomová práce se zabývá managementem změn, analýzou požadavků a návrhem a implementací technické podpory pro management změn v reálné společnosti. Druhým důležitým tématem práce je inovace, v rámci níž je tento nástroj vyvíjen.

Základem práce bylo seznámení se s managementem IT služeb, jehož součástí je jak management změn, tak zmíněná inovace. ITIL nabízí osvědčené praktiky, které svým rozsahem pokrývají prakticky celý management IT služeb. V práci jsem také vystavěl podklady pro určení postupu i přístupu k analýze problému a analýze požadavků. Tyto podklady jsem hledal procesu nestálého zlepšování v knize ITIL, metodologii Lean Six Sigma i doporučení dalších stran zabývajících se managementem IT služeb.

Na základě těchto znalostí jsem provedl analýzu problému – sběr a analýzu požadavků. Požadavky jsem analyzoval v kontextu zavedených procesů a cílů reálné společnosti. Postupoval jsem tak abych neopomenul žádnou z oblastí managementu IT služeb – lidí, procesů a nástroje.

Výstupem první analýzy byly návrhy na změny, které mají přinést zlepšení v činnostech procesů managementu změn v dané společnosti. Celkem bylo navrženo 5 změn. Zadavatelem byl vybrán jeden z návrhů pro ověření experimentem a to „rozšíření o systém vzorů“. Experiment měl být proveden pomocí aplikace vytvořené pro tento účel. V rámci analýzy požadavků na nástroj jsem tedy zvolil ty požadavky, které byly podstatné pro tento experiment nebo by případně mohly sloužit jako prezentace navržených změn.

Ve specifikaci požadavku jsem uvedl více požadavků, s kterými se může počítat při pokračování a dalšího rozvoje této práce. Uvedl jsem také, které z těchto požadavků jsou relevantní pro plánovaný experiment.

Zachycené požadavky byly pak zpracovány do specifikace požadavků a na základě nich vytvořen návrh aplikace. Podle této části specifikace požadavků a návrhu jsem následně implementoval aplikaci. Tuto aplikaci jsem využil pro provedení požadovaného experimentu k ověření přínosu navržené změny.

Vzhledem k omezené podpoře v projektu měl tento experiment omezené možnosti a tak je jeho vypovídající hodnota omezena. Přesto byl přínosem, protože zajistil získání další podpory pro pokračování projektu. V plánu je zajištění podpory pro vytvoření „rozhraní“ do aktuálního nástroje, které umožní provedení dlouhodobějšího sběru metrik a ověření i ostatních navržených změn. Lze říci, že tyto výsledky poukazují na „správný směr“.

Literatura

- [1] ITSM – IT Service Management Information General Information [online]. [cit. 2013-01-05]. Dostupné na URL: <<http://www.itsm.info/ITSM.htm>>
- [2] ITIL – výkladový slovník a zkratky v češtině [online]. 2011 [cit 2013-05-22]. Dostupné na URL: <http://www.ital-officialsite.com/InternationalActivities/ITILGlossaries_2.aspx>
- [3] SKÁLA, J.: Odborné informace o ITIL® best practice přístupu k řízení IT služeb. - itsmportal.cz [online]. [cit. 2013-01-29]. Dostupné na URL: <<http://www.itsmportal.cz/cs/-ITSM-ITIL-.alej>>
- [4] GÁLA, L., POUR, J., TOMAN, P.: *Podniková informatika*. Praha: Grada, 2006. ISBN 80-247-1278-4.
- [5] VOŘÍŠEK, Jiří, a kolektiv. *Principy a modely řízení podnikové informatiky*. Praha: Oeconomica, 2008. ISBN 978-80-245-1440-6.
- [6] OGC. *Service Transition*. 1. vyd. Norwich: TSO, 2007. 262s. ISBN 9780113310487
- [7] OGC. *Continual Service Improvement*. Norwich: TSO, 2007. 221s. ISBN 9780113310494.
- [8] itilservice.cz O ITIL [online]. [cit. 2013-02-20]. Dostupné na URL: <<http://www.itservice.cz/neco-o-me/o-ital/>>
- [9] OGC. *ITIL® Service Transition*, 2.vyd. London: TSO, 2011, ISBN. 9780113313068
- [10] SVOZILOVÁ, A.: *Zlepšování podnikových procesů*. Praha: Grada, 2011. ISBN 978-80-247-3938-0.
- [11] PANDE, P. S., COVANAGH, R. R., NEUMAN, R. P.: *Zavádíme metodu Six Sigma: aneb jakým způsobem dosahují renomované světové společnosti špičkové výkonnosti*. 1. vyd. Brno: TwinsCom, 2002, 416s. ISBN 80-238-9289-4.
- [12] IBM Tivoli Unified Process [svazek WWW stránek]. Verze 7.1.4. IBM Corporation. 2008. Svázán s produktem Tivoli Unified Process Composer, informace na URL: <<http://www-01.ibm.com/software/Tivoli/products/unified-process-composer.>>
- [13] NEUDSTADT, I., ARLOW, J.: *UML 2 a unifikovaný proces vývoje aplikací*. 2. vyd. Brno, 2007, 568s., ISBN 978-80-251-1503-9.
- [14] Dokumentace | Nette Framework [online]. 2013 [cit. 2013-04-11]. Dostupné na URL: <<http://doc.nette.org/cs/>>
- [15] Model View Presenter (MVP) VS Model View Controller (MVC) [online]. [cit. 2013-04-15]. Dostupné na URL: <<http://blog.vuscode.com/malovicn/archive/2007/12/18/model-view-presenter-mvp-vs-model-view-controller-mvc.aspx>>

Seznam příloh

Příloha 1. ITIL – výkladový slovník

Příloha 2. CD, obsahující zdrojové kódy a databázový skript

Příloha 1. ITIL – výkladový slovník

Term	Výraz v češtině	Definice v češtině
activity	činnost	Množina akcí navržena tak, aby se dosáhlo určitých výsledků. Činnosti jsou obvykle definovány jako část procesů nebo plánů a jsou dokumentovány v postupech.
assessment	posouzení	Inspekce a analýza sloužící ke kontrole toho, zda jsou dodržovány normy / standardy a směrnice, zda jsou záznamy přesné a zda jsou dosahovány cíle hospodárnosti a efektivity. Viz také audit.
asset	aktivum	Zdroj nebo způsobilost. Aktiva poskytovatele služby zahrnují vše, co může přispět k dodávce služby. Typy aktiv mohou být následující: management, organizace, proces, znalost, lidé, informace, aplikace, infrastruktura nebo finanční kapitál. Viz také aktiva zákazníka; aktiva služby; strategická aktiva.
best practice	nejlepší praktiky	Osvědčené činnosti nebo procesy, které byly úspěšně použity několika organizacemi. ITIL je příkladem nejlepších praktik.
business	business / obchodní činnost	Korporátní entita nebo organizace, která se skládá z více podnikových jednotek. V kontextu ITSM výraz business zahrnuje veřejný sektor a neziskové organizace stejně jako firmy. Poskytovatel služby IT poskytuje službu IT zákazníkovi v rámci businessu. Provozovatel služby IT může být součástí stejné firmy jako jeho zákazník (interní poskytovatel služby) anebo částí jiné firmy (externí poskytovatel služby).
change	změna	Přidání, modifikace nebo odstranění čehokoliv, co by mohlo mít vliv na služby IT. Rozsah by měl zahrnovat změny veškerých architektur, procesů, nástrojů, metrik a dokumentace, a rovněž tak změny služeb IT a dalších konfiguračních položek.
change advisory board (CAB)	poradní výbor pro změny (CAB)	(ITIL Service Transition) Skupina lidí, která podporuje posouzení, prioritizaci, schválení a plánování změn. Poradní výbor pro změny je obvykle tvořen

		reprezentanty všech oblastí poskytovatele služeb IT, reprezentanty businessu a třetích stran, např. dodavatelů.
change management	správa změn	Proces, odpovědný za řízení životního cyklu všech změn, umožňující realizaci prospěšných změn při minimálním narušení služeb IT.
change model	změnový model	Opakovatelný způsob zacházení s určitou kategorií změny. Změnový model předepisuje specifické dohodnuté kroky, které se budou provádět u změn této kategorie. Změnový model může být velmi složitý, s mnoha kroky vyžadujícími schválení (např. větší release softwaru), nebo může být velice jednoduchý bez požadavků na schválení (např. znovunastavení hesla). Viz také poradní výbor pro změny; standardní změna.
change record	záznam o změně	Záznam obsahující detaily o změně. Každý záznam o změně dokumentuje životní cyklus jednotlivé změny. Záznam o změně se vytváří pro každý požadavek na změnu, který je přijat, i když může být následně zamítnut. Záznam o změně by se měl odkazovat na konfigurační položky, které jsou změnou dotčeny. Záznamy o změnách mohou být uloženy v systému správy konfigurací, nebo jinde v systému správy znalostí o službách.
COBIT	COBIT	(ITIL Continual Service Improvement) Cíle řízení v informačních a souvisejících technologiích (Control Objectives for Information and related Technology - COBIT) poskytují návody a nejlepší praktiky pro řízení procesů IT. COBIT je vydán organizací ISACA ve spojení s IT Governance Institute (ITGI). Pro více informací viz www.isaca.org .
configuration item (CI)	Konfigurační položka (KP)	Jakákoliv komponenta nebo jiné aktivum služby, které by měly být spravovány za účelem dodávky služby IT. Informace o všech konfiguračních položkách jsou zaznamenány v konfiguračním záznamu v systému správy konfigurací (CMS) a jsou udržovány během

		jejich životního cyklu správou aktiv služeb a konfigurací. Konfigurační položky jsou řízeny správou změn. Typicky zahrnují služby IT, hardware, software, stavby, lidi a formální dokumentaci, jako je dokumentace procesů a SLA.
continual service improvement (CSI)	neustálé zlepšování služeb (CSI)	Etapa životního cyklu služby. Neustálé zlepšování služeb zajišťuje, aby služby odpovídaly měnícím se potřebám businessu, a to tak, že se identifikují a implementují zlepšení služeb IT, která podporují podnikové procesy. Výkonnost poskytovatele služeb IT se průběžně měří a realizují se zlepšení procesů, služeb IT a infrastruktury IT za účelem zvýšení hospodárnosti, efektivity a nákladové efektivity. Neustálé zlepšování služeb je zlepšovacím procesem v sedmi krocích. I když tento proces je přiřazen neustálému zlepšování služeb, většina procesů má činnosti, které se realizují ve více fázích životního cyklu služby. Viz také Demingův cyklus, „plánuj – dělej – kontroluj – jednej“.
critical success factor (CSF)	rozhodující faktor úspěchu (CSF)	Něco, co se musí stát, aby služba IT, proces, projekt, plán nebo další aktivita dosáhly úspěchu. Dosažení tohoto faktoru se měří klíčovými ukazateli výkonnosti (KPI). Např. rozhodující faktor úspěchu „ochrana IT služeb při provádění změn“ je měřitelný takovými klíčovými ukazateli výkonnosti jako jsou „procentuální redukce neúspěšných změn“, „procentuální redukce změn způsobujících incident“, atd.
emergency change	naléhavá změna	Změna, která musí být implementována co nejdříve - např. vyřešení závažného incidentu nebo implementace bezpečnostní záplaty. Proces správy změn obvykle obsahuje specifický postup pro vykonání naléhavé změny. Viz také poradní výbor pro naléhavé změny.
impact	dopad	Míra účinku incidentu, problému nebo změny na podnikové procesy. Dopad je často založen na tom, jaký vliv bude mít na úroveň služby. Dopad a naléhavost jsou užívány pro přiřazení priority.

IT service management (ITSM)	správa služeb IT	Implementace a správa kvality služeb IT, které splňují potřeby businessu. Správa služeb IT je vykonávána poskytovateli služeb IT za využití vhodné kombinace lidí, procesů a informačních technologií. Viz také správa služeb.
metric	metrika	To, co je měřeno a vykazováno za účelem řízení procesu, služby IT nebo činnosti. Viz také klíčový ukazatel výkonnosti (KPI).
model	model	Reprezentace systému, procesu, služby IT, konfigurační položky atd., které se používá k pochopení nebo předpovědi budoucího chování.
normal change	normální změna	Změna, která není naléhavá ani standardní. Normální změny se uskutečňují v definovaných procesních krocích procesu správy změn.
outsourcing	outsourcing	Využívání externích dodavatelů služeb ke správě služeb IT.
proces owner	vlastník procesu	Osoba, která je odpovědná za zajištění toho, aby proces odpovídal danému účelu. Odpovědnostmi vlastníka procesu jsou sponzorování, návrh a řízení změn procesu a neustálé zlepšování procesu a jeho metrik. Role může být přiřazena též osobě, která zastává roli procesního manažera, avšak tyto role mohou být u větších organizací odděleny.
Request for change	Požadavek na změnu	Formální návrh na provedení změny. Obsahuje detaily navrhované změny a může být zaznamenán papírově nebo elektronicky. Pojem je často nesprávně používán ve smyslu záznamu o změně nebo ve smyslu změny samotné.
review	revize	Vyhodnocení změny, problému, procesu, projektu atd. Revize jsou typicky prováděny v určitých bodech životního cyklu, obzvláště po uzavření. Účelem revize je zajistit, aby byly poskytnuty všechny dodávky, a identifikovat příležitosti pro zlepšení.
standard change	standardní změna	Předem schválená změna představující nízké riziko, relativně obvyklá, vykonávaná podle určitého postupu nebo pracovních instrukcí. Např. opětovné nastavení

		hesla nebo poskytnutí standardního vybavení novému zaměstnanci. Implementace standardní změny nevyžaduje požadavek na změnu (RFC), k zaznamenání a sledování se používají jiné mechanismy, jako požadavek na službu. Viz také změnový model
trend analysis	analýza trendů	Analýza dat za účelem identifikace vzorků v čase. Je využívána ve správě problémů pro identifikaci obvyklých poruch nebo konfiguračních položek náchylných k poruchám, ve správě kapacit je používána jako modelovací nástroj k předvídání budoucího chování. Využívá se rovněž jako nástroj managementu k identifikaci nedostatků v procesech správy služeb IT.

Zdroj: ITIL – výkladový slovník a zkratky v češtině [2], 2011, URL: http://www.itil-officialsite.com/InternationalActivities/ITILGlossaries_2.aspx

Zkratka	Význam v anglickém jazyce	Význam v českém jazyce
CSI	continual service improvement	neustálé zlepšování služeb
CI	configuration item	konfigurační položka (KP)
ITSM	IT service management	správa služeb IT
KPI	key performance indicator	klíčový ukazatel výkonnosti
PMBOK	Project Management Body of Knowledge	Project Management Body of Knowledge
PDCA	Plan-Do-Check-Act	plánuj-dělej-kontroluj-jednej
PMI	Project Management Institute	Project Management Institute
SMART	specific, measurable, achievable, relevant and time-bound	specifické, měřitelné, dosažitelné, relevantní a časově závazné

Zdroj: ITIL – výkladový slovník a zkratky v češtině [2], 2011, URL: http://www.itil-officialsite.com/InternationalActivities/ITILGlossaries_2.aspx