

Česká zemědělská univerzita v Praze

Provozně ekonomická fakulta

Katedra informačních technologií



Diplomová práce

**Optimalizace a implementace procesu v systému
ServiceNow ve zvoleném podniku**

Bc. Tereza Katzerová

© 2019 ČZU v Praze

ZADÁNÍ DIPLOMOVÉ PRÁCE

Bc. Tereza Katzerová

Informatika

Název práce

Optimalizace a implementace procesu v systému ServiceNow ve zvoleném podniku

Název anglicky

Process optimization and implementation in ServiceNow system in a selected company

Cíle práce

Hlavním cílem diplomové práce je optimalizovat současné řešení ve zvoleném podniku v software ServiceNow. Na základě provedené analýzy dle požadavků zákazníka provést optimalizaci řešení. Optimalizované řešení bude navrženo k implementaci. Dílčím cílem diplomové práce je optimalizovat firemní řešení v systému ServiceNow a navrhnout obecně vhodnou variantu řešení pro střední a velké firmy.

Metodika

Teoretická část diplomové práce bude založena na sběru, studiu a interpretaci odborné a vědecké literatury. V praktické části budou využity poznatky získané studiem sekundárních zdrojů a budou využity návrhy zákazníků. Na jejich základě bude vytvořen návrh optimálního řešení, vhodného pro implementaci ve zvoleném podniku a ve středních a velkých podnicích.

Doporučený rozsah práce

60 stran

Klíčová slova

ITSM, ITIL, ServiceNow, implementace, automatizace, optimalizace, software

Doporučené zdroje informací

BUCKSTEEG, Martin. ITIL 2011. Brno: Computer Press, 2012. ISBN 978-80-251-3732-1.

DOHNAL, J. – POUR, J. *IT v řízení podniku : MBI*. Praha: Professional Publishing, 2016. ISBN 978-80-7431-160-4.

OŠKRDAL, Václav a Petr DOUCEK. *Praktické řízení ICT projektů*. Praha: Oeconomica, nakladatelství VŠE, 2014. ISBN 978-80-245-2073-5. citace PRO

ŠILEROVÁ, Edita a Klára HENNYEYOVÁ. *Informační systémy v podnikové praxi*. Druhé vydání. Praha: Powerprint, 2017. ISBN 978-80-7568-065-5.

VELTE, Anthony T, Toby J VELTE a Robert C ELSENPETER. *Cloud Computing: praktický průvodce*. Přeložil Jakub GONER. Brno: Computer Press, 2011. ISBN 978-80-251-3333-0.

Předběžný termín obhajoby

2018/19 LS – PEF

Vedoucí práce

Ing. Edita Šilerová, Ph.D.

Garantující pracoviště

Katedra informačních technologií

Elektronicky schváleno dne 15. 2. 2019

Ing. Jiří Vaněk, Ph.D.

Vedoucí katedry

Elektronicky schváleno dne 20. 2. 2019

Ing. Martin Pelikán, Ph.D.

Děkan

V Praze dne 10. 03. 2019

Čestné prohlášení

Prohlašuji, že svou diplomovou práci "Optimalizace a implementace systému ServiceNow ve zvoleném podniku" jsem vypracovala samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou citovány v práci a uvedeny v seznamu použitých zdrojů na konci práce. Jako autorka uvedené diplomové práce dále prohlašuji, že jsem v souvislosti s jejím vytvořením neporušil autorská práva třetích osob.

V Praze dne 29.3.2019

Poděkování

Ráda bych touto cestou poděkovala Ing. Editě Šilerové, Ph.D. za odborné vedení a cenné rady při tvorbě diplomové práce. Dále bych ráda poděkovala kolegům, kteří se společně se mnou podíleli na projektu, na jehož základě byla diplomová práce vytvořena. V neposlední řadě mé poděkování patří rodině a přátelům, kteří mě v průběhu celého studia podporovali.

Optimalizace a implementace procesu v systému ServiceNow ve zvoleném podniku

Abstrakt

Podstatou diplomové práce je optimalizace stávajícího řešení zvolené společnosti a jeho následná implementace do nové prostředí systému ServiceNow. Zvolená společnost využívá systém ServiceNow mimo jiné pro tvorbu a správu požadavků na síťové připojení. Práce je rozvržena na teoretickou a praktickou část.

V teoretické části jsou vymezeny základní pojmy související s informačními systémy přiblížena problematika Cloud computingu, a metodika ITIL, která popisuje nejlepší metody pro správu IT služeb. Stěžejní část teorie tvoří kapitola o systému ServiceNow, možnostech jeho konfigurace a popsání nejdůležitějších best practices, kterými se později řídím v praktické části práce.

Úvodem do praktické části práce je představení zainteresovaných společností a popsání procesu Network Service Request, který je předmětem optimalizace a implementace. Následuje technická analýza současného řešení, která zkoumá míru chybovosti, včetně určení kritičnosti a navržení doporučení pro jednotlivá zjištění. Na základě této analýzy jsou navrženy scénáře pro optimalizované řešení, odhadnuty finanční náklady a shromážděny požadavky zákazníka. Stěžejní částí práce je samotná implementace optimalizovaného řešení, která, tak jak je v práci popsána, je mou vlastní prací a požadovaným cílem. Na závěr je vyhodnocena skutečná finanční náročnost a porovnání s uvedeným odhadem.

Klíčová slova: ServiceNow, ITIL, Cloud computing, informační systém, implementace, optimalizace, software, ITSM

Process optimization and implementation in ServiceNow system in a selected company

Abstract

The purpose of this diploma thesis is to optimize the current solution of the selected organization and the following implementation into a new environment of the ServiceNow system. The chosen company uses the ServiceNow system for many purposes, but in this case for creating and managing the requests for the network services. The thesis is split into two parts, the theoretical and the practical one.

In the theoretical part, there are stated the basic concepts related to information systems, as well as the ITIL methodology describing the best practices for managing IT services. There is also a chapter about Cloud Computing. The main part of the theoretical part is a chapter about ServiceNow system, what are the main ways of its configuration and there are also described the best practices relevant to the practical part of the thesis.

The practical part starts with the introduction of the involved companies and description of the Network Service Request process, which is the subject of optimization and implementation. Following chapter is dedicated to the technical analysis of the current solution, which examines the error rate, criticality of each error and proposed recommendations. Based on this analysis, there are two scenarios for the optimized solution proposed, the financial costs for both scenarios were estimated and also the customer's requirements were gathered. The main part of the practical part is the implementation of the optimized solution based, which was my own work and the required target. At the end of the thesis, there is the financial evaluation of the actual costs and comparison with the estimation.

Keywords: ServiceNow, ITIL, Cloud Computing, information system, implementation, optimization, software, ITSM

Obsah

1. Úvod	12
2. Cíl práce a metodika	13
3. Teoretická východiska	14
3.1. Základní pojmy	14
3.1.1. Informace a data	14
3.1.2. Systém	14
3.2. Informační systém	15
3.2.1. Členění informačních systémů	15
3.2.2. Ekonomika informačních systémů	18
3.3. ITIL	19
3.3.1. Strategie služeb	20
3.3.2. Návrh služby	22
3.3.3. Přechod služby	23
3.3.4. Provoz služby	24
3.3.5. Neustálé zlepšování služby	26
3.4. Cloud Computing	27
3.4.1. Typy cloud computingu	27
3.5. ServiceNow	31
3.5.1. Konfigurace ServiceNow	32
3.5.2. ServiceNow Best Practices	38
4. Praktická část	40
4.1. Představení společností	40
4.2. Network Service Request proces	41
4.3. Technická analýza stávajícího řešení	42
4.3.1. Předmět a cíle analýzy	42
4.3.2. Výstupy analýzy	42
4.3.3. Shrnutí analýzy	56
4.4. Návrhy řešení	58
4.4.1. Scénář A – Převod řešení do nové instance, oprava chyb a vylepšení	58
4.4.2. Scénář B – Nové řešení “na zelené louce”	59

4.5.	Finanční odhady pro jednotlivé scénáře	60
4.5.1.	Scénář A	60
4.5.2.	Scénář B	61
4.6.	Požadavky zákazníka	62
4.6.1.	Katalogové položky	62
4.6.2.	Workflow	62
4.6.3.	Možnost zrušení požadavku	63
4.6.4.	Schvalovací proces	63
4.6.5.	Předpokládané doručení	64
4.6.6.	Průzkum spokojenosti	65
4.7.	Implementace optimalizovaného řešení	65
4.7.1.	Katalogové položky	65
4.7.4.	Requested Item	70
4.7.5.	Workflow	72
4.7.6.	Zrušení požadavku	76
4.7.7.	Předpokládané doručení	78
4.7.8.	Průzkum spokojenosti	78
5.	Vyhodnocení finančních nákladů projektu	79
6.	Závěr	81
7.	Seznam použitých zdrojů	82
8.	Přílohy	84

Seznam obrázků

Obrázek 1: Životní cyklus podle ITIL knih a jejich procesů.....	20
Obrázek 2: Model SaaS	28
Obrázek 3: Model PaaS	29
Obrázek 4: Model HaaS.....	31
Obrázek 5: Logo společnosti ServiceNow	32
Obrázek 6: Procesní tok Business rule.....	34
Obrázek 7: Formulářové UI Actions.....	36
Obrázek 8: Seznamové UI Actions	36
Obrázek 9: Tvorba Script Include	38
Obrázek 10: Volání funkce ze Script Include.....	38
Obrázek 11: Skript znázorňující pevně nakódované hodnoty	45
Obrázek 12: Příklad velkého množství nastavených UI Policies.....	54
Obrázek 13: Příklad nepřehledného workflow.....	55
Obrázek 14: Výsledný katalogový formulář pro Network Infrastructure Request	67
Obrázek 15: Výsledný katalogový formulář pro Managed Service Request	70
Obrázek 16: Formulář požadavku pro Network Infrastructure Request.....	71
Obrázek 17: Formulář požadavku pro Managed Service Request.....	72
Obrázek 18: Tlačítko pro zrušení požadavku	77
Obrázek 19: Okno pro zrušení požadavku.....	77
Obrázek 20: Stornované workflow v případě zrušení požadavku.....	77
Obrázek 21: Client Script pro pole Estimated Delivery	78
Obrázek 22: Průzkum spokojenosti	79
Obrázek 23: Graf výsledků dotazníkového průzkumu spokojenosti	79

Seznam tabulek

Tabulka 1: Oblasti dopadu nalezených chyb	43
Tabulka 2: Celkové skóre	57
Tabulka 3: Celkový přehled kritičnosti.....	57
Tabulka 4: Finanční náklady na scénář A.....	61
Tabulka 5: Finanční náklady na scénář B	61
Tabulka 6: Úrovně schvalování.....	64
Tabulka 7: Kombinace polí a jejich akce.....	69
Tabulka 8: Dodatečné náklady na vývoj.....	80

Seznam příloh

Příloha 1: Workflow pro Internet Connection (Společnost A).....	84
Příloha 2: Workflow pro WAN Request (Společnost A)	85
Příloha 3: Workflow pro Managed Service Request (Společnost A)	86
Příloha 4: Workflow pro LAN / WLAN Request (Společnost A)	87
Příloha 5: Návrh nového workflow pro LAN / WLAN Service Request = Network Infrastructure Request	88
Příloha 6: Návrh nového workflow pro Managed Service Request.....	89
Příloha 7: Implementované workflow pro Network Infrastructure Request (Společnost E)	90
Příloha 8: Implementované workflow pro Managed Service Request (Společnost E).....	91

1. Úvod

V době, kdy světem hýbou informační technologie, se již téměř žádný podnik neobejde bez informačního systému. Ten podnikům slouží nejen k uchovávání informací a dat napříč celou organizací, ale také pomáhá s operativním řízením podniku, se správou dodavatelů či zákazníků, ale také při dlouhodobém strategickém plánování. Nepochybnou výhodou informačního systému je schopnost automatizace určitých procesů podniku, proto by měl být informační systém navržen a nasazen co nejefektivněji.

V poslední době roste zájem o tzv. Cloud computing, který podnikům umožňuje využívání informačních technologií bez nutnosti instalace softwaru či správy lokální infrastruktury. Jednoduše si požadované řešení pronajímají formou služby. Na této bázi je založen také informační systém ServiceNow, který je primárně určen pro správu IT služeb v podniku.

System ServiceNow nabízí široké možnosti customizace na míru podniku tak, aby přesně vyhovoval požadavkům na daný proces společnosti. Aby však byl systém efektivní, je nezbytné dodržovat doporučené best practices (nejlepší postupy) pro implementaci systému ServiceNow. Ty se vztahují nejen na nativní funkcionality systému, ale především na pokročilejší vývoj pomocí programovacího jazyka JavaScript.

Společnost, která vznikla odštěpením od jiné společnosti, využívá systém ServiceNow mimo jiné pro tvorbu a správu požadavků týkajících se síťových služeb. Odštěpením vznikla potřeba společnosti nasadit tento proces také do nového prostředí této společnosti, nicméně s nutnou optimalizací. Proto se obrátila na konzultantskou společnost, která se na implementaci systému ServiceNow specializuje.

2. Cíl práce a metodika

Cíl práce

Cílem práce je implementace optimalizovaného řešení pro správu požadavků do systému ServiceNow na základě provedené technické analýzy stávajícího řešení a požadavků zákazníka. Dílčími cíli jsou návrhy variant na implementaci, finanční odhady a vyhodnocení skutečných nákladů.

Metodika

V teoretické části práce byly na základě analýzy primárních a sekundárních dat vymezeny pojmy související s informačními systémy, popsána metodika ITIL a přiblížena problematika Cloud computingu. V neposlední řadě se v této části detailněji zaměřím na možnosti konfigurace systému ServiceNow, kterými se pak řídím v praktické části práce.

Technická analýza stávajícího řešení byla provedena formou expertního posudku týmem zkušených konzultantů, jejímž jsem členem. Dále byl proveden sběr požadavků zákazníka a jejich přenesení formou implementace do nového řešení.

Výstupem práce tedy bude optimalizované řešení na základě splnění požadavků zákazníka.

3. Teoretická východiska

3.1. Základní pojmy

Vzhledem k tomu, že práce je zaměřena na informační systémy, je třeba si definovat základní pojmy, které jsou sice všem dobře známé, nicméně je na místě uvést zde jejich definice.

3.1.1. Informace a data

Informace je velice rozšířený pojem, který má mnoho definic pro různé pohledy na tento pojem. V zásadě se dají informace obecně definovat jako *“poznatky, které uspokojují konkrétní informační potřebu svého příjemce”* [1, str. 13].

Informace mohou mít různou formu, např.:

- číselná data
- zvuk
- obraz
- text [1]

Informace představují data, která mají již pro konkrétního člověka nějaký význam. **Data** tedy můžeme definovat jako objektivní fakta o událostech, kterým se snažíme porozumět a interpretovat je. Data se získávají měřením, šetřením, experimentem či pozorováním a mohou mít stejné formy jako informace. [2]

3.1.2. Systém

Stejně jako pojem informace je obecně známý a rozšířený také pojem systém. Dle zakladatele teorie systémů, Ludwiga von Bartalanffy, je systém *“agregací podobných nebo alespoň vzájemně souvisejících jevů, věcí, procesů a souborů pravidel pro jejich jednání (fungování).”* [3]

Každý systém se skládá z množiny prvků představující části, na které lze systém účelně dělit. Systém je možné dále členit na subsystémy, přičemž každý systém je součástí vyššího systému. Nicméně systém může také existovat samostatně bez nutnosti vztahu k jinému systému. [3]

3.2. Informační systém

S nárůstem dat a jejich transformace na informace vznikla potřeba najít způsob, jak takto velké množství dat a informací efektivně získávat, uchovávat, zpracovávat, předávat dalším uživatelům či prezentovat. Z těchto důvodů se začaly využívat informační systémy, představující provázané soustavy všech dat, které se v daném podniku vyskytují. [1]

Také pojem informační systém má mnoho definic, ze všech možných jsem vybrala definici, popisující informační systém následovně: *“Informační systém je definován jako soubor lidí, technických prostředků a metod zabezpečujících sběr, přenos, uchování a zpracování dat za účelem tvorby a prezentace informací pro potřeby uživatelů činných v systémech řízení.”* [1, str. 32]

3.2.1. Členění informačních systémů

Většina větších podniků je rozdělena na různá oddělení s různými řídicími úrovněmi. Každé z těchto oddělení pracuje s odlišnými typy informací a dat, proto je možné na informační systémy nahlížet z různých úhlů pohledu, ať už z pohledu architektury, z operativního či strategického pohledu. Některé z těchto pohledů jsou detailněji popsány níže.

3.2.1.1. Systémy pro podporu řízení

V dnešní době lze již s jistotou konstatovat, že všechny podniky, bez ohledu na jejich velikost, využívají informační systém pro správu svých dat. Robustnost systému je závislá na velikosti podniku, rozsahu zpracovávaných dat a schopnosti zaměstnanců s tímto systémem pracovat. Počet využívaných modulů daného systému se tedy odvíjí od velikosti podniku a předmětu jeho činnosti. [1]

Mezi systémy pro podporu řízení podniku patří zejména:

ERP systémy (Enterprise resource planning) - neboli také systém pro plánování podnikových zdrojů je aplikační software, který podporuje řízení a koordinaci veškerých disponibilních podnikových zdrojů s cílem zajistit potřeby trhu a vlastního podniku. ERP systém pokrývá všechny základní sféry podnikového řízení: nákup, prodej, sklad, majetek, lidské zdroje, finanční účetnictví, controlling, práce a mzdy, plánování a technickou přípravu výroby a mimo jiné také podporuje operativní řízení. ERP systémy jsou

považovány za stěžejní systémy podniku, které výrazně ovlivňují celkový podnikový business. [1]

MIS systémy (Management Information System) - neboli Manažerské informační systémy jsou určeny pro podporu práce manažerů a analytiků na různých organizačních úrovních. Spojuje nejen detailní pohledy, ale také pohledy s jistou mírou abstrakce a agregace. MIS obsahuje tři základní komponenty – **extrakční nástroje** (starají se o přenos, čištění a konverzi zdrojových dat ze systému do datového skladu), **databáze** (úložiště získaných dat z extrakčního nástroje), **analytické nástroje** (umožňují tvorbu cenných informací a znalostí pro podporu rozhodování na základě různých kombinací dostupných dat) a **prezentační nástroje** (představují grafickou podobu získaných dat formou tabulek a grafů). [1]

DSS systémy (Decision Support System) – systém pro podporu rozhodování bývá často doplňkovým systémem MIS, který v podstatě plní analytickou funkci, je tedy oporou firemním analytikům, kteří připravují podklady pro strategické řízení. Úkolem DSS systémů je shromáždění a rozpad velkého objemu dat na různých úrovních hierarchie podniku s různými kritérii a jejich kvalifikovaná analýza. [1]

EIS systémy (Executive Information System) - neboli také Strategické informační systémy, jsou používány na nejvyšší úrovni řízení. Využívají všechny dostupné informační zdroje vytvořené na nižších úrovních informačního systému, tj. úlohy transakčního charakteru, úlohy pro taktické a operativní řízení či úlohy pro podporu rozhodování. Úkolem těchto systémů je především připravovat a předkládat manažerům údaje pro podporu jejich činností (rutinních i nestandardních) v přehledné formě (textové i grafické). [1]

3.2.1.2. Systémy pro řízení vztahů s externími partnery

Pro řízení vztahů s externími partnery, ať už se jedná o zákazníky či dodavatele, slouží následující systémy:

CRM systémy (Customer Relationship Management) – systémy pro řízení vztahů se zákazníky se dle řady výzkumů považují za jeden ze základních konkurenčních zdrojů. Proces řízení a budování vztahů se zákazníky se skládá ze čtyř fází, kdy první z nich je výběr a oslovení zákazníka, druhým je obchodní transakce, kdy dochází k uzavření

smlouvy, třetí fáze plnění objednávky spočívá v dodání služby či produktu a čtvrtá fáze představuje zákaznický servis, kdy dodavatel poskytuje podporu implementační a instalační aktivity a zajišťuje služby. CRM musí zajistit optimální funkčnost celého tohoto procesu. Mezi přínosy tohoto systému rozhodně patří optimalizace všech procesů souvisejících se zákazníkem (marketing, komunikace, obchodní aktivity), anebo také možnosti vytvoření nových obchodních příležitostí či služeb. [1]

SCM systémy (Supply Chain Management) – systémy pro řízení dodavatelských řetězců koordinují všechny podnikové zdroje od dodavatelů, přes zpracovatele až k zákazníkovi, přičemž zákazník je buď koncový uživatel nebo další podnik. SCM vyžaduje nový přístup v řízení nákupu, kdy je nezbytné se soustředit místo na optimalizaci v rámci podniku na soustředění v rámci celkového dodavatelského řetězce. Mezi zásadní přednosti zavedení SCM v první řadě patří fakt, že veškerá data o dodavatelích jsou uchovávána v tomto modulu a všechny související informace jsou tedy ihned k dispozici. Další velkou výhodou je možnost kompletní integrace do ERP systému. [1]

Kancelářské informační systémy (Office Information Systems) – slouží k usnadnění komunikace v rámci celé firmy, k podpoře a organizaci jednotlivých týmů, správě dokumentů aj. Tyto systémy se využívají na všech úrovních řízení, kdy na operativní úrovni řeší předávání směrnic, dokumentů, vzájemnou komunikaci mezi členy týmu nebo také docházková aktivita. Na taktické úrovni jsou využívány pro řízení týmů, řešení marketingových či vývojových aktivit nebo pro spolupráci podnikových útvarů. A v neposlední řadě na strategické úrovni se využívá tento systém především pro zkvalitnění procesů tvorby nových strategií, spolupráci jednotlivých útvarů podniku a správu týmových aktivit. [1]

V rámci kancelářských informačních systémů jsou využívány také **systémy řízení workflow**, které zajišťují procedurální automatizaci podnikových procesů řízením posloupnosti pracovních činností za pomoci relevantních lidských či technických zdrojů. Jedna z **definic workflow** je následující: „*Workflow znamená automatizaci celého nebo části podnikového procesu, během kterého jsou dokumenty, informace nebo úkoly předávány od jednoho účastníka procesu ke druhému podle sady procedurálních pravidel tak, aby se dosáhlo nebo přispělo k plnění globálních podnikových cílů.*“ [1, str. 85]

3.2.2. Ekonomika informačních systémů

Otázky typu, jaké budou celkové výdaje na informační a komunikační technologie, jak efektivní tyto výdaje budou či zda bude návratnost investic uspokojivá není příliš vhodné si pokládat, jelikož všechny tyto odpovědi závisí především na firmě, která si nechává informační systém budovat. Finanční náklady lze velmi těžko specifikovat, jelikož nejsou definovány požadavky ani časový horizont. Může se jednat o tvorbu úplně prvotní fáze životního cyklu, tedy informační strategii, o fázi tvorby architektury, či pouze fázi implementace nebo provozu informačního systému na každodenní bázi. [1]

3.2.2.1. Členění výdajů na informační systém

Výdaje na informační systém lze klasifikovat dle následujících tří hledisek:

Časové hledisko – životní cyklus IS se zpravidla pohybuje v horizontu 7-10 let, v jehož průběhu lze náklady rozdělit. Tato doba se však postupně stále zkracuje v důsledku zvyšujících se požadavků na kvalitu a objem zpracovávaných dat a také z důvodu velmi rychlého rozvoje ICT. Z časového hlediska se **výdaje** rozlišují na:

- **Jednorázové** – nákup a instalace technických prostředků, testování SW, náklady na proškolení zaměstnanců aj.
- **Běžné** – pravidelné platby za zajištění provozu systému, výdaje na běžný provoz (upgrade, tonery, tiskárny aj.)
- **Skryté** – tyto náklady se nepřičítají k celkovým nákladům souvisejícím s provozem IS, ale jsou účtovány k nákladovému středisku. Představují náklady na doškolení zaměstnanců, služby, vstupy dat a obsluha terminálů aj. Žádoucí jsou nulové skryté náklady. [1]

Druhové hledisko – člení výdaje na IS dle nákladových položek vynaložených na jednotlivé části spojené s provozem. Tyto **výdaje** se dále člení na:

- **Hardware** – osobní počítače, notebooky, servery, mainframe, aj.
- **Software** – operační systémy, databázové systémy, aplikační software, síťový software, aj.
- **Pracovníky** – systémové správce, techniky, provozní personál, projektanty, vývojové pracovníky, aj.

- **Služby nakupované od externistů** – servis HW, vývoj a údržba aplikačního SW, komunikační služby, aj.
- **Režii útvarů pro IS** – správní režie, materiálová režie, energie, aj. [1]

Aplikační hledisko – představuje výdaje na zpracování jednotlivých agend, jako např. účetnictví, mzdová agenda, skladová evidence, manažerský informační systém, aj. Zpravidla se výdaje z tohoto hlediska nesledují a pokud ano, tak krátkodobě, např. když je potřeba zdůvodnit zavedení aplikace. Sledovat výdaje z aplikačního hlediska je smysluplné pouze v případě, že lze sledovat přínosy z daných aplikací, ty se však vyjadřují velmi těžko, jelikož je třeba tyto přínosy předem definovat a stanovit způsob jejich hodnocení (klasifikaci). [1]

3.3. ITIL

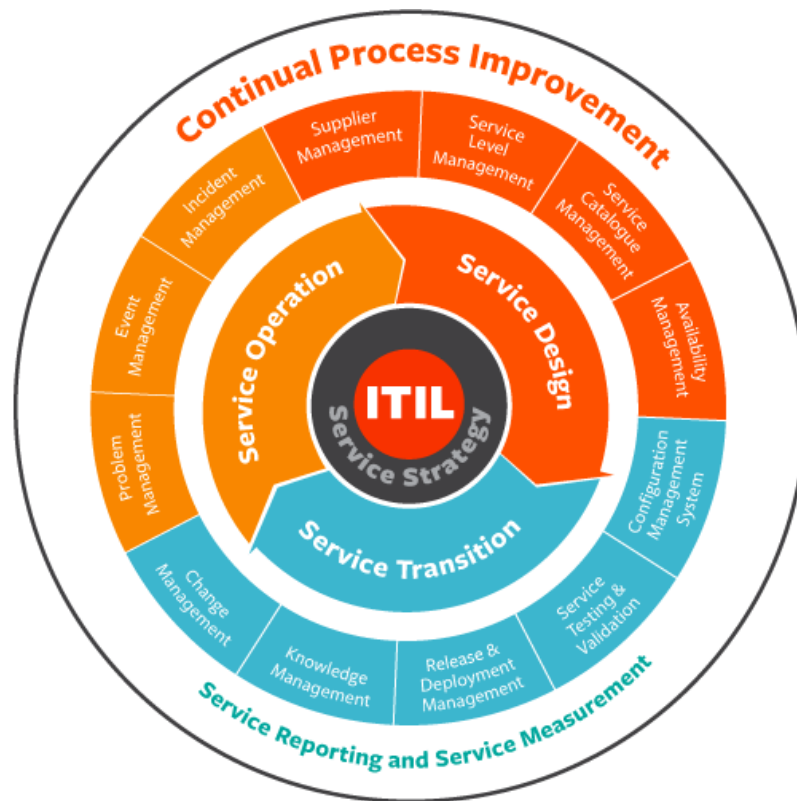
ITIL (Information Technology Infrastructure Library) představuje formou sbírky knih rozsáhlý a všeobecně rozšířený návod pro správu IT služeb, který se stal postupem času v podstatě standardem a souhrnem nejlepších praktik. ITIL poskytuje volně dostupný rámec zahrnující kompletní cyklus služeb IT a jako takový je dostatečně flexibilní pro přizpůsobení doporučení z knih ITIL potřebám a požadavkům konkrétní organizace. [4]

ITIL spravuje britský úřad vlády Jejicho Veličenstva neboli Cabinet Office a nejnovější verzí je aktualizovaná ITIL Verze 3 nesoucí označení **ITIL Edice 2011**. [4]

Stěžejními publikacemi této sbírky je 5 knih, přičemž každá z nich představuje fázi životního cyklu a popisuje příslušné procesy, principy, funkce a další související témata. Jedná se o následující publikace:

- Strategie služeb (Service Strategy)
- Návrh služby (Service Design)
- Přechod služby (Service Transition)
- Provoz služeb (Service Operation)
- Neustálé zlepšování služeb (Continual Service Improvement) [5]

Obrázek 1: Životní cyklus podle ITIL knih a jejich procesů



Zdroj: [5]

3.3.1. Strategie služeb

Strategie služeb je první knihou ITIL a také první fází životního cyklu, v jehož rámci se navrhuje, vyvíjí a implementuje Správa služeb IT jako strategické aktivum. Přitom je kladen důraz na celkové strategické pojetí nabídky služeb z hlediska potenciálu, výkonu a klíčových kompetencí a zároveň je nezbytné vzít v úvahu poptávku klientů a potenciální investice. [4]

„Strategie je plán, který ukazuje, jak chce organizace dosáhnout svých vytyčených cílů.“ [4, str. 33]. Nicméně strategie jsou ve skutečnosti mnohem komplexnější, strategie služeb mimo jiné specifikuje, jakým způsobem poskytovatel využívá služby k dosažení obchodních cílů svých zákazníků a tím zároveň i svých cílů. [4]

Svazek Service Strategy z řady ITIL nabízí nezbytné koncepty, které je nutné vytvořit a implementovat k dosažení úspěšné strategie služeb, přičemž jednoznačné zaměření je na definici strategie služeb IT, která je typickou součástí strategie IT. Strategie služeb

představuje specifický přístup poskytovatele, díky čemuž se odlišuje od konkurentů a nabízí tak zákazníkům jedinečnou přidanou hodnotu. [4]

3.3.1.1. Procesy strategie služeb

Procesy strategie služeb mají na starosti manažerské řízení správy služeb. Zároveň zajišťují orientaci na celkovou strategii služeb pro všechny fáze životního cyklu a také řádný finanční a strategický základ pro poskytování služeb. Strategie daného poskytovatele služeb, v závislosti na jeho typu, se přitom zaměřuje na strategii interního či externího zákazníka. [4]

Mezi procesy strategie služeb se řadí:

Správa strategie služeb IT – generuje strategické cíle tak, aby byla správa služeb IT souladu se strategií podniku. Tento proces se stará o perspektivu, plán, pozici a dané vzorce chování podniku ve vztahu ke službám a jejich správě v IT organizaci. [4]

Správa portfolia služeb – portfolio služeb zahrnuje všechny služby, které kdy poskytovatel IT služeb vyvinul, vytvořil nebo už poskytoval. Na základě poptávky podniku definuje, které IT služby má nabízet, proč by je měl zákazník využívat, jaké jsou cenové modely těchto služeb aj. Správa portfolia služeb tedy řídí a spravuje služby po celou dobu jejich životního cyklu. [4]

Správa financí služeb IT – má na starosti ekonomické hledisko vytvoření služby, tzn. skutečné náklady na vytvoření a správu IT služeb, či zvažování, zda službu vůbec vytvořit. Vyúčtování IT služeb je mnohdy složité, jelikož např. málokdy jsou skutečné náklady na zákazníka nebo službu adekvátně rozkryty. Tomu se tedy snaží správa financí zabránit. [4]

Správa požadavků – tento proces má za úkol pochopit, předpokládat a řídit požadavky zákazníka a včas zajistit kapacity tak, aby nedocházelo jejím nadbytkům či nedostatkům, což je vnímáno negativně jak pro poskytovatele, tak pro zákazníka. Spoluprací se zákazníkem by mělo plánování, analýza trendů či dohoda o úrovni služeb snížit výkyvy a nejistotu poptávky. [4]

Správa vztahů s businesssem – jedná se o spojení poskytovatele se zákazníkem na strategické a taktické úrovni podporující vzájemné porozumění. Poskytovatel služby na

základě tohoto procesu rozumí potřebám zákazníka a je schopen mu nabídnout služby odpovídající jeho požadavkům. [4]

3.3.2. Návrh služby

Návrhem služby se zabývá druhá kniha ITIL a jejím účelem je, jak již název napovídá, návrh a tvorba služeb v souladu s první knihou – strategií služeb. Mezi návrhem služby a neustálým zlepšováním služby (kniha pátá) je velmi úzká spojitost, jelikož už v průběhu návrhu služby se zohledňují a implementují aspekty a mechanismy pro zlepšení služby pro použití v dalším provozu. [4]

Počátečním bodem návrhu služby je integrace obchodních požadavků, pokračující přes vývoj a dokumentaci až k SLA (Service Level Agreement) a OLA (Operational Level Agreement). Procesy ITIL představují nedílnou součást návrhu služby a využívají se k poskytování a provozu služeb. [4]

Mezi procesy návrhu služeb patří:

Správa katalogu služeb – katalog služeb je nedílnou součástí poskytování služeb určenou zákazníkovi. Úkolem tohoto procesu je vytvoření a správa katalogu služeb, a to hlavně proto, aby byl k dispozici spolehlivý a ucelený zdroj informací o dostupných provozovaných službách. [4]

Správa úrovně služeb – tento proces smluvně ošetřuje požadavky zákazníků na poskytované IT služby a dále zajišťuje kontrolu plnění této smlouvy, tzn. zda je smluvená úroveň služby dodržována. Požadavky zákazníka na kvantitu a kvalitu služeb mohou být splněny pouze za předpokladu, že cíle úrovně služeb sledují obchodní požadavky. [4]

Správa dostupnosti – zajišťuje, aby služby odpovídaly schválené úrovni dostupnosti. Ta je velmi důležitým aspektem úspěšnosti. Pokud není dostupnost k dispozici, nevytváří služba pro uživatele očekávaný užitek a není tak pro něj užitečná. [4]

Správa kapacit – zajišťuje, aby kapacity služeb efektivně splňovaly požadavky týkající se kapacit a výkonnosti služby. Výstupem tohoto procesu je kapacitní plán a CMIS (informační systém správy kapacit), nebo také dlouhodobé předpovědi pro taktické a strategické plánování. [4]

Správa kontinuity služeb IT – zajišťuje přežití nezbytných služeb a infrastruktury IT v případě mimořádných událostí či katastrof. Plánování a příprava na krizové situace a následnou obnovu služeb IT umožňuje co nejnižší dopady na obchod a co brzký návrat k normálnímu režimu dle SLA. Výstupem tohoto procesu jsou strategie, pravidla a plány, ale také plánování testů, scénářů a hodnocení. [4]

Správa bezpečnosti informací – cílem tohoto procesu je pro podnik vytvoření a správa definované bezpečnosti informací. Ta se musí řídit zároveň podnikovou bezpečností, která pro aktiva, informace, data a služby IT představuje důvěrnost, integritu a dostupnost. Výstupem tohoto procesu mohou být předpisy bezpečnosti informací, procesy analýzy a správy rizik aj. [4]

Správa dodavatelů – zajišťuje dodržování smluv s dodavateli a třetími stranami a tím napomáhá k podpoře obchodních požadavků. Nezbytné samozřejmě je i dodržování smluv ze strany dodavatele. Výstupem mohou být informace o smlouvách, zvážení rizik nebo zprávy o dodavatelích. [4]

3.3.3. Přejít služby

Třetí kniha ITIL se zabývá přechodem navržené služby do reálného provozu. Tato fáze životního cyklu přebírá zodpovědnost za přenos strategie služby obsažené v balíčku návrhu služby z teorie do praxe, tzn. organizované a řízené převedení IT služby se všemi součástmi do provozu. [4]

Mezi procesy přechodu služeb patří:

Plánování a podpora přechodu – zajišťuje plánování a koordinaci potřebných zdrojů. Lze tak vytvořit pravidla, standardy a postupy pro činnosti spojené s přechodem služby. Výstupem tohoto procesu je nejen strategie a rozpočet přechodu služby, ale také integrované plány jejího přechodu. [4]

Správa změn – hlavním cílem správy změn je minimalizace počtu incidentů a výpadků způsobených změnami, možná nebezpečí plynoucí ze změn a maximalizaci užítka ze změn. Ty by měly probíhat rychle a kontrolovaně, díky standardizovaným metodám a postupům. Výstupem může být odmítnutí požadavku na změnu, schválená změna, návrh změny, aktualizovaný plán změn aj. [4]

Správa aktiv služeb a konfigurací – tento proces zajišťuje, aby byly k dispozici spolehlivé a pravdivé informace o aktivech společnosti, která jsou nezbytná pro dodávku služeb. Tyto informace zahrnují detaily k jednotlivým konfiguračním položkám, resp. aktivům, zvláště pak jejich vzájemné vztahy. Výstupem mohou být jak aktualizované konfigurační záznamy nebo informace o aktivech, tak i různé zprávy o stavu či informace o vztazích konfiguračních položek. [4]

Správa releasů a provozního nasazení – tento proces má na starost dodávku služby zákazníkovi tak, jak byla koncipována během jejího návrhu a aby ji mohl zákazník využívat a získávat tak požadované hodnoty. Proces plánuje a řídí kompletní sestavení, testování, a nasazení služby zákazníkovi. [4]

Validace a testování služby – Pomocí tohoto procesu by mělo být zajištěno dodání služby dle specifikace a požadavků obchodu. Prostřednictvím validace a testování je poskytnut objektivní důkaz, že služba splňuje požadavky v rámci definované úrovně služeb. Výstupem tohoto procesu mohou být výchozí konfigurace pro rozdílná testovací prostředí, uskutečněné testy a jejich výsledky aj. [4]

Vyhodnocení změny – vyhodnocuje úspěšnost změny a usměrňuje očekávání zapojených stran do reálných mezí. Poznatky z tohoto vyhodnocení slouží jako podpora pro schvalování správy změn. Výsledkem jsou mimo jiné průběžné zprávy vyhodnocení, a hlavně závěrečná zpráva pro správu změn. Proces vyhodnocení změny poskytuje také podstatné informace pro proces neustálého zlepšování služby. [4]

Správa znalostí – správa znalostí umožňuje přístup k podstatným, srozumitelným a rychle dostupným informacím tak, aby byla zlepšena nejen kvalita rozhodovacího procesu, ale také správně vyhodnocena daná situace. Díky správné správě znalostí jsou veškeré informace (znalosti) znovu použitelné, čímž se snižují náklady a zvyšuje spokojenost. [4]

3.3.4. Provoz služby

Čtvrtá kniha ITIL se věnuje provozu služby, jehož procesy a funkce se zaměřují na oblast organizace IT. Ta má za cíl zvyšovat účinnost a efektivnost poskytování služeb, stabilitu IT služeb je tedy nutné zajistit za všech okolností. [4]

Pro zákazníka je tato část životního cyklu nejdůležitější, je totiž viditelná skutečná hodnota služby, která je realizována. Implementované procesy a funkce však přinášejí užitek pouze za předpokladu, že jsou důsledně řízeny a kontrolovány. [4]

Mezi procesy provozu služby patří:

Správa událostí – hlavní úlohou správy událostí je identifikace a analýza událostí a vyvození vhodných opatření. Proces správy událostí začíná odhalením události, pokračuje vyvozením závěrů a končí volbou vhodné akce. Výstupem jsou záznamy již proběhlých událostí, mezi které patří události poukazující např. na porušení SLA nebo OLA aj. [4]

Správa incidentů – tento proces zaznamenává a zpracovává odchylky od běžného provozu systému. Incident je kategorizován a prioritizován tak, aby mohl být co nejrychleji a nejefektivněji vyřešen a provozní stav byl navrácen do stavu dle SLA. Výstupem je především obnovení služby a vyřešený incident, ale také pozitivní zpětná vazba zákazníkovi s informací o uzavření incidentu. [4]

Plnění požadavků – zabývá se plněním uživatelských požadavků, které však nejsou incidenty. Jedná se o požadavky související s provedením nějaké služby, jako např. žádost o nové zařízení. Požadavky mají zpravidla předem definovaný rámec a postup řešení a podléhají schvalování. Výsledkem tohoto procesu je ukončený požadavek, a to buď pozitivně, či negativně. [4]

Správa problémů – proces správy problémů má na starost snížení počtu opakujících se incidentů, v případě, že je zaznamenáno jejich zvýšení. Životní cyklus správy problémů sleduje problém od začátku do konce, tzn. od identifikace, přes analýzu, dokumentaci až po odstranění příčiny problému. Výstupem mohou vyřešené problémy a související aktivity, aktualizované záznamy o problémech, návrhy na změny aj. [4]

Správa přístupů – proces, který se stará o udělování přístupů k využívání služby oprávněným uživatelům, a naopak o zamezení přístupu neautorizovaným uživatelům. Proces má také zamezit zneužití služby pomocí nepřiměřených přístupových práv. Výstupem je tedy udělení oprávnění uživateli v souladu s předpisy bezpečnosti informací, dále záznamy současných a historických poskytnutých i zamítnutých oprávnění nebo zprávy o zneužití přístupových práv. [4]

Mezi funkce provozu služeb patří:

Service desk – slouží uživatelům jako primární kontakt pro hlášení incidentů a tvorbu požadavků. Interně slouží jako koordinační místo, odkud jsou incidenty a požadavky případně dále delegovány. Service desk může také v některých případech plnit funkci informování uživatelů či dotazování uživatelů na spokojenost. [4]

Technická správa – jedná se o skupiny, oddělení, nebo týmy pracovníků, kteří disponují expertními znalostmi a spravují infrastrukturu IT v podniku. Ve větších podnicích bývají znalosti rozděleny dle specializace do více oddělení (servery, síť, operační systémy atd).

Správa provozu IT – představuje tým, který má na starost každodenní činnosti potřebné pro správu IT služeb, podporu IT infrastruktury, a hlavně mají za úkol udržovat stabilitu infrastruktury a konzistentnost služeb. [4]

Správa aplikací – funguje na obdobném principu jako technická správa, pouze na úrovni aplikací. Má za úkol podporovat obchodí procesy podniku prostřednictvím aplikací, resp. jejich návrhem a provozem. [4]

3.3.5. Neustálé zlepšování služby

Poslední kniha ITIL nese název „Continual Service Improvement“ (CSI) a popisuje způsoby a metody neustálého zlepšování služeb. Vzhledem k tomu, že během celého životního cyklu služby se mohou měnit obchodní požadavky, je třeba odhalovat optimalizační potenciál a provádět zlepšení se zaměřením na nárůst efektivnosti, optimalizaci nákladů a maximalizaci účinnosti. [4]

Tato kniha je založena na principu **Demingova cyklu**, který se skládá ze 4 částí:

- **Plan** – hledá se potenciál ke zlepšení aktuálního stavu, provádí se odhad a dochází k vytvoření plánu na zlepšení.
- **Do** – dochází k realizaci plánu vytvořeného v předchozím kroku. Je nutné mimo jiné zajistit dostatečný rozpočet, identifikovat role a delegovat odpovědnosti, nebo také aktualizovat dokumentaci.
- **Check** – v tomto kroku dochází ke kontrole realizovaného zlepšení služby, zda došlo k očekávanému užítku, případně zda nastaly neočekávané vedlejší efekty a jak je vyhodnotit.

- **Act** – tento krok obnáší realizaci opatření k nápravě zjištěných odchylek a zajišťuje, aby nedošlo k navrácení do původního stavu. Zároveň se tím dostává opět do první fáze Plan a nastává nové kolo cyklu. [4]

3.4. Cloud Computing

Důvody vzniku cloud computingu se dají popsat jako snaha firem zaměřovat se primárně na cíl IT infrastruktury než na její technické provedení. Cloud computing se dá tedy definovat jako IT formou služby, kterou zákazníkům poskytuje provozovatel cloudu. Tyto IT služby mohou mít mnoho forem, ať už se jedná o infrastrukturu (servery, firewall), aplikační platformu (operační systémy) nebo aplikační software (ERP, CMS, emailový server). [6]

Základní princip cloud computingu spočívá v tom, že poskytovatel cloudu, který provozuje IT infrastrukturu, se kompletně stará o její údržbu. Na této infrastruktuře pak může provozovat i operační systémy a aplikace, které sdílí se svými zákazníky. Počet koncových uživatelů jednoho poskytovatele cloudu tak může být až několik tisíc. Zákazník si vybere z portfolia nabízených služeb poskytovatele ty, které chce využívat a za ty poskytovali platí zpravidla měsíční poplatky. Zákazník se tak nemusí starat o žádné systémy a jejich instalace a nevznikají mu tak žádné dodatečné náklady na jejich údržbu. [6]

3.4.1. Typy cloud computingu

Cloud computing se dá rozlišovat podle dvou hledisek, které se navzájem prolínají. První hledisko určuje, jak je cloud poskytován a říká se mu **model nasazení**. Podle něj se cloud rozlišuje na:

- **Veřejný** – někdy označován také jako klasický cloud computingový model, je takový model, kdy je výpočetní služba poskytnuta široké veřejnosti. Příkladem veřejného cloudu je třeba Skype nebo seznam.cz a to proto, že je určen celé populaci, nebo velkému počtu klientů. Poskytovaná funkcionalita je víceméně stejná pro všechny.
- **Soukromý** – tento typ cloudu je provozován pouze pro danou organizaci, a to buď samotnou organizací anebo třetí stranou.

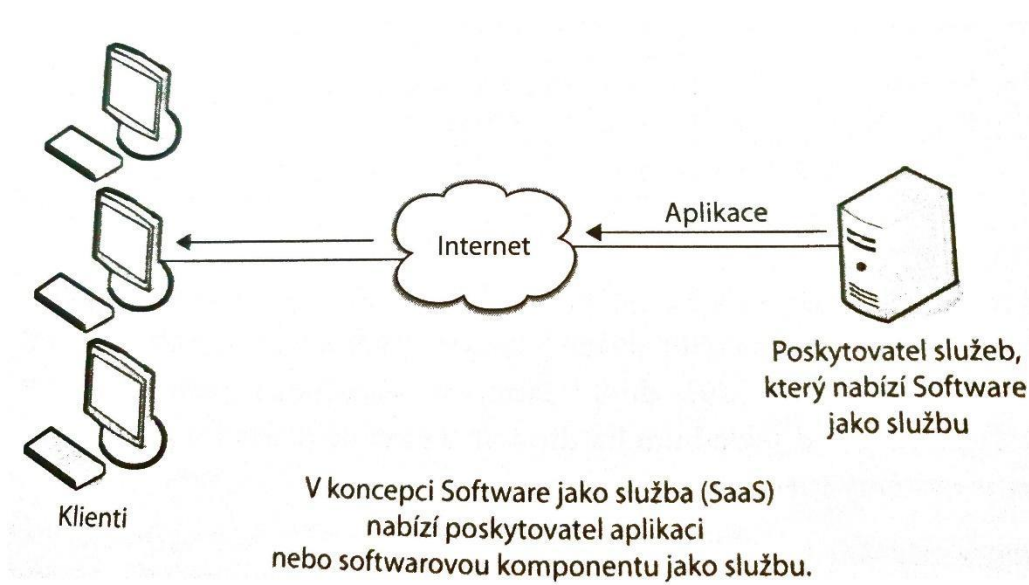
- **Hybridní** – hybridní cloudy jsou kombinací jak veřejných, tak soukromých cloudů, navenek působící jako jeden cloud, které jsou však propojeny pomocí standardizačních technologií.
- **Komunitní** - tento typ cloudu představuje infrastrukturu sdílenou mezi několika organizacemi, které ji používají. Tyto organizace zpravidla spojuje stejný obor zájmů. [6]

Druhé hledisko určuje, co je v rámci cloudu nabízeno, obvykle se jedná o software, hardware nebo jejich kombinace. Označuje se jako **služby** a ty jsou definovány následovně:

3.4.1.1. Software jako služba

Označována také zkratkou SaaS (Software as a Service). Je to model, kdy je aplikace nabízena a hostována jako služba zákazníkovi, který se k ní připojuje prostřednictvím internetu. Vzhledem k tomu, že je software hostován externě, nemá zákazník starosti s jeho správnou a podporou. Pokud se však poskytovatel rozhodne aplikaci změnit, zákazník nemá šanci jeho rozhodnutí jakkoliv ovlivnit. [7]

Obrázek 2: Model SaaS



Zdroj: [7]

Výhody tedy spočívají v tom, že zákazník využívá hotový software, který není nutné složitě integrovat s jinými systémy a o jehož kompletní údržbu se stará poskytovatel (opravy a aktualizace, udržování infrastruktury). [7]

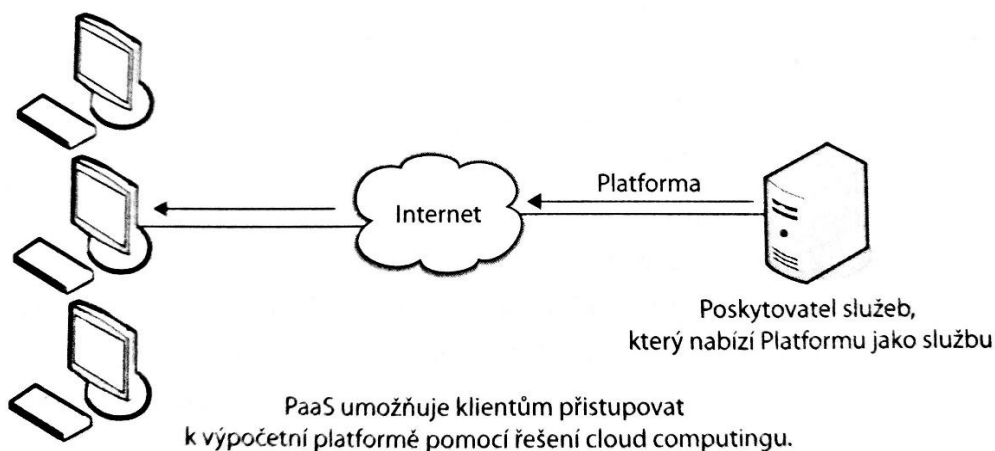
Mezi softwary, které se hodí pro model SaaS, lze zařadit takové programy, které nevyžadují rozsáhlou interakci s jinými systémy a plní jednoduchou úlohu vysoce výkonné aplikace. Příkladem těchto softwarů jsou například:

- Systémy CRM (Customer Resource management)
- Software pro videokonference
- Správa služeb IT
- Účetnictví
- Analýza webu
- Správa webového obsahu [7]

3.4.1.2. Platforma jako služba

Neboli také označována jako PaaS (Platform as a Service) a někdy i cloudware, poskytuje veškeré prostředky potřebné k vytváření služeb a aplikací výlučně za použití internetu, bez stahování nebo instalace software. K těmto službám patří především návrh, vývoj, testování, implementace a hostování aplikací, ale také týmová spolupráce, integrace databází či webových služeb, bezpečnost, škálovatelnost aj. [7]

Obrázek 3: Model PaaS



Zdroj: [7]

Velkou nevýhodou PaaS je však nepřenositelnost mezi poskytovateli, takže pokud si zákazník vytvoří aplikaci u jednoho poskytovatele cloudu a následně se rozhodne přejít k jinému poskytovateli, ne vždy je to možné – případně se jedná o velmi nákladnou investici. Rizikem je také ukončení činnosti poskytovatele, kdy zákazník přijde o celou aplikaci i s daty. [7]

Možnosti PaaS – koncepci PaaS lze nalézt ve třech různých typech systémů:

- **Nástroje pro vývoj doplňků (add-on)** – umožňují přizpůsobení stávající aplikace SaaS. Uživatelé a vývojáři PaaS si zpravidla musí zakoupit předplatné na tyto doplňkové aplikace SaaS.
- **Samostatná prostředí** – tato prostředí se využívají při obecném vývoji a nezahrnují licenční, finanční ani technické závislosti na specifických aplikacích SaaS.
- **Prostředí pouze pro poskytování aplikací** – podporují služby na úrovni hostingu, např. zabezpečení a škálovatelnost na vyžádání. Tato prostředí jsou bez vývojových, ladicích i testovacích funkcí. [7]

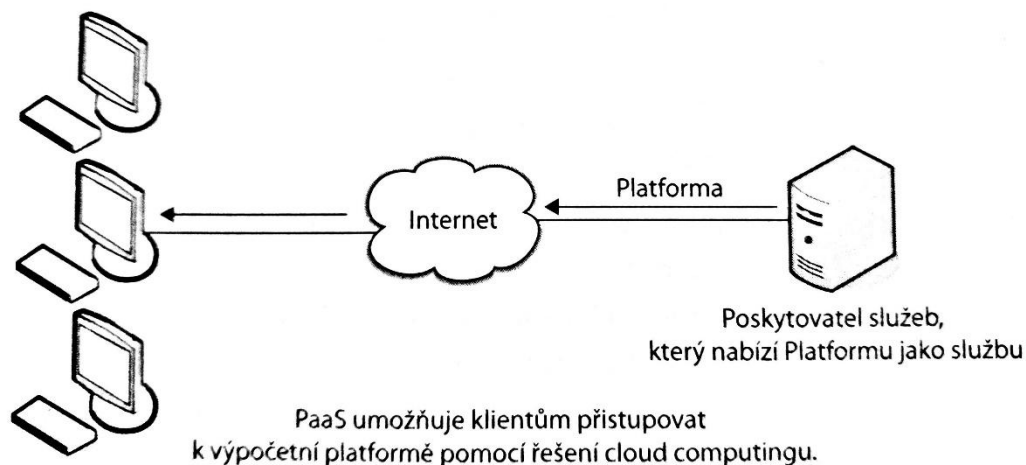
3.4.1.3. Hardware jako služba

Hardware jako služba (HaaS – Hardware as a Service) se také často označuje jako Infrastruktura jako služba (IaaS – Infrastructure as a Service). Od předchozích dvou možností se HaaS odlišuje jednoduše tím, že místo aplikace nabízí zákazníkům hardware k použití libovolným způsobem. To znamená, že místo nákladů na zakoupení serverů a jejich umístění v datovém centru si tyto prostředky zákazník pronajme od poskytovatele cloudu. [7]

HaaS umožňuje pronajímat **zdroje typu:**

- Místo na serveru
- Síťová zařízení
- Paměť
- Cykly procesoru
- Úložné místo [7]

Obrázek 4: Model HaaS



Zdroj: [7]

Kromě toho zahrnuje HaaS několik **složek**:

- **Smlouvy o úrovni služeb** – dohoda mezi poskytovatelem a zákazníkem o zaručení určité úrovně výkonu systému.
- **Počítačový hardware** – prostředky těchto komponent jsou pronajímány a často bývají kvůli lepší škálovatelnosti zapojovány do gridu (grid – sdílení výpočetní kapacity hardwaru pro řešení větších problémů).
- **Síť** – jedná se o hardware firewallů, směrovačů, vyrovnávání zatížení atd.
- **Internetová konektivita** – zákazník díky ní může přistupovat k hardwaru z vlastních organizací.
- **Prostředí virtualizace platformy** – umožňuje spouštět požadované virtuální počítače.
- **Fakturace spotřebovaných výpočetních zdrojů** – fakturace zákazníkům se zpravidla provádí na základě objemu využitých systémových prostředků. [7]

3.5. ServiceNow

ServiceNow je informační systém, který funguje na bázi cloudové služby SAAS (Software-as-a-service). Poskytuje zejména kompletní ITSM řešení, jádro společnosti se tedy točí kolem řízení provozních událostí informačních technologií: Incident, Problem,

Change. Kromě IT Service Managementu poskytuje ServiceNow řešení také pro mnoho dalších oblastí. Mezi hlavní kategorie patří:

- IT – IT Service Management, IT Operation Management, IT Business Management
- Bezpečnost (Security) - Security Operations, GRC – Governance, Risk & Compliance
- Lidské zdroje (HR) - Case and Knowledge Management, Employee Service Center, Enterprise Onboarding and Transition
- Zákaznický servis (Customer Service) [8]

Obrázek 5: Logo společnosti ServiceNow



Zdroj: www.servicenow.com

3.5.1. Konfigurace ServiceNow

Pro správnou konfiguraci a customizaci systému se používají následující základní developerské funkcionality. Ve většině z nich je možné použít programovací jazyk JavaScript, který využívá vlastní ServiceNow API.

3.5.1.1. Out-of-the-box funkce

ServiceNow nabízí tzv. Out-of-the-box funkcionality, což znamená vestavěné konfigurace systému, kde není nutné použít skriptování nebo kde by se skript neměl měnit. Je to velice důležitý koncept, jelikož systém nedělá aktualizace customizovaných objektů.

Například, pokud někdo přidá řádek do Client scriptu, který je vytvořen systémem, tato změna způsobí, že se Client script při updatu systému na vyšší verzi neaktualizuje a nemusí tedy fungovat správně. Je tedy lepší nastavit ho jako neaktivní a vytvořit kopii, kterou lze konfigurovat dle potřeb.

3.5.1.2. Business Rules

Business rule je skript na straně serveru, který se spouští, když je záznam zobrazen, vložen, aktualizován, smazán anebo pokud je vytvořen dotaz na tabulku.

Používá se pro splnění tasku (úkolů), jako je automatická změna hodnot v polích formuláře za splnění určitých podmínek nebo vytvoření událostí pro e-mailové oznámení a akce nastavené skriptem. [9]

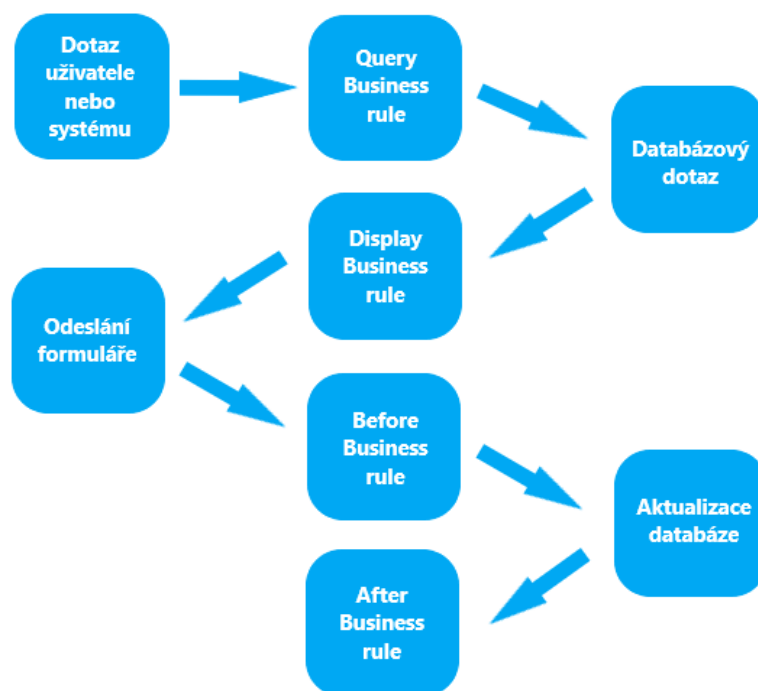
Jak Business rule funguje

Pro správnou konfiguraci Business rule je nutné nejprve stanovit, kdy se má spouštět a jakou akci má provést.

Business rule se spouští na základě dvou sad kritérií:

- Nakonfigurovaný čas, kdy se má Business rule spustit s ohledem na záznam, který je procházen nebo modifikován.
 - **Before** – poté, co uživatel odešle formulář, ale předtím, než se akce zapíše do záznamu v databázi.
 - **After** – poté, co uživatel odešle formulář a poté, co se akce zapíše do záznamu v databázi.
 - **Async** – když se spustí naplánovaná úloha vytvořená Business rulou. Systém vytvoří naplánovanou úlohu z Business rule poté, co uživatel odešle formulář a poté, co se zapíše akce do záznamu v databázi.
 - **Display** – před zobrazením formuláře uživateli, těsně po čtení dat z databáze.
- Databázová operace, kterou má systém provést na záznamu.
 - **Insert** – když uživatel vytvoří nový záznam a systém ho vloží do databáze
 - **Update** – když uživatel upravuje existující záznam
 - **Query** – před odesláním dotazu na záznam nebo seznam záznamů do databáze. Typicky se dotaz na databázi používá pro Before Business rule.
 - **Delete** – když uživatel smaže záznam. [10]

Obrázek 6: Procesní tok Business rule



zdroj: [10]

3.5.1.3. UI Policies

UI Policies dynamicky mění chování informací a polí na formuláři a kontrolují vlastní procesní tok úkolů. Je to tedy funkcionalita na straně klienta.

Například, UI Policies mohou být použity pro nastavení pole “Number” pouze pro čtení, pole “Short description” povinné anebo pro schování jiných polí.

Základní UI Policies nevyžadují žádné skriptování, nicméně pro pokročilejší akce lze využít možnost skriptu. Všechny akce, které UI Policy mohou provádět. Lze také nakonfigurovat přes Client Script, nicméně pokud není nutné skriptování a pro rychlejší načítání formuláře se doporučuje používat UI Policies. [11]

Tvorba UI Policies

Pro správnou tvorbu UI Policy je důležité nadefinovat následující pole, podmínky a akce:

- Tabulka – tabulka, pro kterou má být modifikován její formulář
- Active – zaškrťovací pole, které určuje, zda je UI Policy aktivní. Pouze aktivní UI Policies jsou aplikované

- Krátký popis – pole s krátkým shrnutím k čemu UI Policy slouží
- Podmínky – pokud jsou zadané podmínky splněny, UI Policy se aplikuje. Podmínky se definují přes filtr podmínek na základě zvolené tabulky. Vybírají se dostupná pole na zvolené tabulce a akce, které se mají s vybranými poly stát.
- UI Policy akce – po splnění výše zadaných podmínek se aplikují nadefinované akce, které má UI Policy provést. V rámci UI Policy akcí se definuje následující:
 - Název pole – pole ze zvolené tabulky, na které se má akce stát, po splnění podmínek. V rámci UI Policy může být vybrané libovolné množství polí. Pro každé pole se určuje, zda má být povinné, viditelné, nebo pouze pro čtení.
 - Povinné – určuje, zda bude zvolené pole povinné vyplnit pomocí výběru z možností: nechat být, pravda, nepravda
 - Viditelné – určuje, zda bude zvolené pole viditelné na formuláři pomocí výběru z možností: nechat být, pravda, nepravda
 - Pouze pro čtení – určuje, zda bude zvolené pole na formuláři pouze pro čtení pomocí výběru z možností: nechat být, pravda, nepravda
- Skript – pokud pro správné definování UI Policy nestačí výše uvedené možnosti. lze využít možnosti skriptování pomocí API na straně klienta. [11]

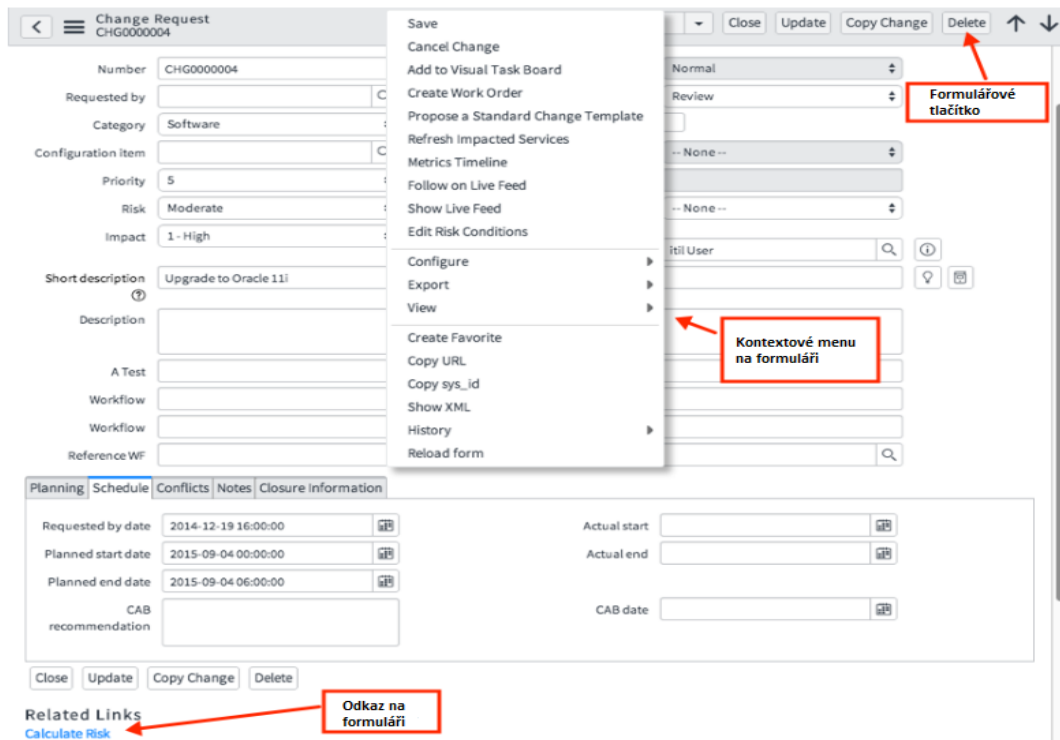
3.5.1.4. UI Actions

UI actions zahrnují tlačítka, odkazy a položky kontextového menu na formuláři či seznamu. Nastavením UI actions vzniká interaktivní uživatelské rozhraní, customizované a specifické pro aktivity daného uživatele. [12]

Ovládací prvky UI actions

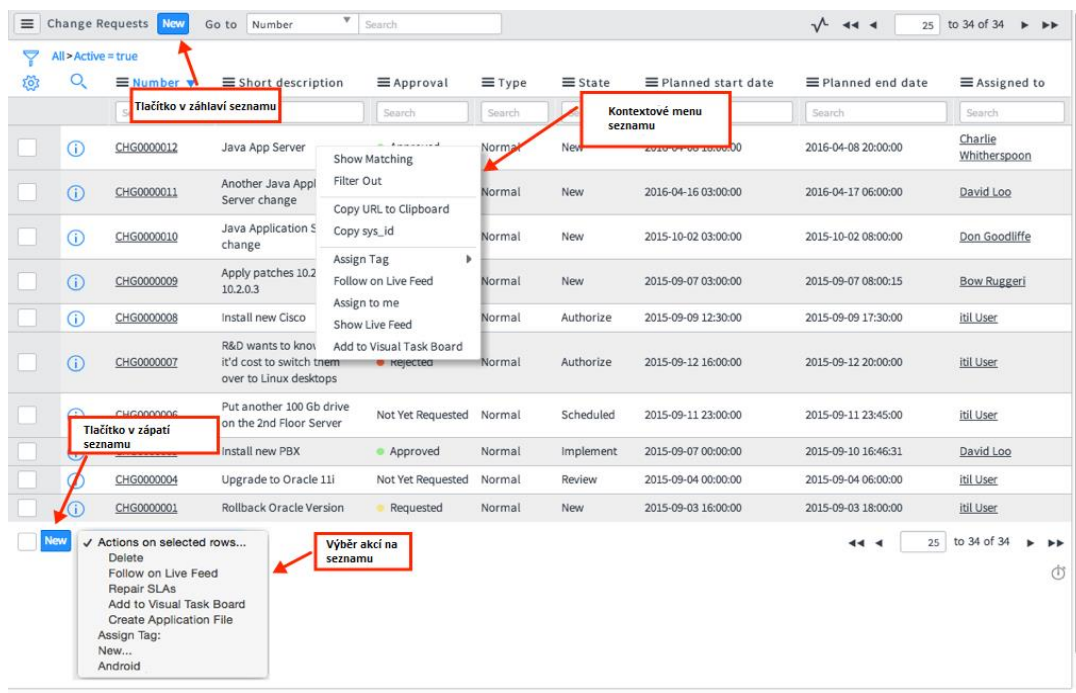
- Tlačítko na formuláři, tlačítko v záhlaví či zápatí seznamu
- Položka kontextového menu na formuláři či seznamu, která se zobrazí při kliknutí pravým tlačítkem na hlavičce formuláře
- Související odkaz na formuláři či v zápatí seznamu
- Položka v menu pro výběr akce v zápatí seznamu [12]

Obrázek 7: Formulářové UI Actions



zdroj: [12]

Obrázek 8: Seznamové UI Actions



zdroj: [12]

3.5.1.5. Client Script

Client Script umožňuje systému spustit JavaScript na straně klienta, jakmile dojde k určité události, jako například při načtení formuláře, při odeslání formuláře či při změně hodnoty pole.

Client script se využívá pro konfiguraci formulářů, polí a hodnot v polích, zatímco uživatel využívá daný formulář. Client script může nastavovat

- pole viditelná či neviditelná,
- pole jen pro čtení či editovatelná,
- pole povinná či nepovinná na základě uživatelské role,
- hodnotu pole na základě hodnoty jiného pole,
- měnit možnosti výběru na základě uživatelské role,
- zobrazovat zprávy či oznámení na základě hodnoty v poli.

Typy Client scriptů určují, kdy se daný skript má spustit:

- **onLoad()** – spouští se při prvním vykreslení formuláře systémem a předtím, než uživatel vloží data. Obvykle onLoad() Client skripty provádějí manipulaci s aktuálním formulářem na straně klienta nebo nastavují výchozí hodnoty záznamu.
- **onSubmit()** – spouští se, jakmile je formulář odeslán. Tento typ skriptu zpravidla validuje data na formuláři a zajišťuje, že odeslaná data dávají smysl. Client script onSubmit() může zrušit odeslání formuláře a vrátit zadanou hodnotu jako chybnou.
- **onChange()** – spouští se, když se změní hodnota specifického pole, pro které je tento skript nastaven.
- **onCellEdit()** – tento skript se spouští při jakékoliv interakci s polem (např. kliknutím do pole). [13]

3.5.1.6. Script Include

Script Include se využívá jako úložiště JavaScript kódu běžícího na straně serveru.

Pokud např. developer potřebuje v Business rule využít script, který je příliš dlouhý, vytvoří Script Include, kam celý script zapíše a v Business rule pak jen volá tento script. Script include může obsahovat hned několik funkcí, které např. souvisí s daným vývojem a volat jen konkrétní funkci, kterou zrovna potřebuje. Zpravidla se využívá pro serverovou

stranu, nicméně je možné daný Script Include zpřístupnit také pro klientskou stranu a volat v Client Scriptu. [14]

Použití Script Include

Při tvorbě Script Include lze využít jako formát jakýkoliv již existující Script Include. V následujícím příkladu má Script Include název „NewInclude“ a obsahuje funkci s názvem „myFunction“. Je důležité, aby název skriptu odpovídal názvu třídy, prototypu a typu. Po vytvoření nového skriptu, je nutné uvést jeho název a ten poskytne fragment kódu s třídou a prototypem nastaveným správně. [14]

Obrázek 9: Tvorba Script Include

```
var NewInclude =Class.create();

NewInclude.prototype={
  initialize :function(){},

  myFunction :function(){//Put function code here},

  type : 'NewInclude'};
```

zdroj: [14]

Volání uložené funkce ve Script Include pak probíhá následovně:

Obrázek 10: Volání funkce ze Script Include

```
var foo =new NewInclude();
foo.myFunction();
```

zdroj: [14]

3.5.2. ServiceNow Best Practices

Každý developer, který konfiguruje systém ServiceNow, by se měl řídit následujícími doporučenými praktikami, které mohou sloužit jako návod pro vývoj aplikací napsáním efektivních a snadno čitelných skriptů, vytvářením opětovně použitelné logiky a předcházením úskalím výkonnosti. ServiceNow popisuje celkem 37 doporučených postupů, pro potřeby práce budou popsány jen ty, které jsou relevantní pro daný proces.

3.5.2.1. Skriptování

Lehce čitelný kód – K vyvinutému kódu může mít přístup i jiný developer, proto je nezbytné psát kód tak, aby ho mohl snadno přečíst i kdokoliv jiný. Znamená to tedy používat komentáře k jednotlivým částem kódu, používat mezery mezi částmi kódu, nebo psát kód jednoduše, aby ho zvládl přečíst i méně zkušený JavaScript developer.

Proměnné – Při tvorbě kódu je potřeba dát proměnným smysluplná a odpovídající jména, díky kterým bude na první pohled jasné, co daná proměnná představuje. Tzn. místo použití jednotlivých písmen jako názvu proměnné (a=0), definovat např. var answer = 0. [15]

3.5.2.2. Client Script

Pořadí – Pro optimální využívání Client scriptů se doporučuje nastavit pro jednotlivé skripty pořadí, ve kterém se mají spouštět a tím tedy specifikovat, která akce se má provést dříve a která později. Pořadí spouštění se počítá vzestupně, tzn. Client script s nižší hodnotou (100) se spustí dříve, než ten s vyšší hodnotou (300). [15]

UI Policy – pokud je to možné, je třeba zvážit možnost použití UI Policy pro nastavení atributů polí na povinné, pouze ke čtení, nebo viditelné. V tomto případě není nutné použít skript, a tudíž nebude tolik ovlivněn výkon platformy. [15]

Kód uzavřený ve funkcích – Client Script bez použití funkce může způsobit problémy napříč platformou, proto je v jejich výchozím stavu již funkce předvyplněná. Pokud by Client Script nebyl uzavřen do funkce, jeho proměnné a další objekty by byly dostupné a sdílené se všemi ostatními Client Skripty a pokud jsou v různých skriptech definované proměnné se stejným názvem, mohou tyto skripty kolidovat. Může to vést k nečekaným následkům, které je těžké odstranit. [15]

3.5.2.3. Business Rule

Používat podmínky – Vzhledem k tomu, že Business Rule jsou vyhodnocovány vždy, když na záznamu dojde k vložení, aktualizaci, odstranění nebo dotazování, je důležité používat podmínky. Podmínky jsou vyhodnoceny před provedením Business Rule, pokud je podmínka splněna, skript se vyhodnotí a provede. Pokud neexistuje žádná podmínka, systém předpokládá, že Business Rule by měla být vyhodnocena a provedena pro každou akci k záznamu v zadané tabulce. [15]

Kód uzavřený ve funkcích – stejně jako v případě Client Scriptu, i při tvorbě Business Rule je nutné uzavírat kód do funkcí, v případě jeho použití. I zde je již funkce systémem předvyplněná a je nezbytné toto pravidlo dodržovat, aby nedocházelo ke kolizi proměnných na serverové straně a k negativnímu ovlivnění výkonu platformy. [15]

Vyvarovat se rekurzivní Business Rule – při tvorbě Business Rule nepoužívat metodu `current.update()` ve skriptu. Metoda `update()` spouští Business rule na stejné tabulce pro operace insert a update zároveň, což může vést k opakovanému volání Business rule dokola. Když je rekurzivní Business rule rozpoznána, systém ServiceNow ji automaticky zastaví a zaznamená error do systémového protokolu. Toto chování však může způsobit problémy s výkonem systému. [15]

4. Praktická část

4.1. Představení společnosti

Společnost E je mezinárodní společností působící v oblasti těžební techniky, jako jsou vrtné plošiny, rypadla, nástroje pro vrtání hornin, skalní výztuže a další související produkty a služby. Tato společnost vznikla koncem roku 2017 odštěpením od mateřské společnosti A, která je předním výrobcem a dodavatelem zařízení pro průmysl a stavebnictví.

Odštěpením společnosti E vznikla také potřeba oddělit a optimalizovat současné společné procesy a aplikovat je na nově vzniklou společnost. Pro potřeby této práce jsem použila proces týkající se informačních technologií, tedy interní proces objednávky a správy síťových služeb (Network Service Request). Pro tento proces využívá společnost informační systém ServiceNow.

Společnost E nebyla spokojena se stávajícím řešením současného dodavatele, a proto si vybrala pro realizaci nového řešení společnost D, která je rychle expandující společností v oblasti konzultačních služeb nejen v rámci systému ServiceNow, a která v České republice působí od roku 2005.

4.2. Network Service Request proces

Společnost A a E mají definované tzv. Sites, což jsou v podstatě místa, kde se nachází budovy společnosti po celém světě a ke kterým tedy může být zřízena přípojka. Součet těchto Sites obou společností je necelých tisíc, z toho společnosti E náleží zhruba 300.

Proces Network Service Request zajišťuje průběh objednání zvolené služby, a to:

- LAN nebo WLAN služby,
- Internetové připojení,
- WAN připojení,
- Managed Services.

Proces začíná v okamžiku zadání požadavku do systému kompetentní osobou, tento požadavek je přiřazen určité skupině řešitelů, kteří kontaktují dodavatele požadované služby a žádají o cenovou a technickou nabídku. Zaslání nabídky dodavatelem trvá standardně 21 dní.

Po obdržení nabídky ji řešitel přiloží do systému k danému požadavku, uvede, zda je po technické stránce vše v pořádku a čeká na potvrzení objednávky od dodavatele služeb. Obdržené potvrzení objednávky srovná s dříve zaslouanou nabídkou od dodavatele, zda veškeré údaje uvedené v nabídce souhlasí s potvrzením, tzn. cena, datum dodání atd. Potvrzení objednávky je také přiloženo do systému a je vytvořena související dokumentace.

Zároveň zasílá požadavek na schvalovatele, který podepisuje objednávku. Schvalovací proces je rozdílný dle požadované služby a bude popsán v technické části práce. Jakmile je po technické a schvalovací stránce vše v pořádku, dochází k realizaci požadavku.

Pracovníci dodavatelské společnosti kontaktují lokální pracovníky dané Site (místa určení) a domluví termín instalace. Po dodání se do systému zapíše datum realizace a jakmile je celý požadavek uzavřen, zapíše se také datum kompletnosti.

4.3. Technická analýza stávajícího řešení

4.3.1. Předmět a cíle analýzy

Technická analýza současného řešení se prováděla na testovací instanci společnosti A, se kterou společnost E dosud sdílí prostředí pro řešení procesu. Analýza se prováděla pro zhodnocení technického stavu procesu v systému a odhalení nedostatků, které mají být eliminovány v novém řešení.

Hlavní body analýzy jsou následující:

- Obecná analýza implementace nástroje pro Network Service Request Management založená na dokumentaci poskytnuté zákazníkem.
- Analýza současné úrovně dodržování best practices v oblasti technické implementace.
- Analýza a návrh možností migrace tak, aby bylo možné vytvářet stejné funkce v jiné instanci.
- Analýza a návrh oblastí, které by mohly být vylepšeny s cílem snížit úsilí a náklady potřebné pro upgrade ServiceNow instance.

Cílem technické analýzy je poskytnout následující informace:

- Zjištěné problémy týkající se schopnosti vylepšení, spravovatelnosti, škálovatelnosti a výkonnostních aspektů technické implementace nástroje pro Network Service Request Management implementovaného na platformě ServiceNow.
- Doporučení pro přizpůsobení implementovaných konfigurací a úprav pomocí best practices doporučených systémem ServiceNow.
- Určenou míru dodržování best practices při implementaci nástroje v systému ServiceNow.
- Identifikace rizika souvisejícího s bezproblémovým upgradem instance služby ServiceNow.

4.3.2. Výstupy analýzy

Pro správné porozumění výsledkům analýzy byly definovány následující oblasti dopadu zjištěných nedostatků, přičemž objevené nedostatky mohou zasahovat do více oblastí.

Tabulka 1: Oblasti dopadu nalezených chyb

Oblasti dopadu	Popis
Error	Problém v platformě ServiceNow nebo v konfiguraci instance, který způsobuje neočekávané chování.
Ovladatelnost	Potřebné úsilí pro údržbu konfigurací a customizací
Výkon	Rychlost a odezva aplikace ServiceNow vnímána koncovými uživateli.
Škálovatelnost	Schopnost implementace nebo customizace aplikací ServiceNow vyhovět zvýšeným nárokům s přijatelnou úrovní výkonu.
Upgradovatelnost	Schopnost konfigurace nebo customizace úspěšně upgradovat a nebránit procesu upgradu produktu.
Použitelnost	Schopnost efektivně navigovat koncového uživatele a provádět úkoly v rámci každé aplikace. Vztahuje se na použitelnost platformy, ne použitelnost procesu.

zdroj: [16]

Následující zjištění se týkají zásadních nedostatků u jednotlivých nástrojů konfigurace. U každého zjištění je uvedena oblast (oblasti) dopadu definované výše, míra kritičnosti (nízká, střední vysoká) a uvedena doporučení vhodná pro nápravu.

4.3.2.1. Business rules

Customizované funkcionality auditu

Když se změní nákladové středisko, business rule „Cost Center Changes“ vkládá data do vytvořené tabulky „u_asset_audit“ místo aby byly použity nativní funkce systému.

- Oblast dopadu – ovladatelnost, použitelnost
- Kritičnost – nízká
- Doporučení – využít nativní funkce auditu

Business ruly vytvářející ID a jmenné konvence

Velký počet Business rule manipulujících s daty při vkládání nebo aktualizování (pojmenování, generování ID). Takové Business ruly nejsou nutné, pokud se využívají podobné nativní funkce. Například Business rule „WAN Dynamic name for switch“ vytváří název dynamicky na základě jmenné konvence. Hrozí potenciální problémy s výkonem při manipulaci s velkými daty, např. v CMDB.

- Oblast dopadu – ovladatelnost, výkonnost, použitelnost
- Kritičnost – nízká
- Doporučení – využít nativní funkce platformy – správa čísel, vlastnosti slovníku

Customizované Business ruly místo workflow aktivit

Vysoký počet pomocných Business rule, které se spouštějí během životního cyklu ticketu. Nejsou dodržovány následující doporučené best practices:

- Použití aktivit v rámci workflow
- Nevyvíjet customizovanou logiku, pokud jsou k dispozici nativní funkce

Příklad: Business rule „Autopopulate approver field“ automaticky vyplňuje vlastní pole schvalovatele. Nabízí se tedy příležitost přehodnotit nastavení schvalování tak, aby tyto customizované logiky nebyly nutné.

- Oblast dopadu – ovladatelnost, škálovatelnost, upgradovatelnost
- Kritičnost – střední
- Doporučení – odstranit business rule, které používají vlastní kód v různých krocích procesu, místo toho použít workflow aktivity.

Business ruly, které závisí na pevně nakódovaných hodnotách

Pevně nakódované hodnoty nesmí být nikdy používány ve vyvíjených objektech, jelikož kód poté ztrácí svoji dynamičnost.

Příklad pevně nakódované hodnoty:

```
short_description','='Perform Technical Validations'
```

```
(...) isMemberOf('ServiceDesk-VIP') (...)
```

Změna jmenné konvence ticketu nebo jména skupiny vede k nutným vývojovým změnám (úprava kódu). Také, jelikož jsou tyto Business ruly spouštěny v jednotlivých krocích procesu, měly by být použity workflow aktivity.

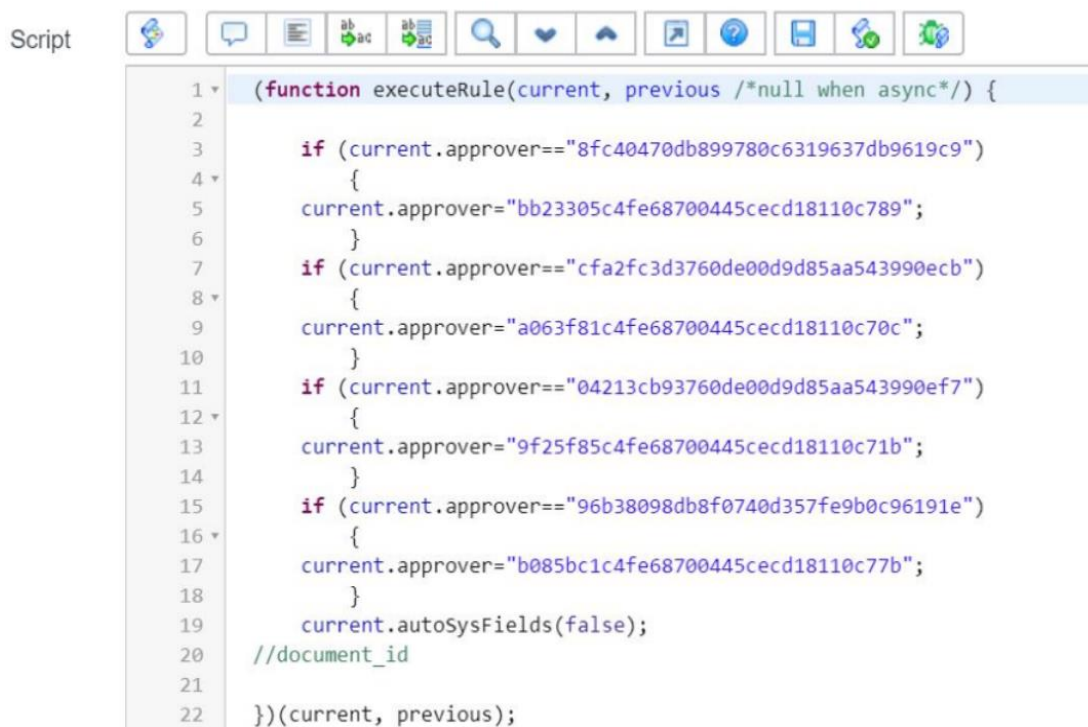
- Oblast dopadu – Error, ovladatelnost
- Kritičnost – vysoká
- Doporučení – předělat celé řešení použitím workflow. Při používání workflow by hodnoty, které mají být v podmínkách, měly být přečteny ze systémových rekvizit nebo z jiných aktualizovatelných dat.

Pevně nakódované osoby ve skriptech

Business Rule “WAN_TEST_Change Approvers” je vyvinuta pro změnu přiřazeného schvalovatele žádosti z jedné předdefinované osoby na jinou.

Ve skriptu jsou pevně nakódovaná sys_id uživatelů.

Obrázek 11: Skript znázorňující pevně nakódované hodnoty



```
Script
1 (function executeRule(current, previous /*null when async*/) {
2
3     if (current.approver=="8fc40470db899780c6319637db9619c9")
4     {
5         current.approver="bb23305c4fe68700445cecd18110c789";
6     }
7     if (current.approver=="cfa2fc3d3760de00d9d85aa543990ecb")
8     {
9         current.approver="a063f81c4fe68700445cecd18110c70c";
10    }
11    if (current.approver=="04213cb93760de00d9d85aa543990ef7")
12    {
13        current.approver="9f25f85c4fe68700445cecd18110c71b";
14    }
15    if (current.approver=="96b38098db8f0740d357fe9b0c96191e")
16    {
17        current.approver="b085bc1c4fe68700445cecd18110c77b";
18    }
19    current.autoSysFields(false);
20    //document_id
21
22 })(current, previous);
```

zdroj: ServiceNow instance společnosti A

Toto je velmi nezralé a jednoduché řešení, které bylo vyvinuto pravděpodobně na základě individuálního požadavku. Zachování této Business rule vede k nutnosti dalšího vývoje při běžných provozních změnách, jako například změna personálu.

Tato Business rule byla pravděpodobně vyvinuta v produkčním prostředí, což absolutně popírá best practices.

- Oblast dopadu – Error, Ovladatelnost
- Kritičnost – vysoká
- Doporučení – odstranit tuto Business rule, lze použít nativní delegativní funkce systému

Nepoužívání Script Include pro zpracování kódu

Nejvýznamnější problém týkající se konfigurace Business rule je nevyužívání Script Include pro globální Business ruly, které nemají žádné podmínky nebo omezení na tabulky a načítají se na každé stránce v systému. Script Include se načítá pouze, když je volán.

Následující příklady Business rule obsahují až 180 řádků kódu, což má velký vliv na výkonnost instance (čas načítání formulářů a seznamů).

- WAN_MSer_Copy SO number on task5
- W_I_M_report_BAN_audit
- W_I_M_SO id
- W_I_M_report_Site_delete
- W_I_M_report_WAN_update
- Oblast dopadu – výkon, ovladatelnost
- Kritičnost – vysoká
- Doporučení – vytvořit Script Include, vložit celý kód do oddělených funkcí a volat Script Include z Business Rule

4.3.2.2. Katalogové položky

Katalogová položka odkazuje na externí obsah

Katalogové položky odkazují na externí obsah pro generování problémů s daty, které mají být vyplněny do formulářů v ServiceNow. Odkazy navádějí uživatele k vznášení problémů e-mailem místo použití ServiceNow, jako nástroj workflow.

Příklad: „Pokud není správná Site v seznamu, požádejte oddělení Network Operations o její vytvoření přes e-mail netops.event@a.support.com. Prosím použijte následující šablonu.“ (odkaz)

- Oblast dopadu – použitelnost
- Kritičnost – nízká
- Doporučení – uvažovat o implementaci těchto podpůrných procesů jako o workflow v ServiceNow, nikoli o použití nestructurovaných způsobů komunikace, jako je e-mail.

Názvy proměnných v katalogových položkách

Proměnné nejsou vždy správně pojmenované, např. „WAN_check local contact“, „INT_check local contact“.

- Oblast dopadu – ovladatelnost
- Kritičnost – nízká
- Doporučení – sjednotit způsob pojmenování, např. používat jen podtržítka, nebo jen mezery, ne kombinaci obojího.

4.3.2.3. Client Script

Problémy s výkonem při spuštění synchronních rozhraní API

Client scripty s vlastními skripty používají synchronní GlideRecord API.

GlideRecord na straně klienta je synchronní volání na server, což způsobuje zpoždění a také vrací celý záznam, způsobující také zpoždění. Pokud se GlideRecord používá v Client Scriptu, zpoždění je znatelné uživateli, což vede ke špatné uživatelské zkušenosti z hlediska výkonu.

Příklady zmíněných Client Scriptů:

- WAN_modify site, preference, internet
- WAN_modify internet of secondary
- WAN_chck local cntct whn site changes

... a mnoho dalších.

- Oblast dopadu – výkonnost, použitelnost
- Kritičnost – vysoká

- Doporučení – nahradit GlideRecord asynchronním GlideAjax voláním za použití funkce zpětného volání

Špatný postup při pojmenování proměnných ve skriptu

V Client sciptech jsou používána jednotlivá písmena jako názvy proměnných např. var a, var b, var c. Díky tomu je administrace a porozumění kódu mnohem náročnější.

Příklady:

- WAN_Decommission request (popup)
- Show State and substate
- WAN_task9 checkbox mandatory
- Oblast dopadu – ovladatelnost
- Kritičnost – střední
- Doporučení – používat lépe popisující názvy proměnných, nebo alespoň celá slova

Používání Client Scriptů místo UI Policy

Požadované funkcionality mohou být zhotoveny pomocí UI Policy místo Client scriptů, které mají vyšší dopad na výkonnost.

Příklady Client sciptů, které mohou být nahrazeny:

- WAN_task otc,mrc mandarty by nxt step
- W_I_M_task read only variable
- LAN_task1 variables visible on task2
- WAN-Internet IDs on task 4, 5 & 9
- MSer_task mandatory variable on close
- Oblast dopadu – ovladatelnost
- Kritičnost – střední
- Doporučení – nahradit zmíněné Client Scripty za UI Policy

Manipulace s DOM elementy

Manipulaci s DOM elementy je doporučováno se vyvarovat, kvůli možným způsobeným problémům s udržitelností při aktualizaci prohlížečů.

V následujících příkladech bere kód hodnoty z DOM elementů a přidává je do textového pole za použití DOM selekce.

- LAN/WLAN_URL in variable help text
- MSer_task3&9 comnt,chng no visibility
- Oblast dopadu – ovladatelnost, škálovatelnost
- Kritičnost – nízká
- Doporučení – vytvořit Script Include, vložit celý kód do oddělených funkcí a volat Script Include z Business Rule

4.3.2.4. Platforma

Zastaralá CMS technologie použitá na Portálu

Zkoumané řešení nepoužívá současnou technologii Service Portálu, místo toho je použita zastaralá CMS technologie.

- Oblast dopadu – ovladatelnost, použitelnost
- Kritičnost – nízká
- Doporučení – vylepšit portál pro koncové uživatele pomocí modulu Service Portal

Vzhled portálových stránek

Portálové stránky využívají formuláře ze systémového rozhraní, což není příliš uživatelsky přívětivé.

- Oblast dopadu – použitelnost
- Kritičnost – střední
- Doporučení – vylepšit portál pro koncové uživatele pomocí modulu Service Portal s elementy stránek Service Portálu

Tabulka Service Order je vytvořena jako customizovaná

Tabulka “Service Order” [u_service_order] je vytvořena jako customizovaná tabulka, místo použití nativních funkcí pro zpracování požadavku (Request Fulfilment). Nejsou použita žádná workflow. Na této tabulce běží mnoho Business rule, které zpracovávají vlastní logiku pro čtení, zpracování, zápis nebo mazání dat do této a dalších souvisejících customizovaných tabulek a spouštějí se na základě různých manipulací s daty.

Jelikož je tato tabulka vytvořena jako základní tabulka, a ne jako rozšíření tabulky Task [task], nemohou být na ni použita workflow pro řízení životního cyklu požadavku.

Tabulka “Service Order” se zdá být jako klíčová tabulka pro implementovaný proces. Je tedy nepochopitelné, proč je tato tabulka implementována jako customizovaná, místo použití nativních funkcí.

- Oblast dopadu – ovladatelnost, škálovatelnost, upgradovatelnost, použitelnost
- Kritičnost – střední
- Doporučení – použít nativní tabulky pro zpracování požadavku (Request Fulfilment) a spouštět veškerou aplikační logiku pomocí workflow

Velké množství customizovaných tabulek

Většina celého řešení je založena na customizovaných tabulkách obsahujících procesní data. Nejlepším postupem je využívat nativní funkce, které mají podobné funkce, jaké vyžadují obchodní procesy, pouze s potřebnými konfiguracemi a přizpůsobením.

Tyto customizované tabulky se nacházejí v oblasti, kde na základě obsahu tabulky by uložené záznamy měly být:

{tabulka} -> {doporučená nativní tabulka/modul}

- Billing Account -> Contract Management
- Country list -> Country [core_country]
- Hemmersbach site -> Location
- Internet connection -> CMDB
- Internet IP Details -> CMDB
- Managed Service -> CMDB
- Network CMDB Audit -> Pravděpodobně nějaká pomocná tabulka pro zachování struktury CMDB – použít nativní nástroje pro řízení CMDB
- Network Connection Billing -> Cost Management, potenciálně Service Portfolio Management
- Region -> Location with parent/child
- SCOA Account Table -> Company, Cost Management, potenciálně Service Portfolio Management
- Service Order -> Request Fulfillment, Request/RITM ticket

- Site -> Location
 - Site all Connections -> Location, CMDB
 - Site Internet -> Location, CMDB
 - Site Managed Service -> Location, CMDB
 - Site User -> Location
 - Site WAN -> Location, CMDB
 - Unit Code -> pravděpodobně Department
 - WAN Connection -> CMDB, Cost Management
- Oblast dopadu – ovladatelnost, škálovatelnost, upgradovatelnost, použitelnost
 - Kritičnost – střední
 - Doporučení – vylepšit celé řešení využitím nativních funkcí co nejvíce, tak jak je uvedeno výše

Customizované výsledky průzkumu

Vyvinutá customizovaná aplikace “Survey response”, která zaznamenává výsledky průzkumů, místo využití nativních funkcí “Survey”.

- Oblast dopadu – ovladatelnost, upgradovatelnost
- Kritičnost – nízká
- Doporučení – využívat nativní funkce pro výsledky průzkumů - “Survey”

Špatná developerská praxe pro zveřejnění nových verzí

Byly zaznamenány indexy, které vedly k podezření, že vývoj byl v několika případech proveden přímo na produkční instanci a byl přeskočen celý proces řízení zveřejňování z developerské a testovací instance.

Z dostupných dat lze vyčíst, že testovací instance byla naklonována z produkční instance v polovině června 2018 a objekt byl vytvořen dříve ve výchozím setu updatů, který je nepřenosný mezi instancemi. To dokazuje, že vývoj byl proveden na produkční instanci.

Taková praxe představuje hrozbu, neboť může způsobit okamžité nežádoucí účinky, jako je například nepoužitelnost nástrojů nebo ztráta dat.

- Oblast dopadu – error, ovladatelnost
- Kritičnost – vysoká

- Doporučení – nikdy nevyvíjet na produkci. Definovat silnou a vynutitelnou vývojovou politiku.

4.3.2.5. Notifikace

Pokročilé podmínky mohou být vytvořeny v nástroji pro tvorbu podmínek

Použití skriptů se nevyžaduje, protože jeho funkce lze dosáhnout pomocí nástroje pro tvorbu pokročilých podmínek.

Příklady:

- End User Service Request Approval Rec
- End User ServiceRequestDenialReceived
- End User Service Request Approval
- Oblast dopadu – ovladatelnost, výkonnost
- Kritičnost – nízká
- Doporučení – používat nástroj pro tvorbu pokročilých podmínek, místo skriptu

Script pro e-mailovou notifikaci není zapsán do funkce

Předdefinované funkce a kódy by měly být vždy součástí vlastního vývoje. Jinak to může ovlivnit další funkce, protože skript vytvoří globální proměnnou nazvanou "gr" pro GlideRecord(). Toto jméno proměnné je obecně nejčastěji používáno pro objekty GlideRecord.

Příklady:

- NetOps_End User_Signature
- WAN_Approval requested for_CC
- WAN_Network CMDB Approvers_CC
- WAN_Service Catalog_CC
- Oblast dopadu – error, výkonnost
- Kritičnost – vysoká
- Doporučení – vždy zahrnovat out-of-the-box předvyplněný kód pro volání samočinné funkce. Upravit stávající e-mailové skripty pro notifikace s tímto problémem a zabalit kód do samočinné funkce. Smazání zbytečného kódu v poznámkách.

4.3.2.6. Script Include

Žádný popis funkcí

Funkce postrádají popis funkčnosti a použití skriptu, což ztěžuje správu a podporu řešení.

Pojmenování proměnných je buď příliš jednoduché (použití jednotlivých písmen) nebo není vypovídající.

Příklady:

- WANcheckInactive
- WANciClassName
- WANsetVariables
- WANsiteInactive

- Oblast dopadu – ovladatelnost
- Kritičnost – střední
- Doporučení – Popsat funkčnost každé funkce, vstupní parametry a jejich očekávaný typ dat, nejlépe přidat název objektu, který volá tuto funkci.

4.3.2.7. UI Policies

Výkonnostní problémy při běžícím synchronním API

UI Policies, na kterých běží customizovaný skript používají synchronní GlideRecord API.

GlideRecord na straně klienta je synchronní volání na server, což způsobuje zpoždění a také vrací celý záznam, způsobující také zpoždění. Pokud se GlideRecord používá v Client Skriptu, zpoždění je znatelné uživateli, což vede ke špatné uživatelské zkušenosti z hlediska výkonu.

Příklad: UI Policy “AC_WAN_MSer_Task 8 editable”

... a mnoho dalších.

- Oblast dopadu – výkonost, použitelnost
- Kritičnost – vysoká
- Doporučení – nahradit GlideRecord asynchronním GlideAjax voláním za použití funkce zpětného volání

UI Policies spouštějí mnoho formulářových akcí

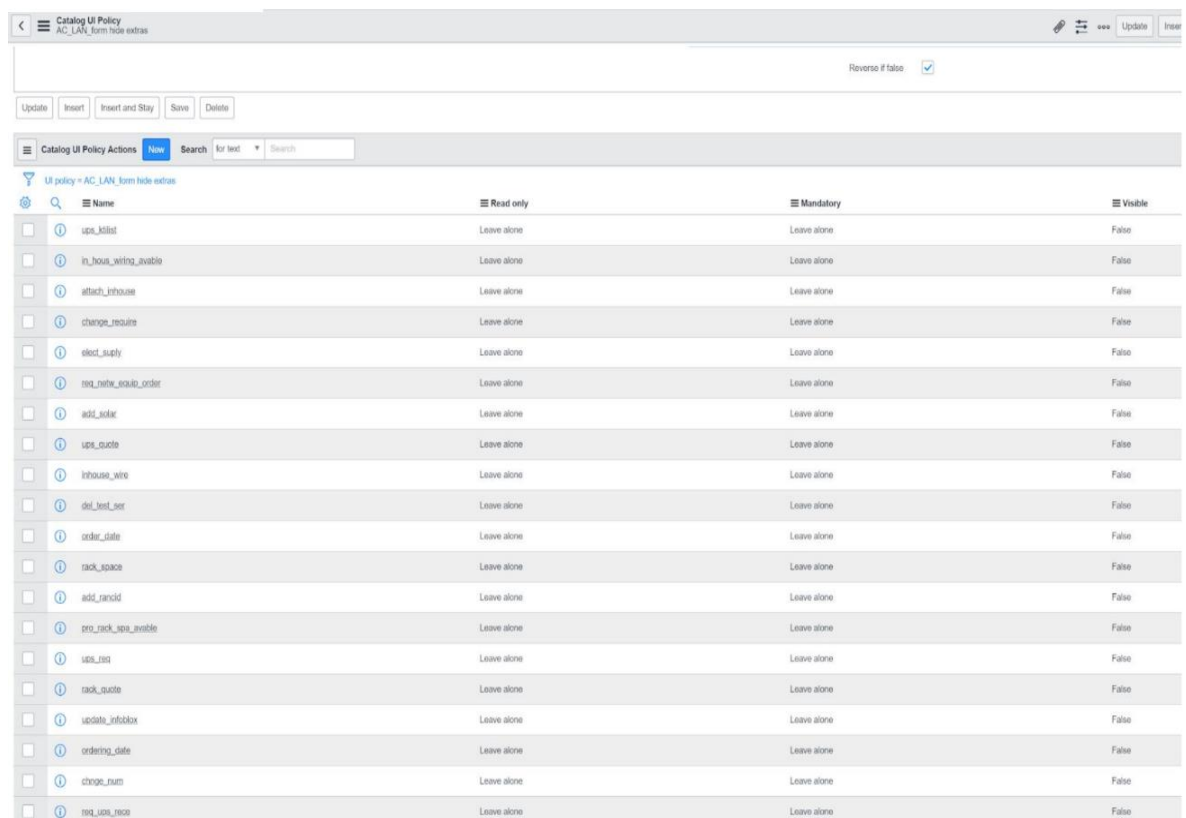
UI Policies skrývají a zobrazují mnoho polí formuláře, což vede k problémům s výkonem na straně klienta při načítání dokumentu se všemi poli v době vykreslování a následném skrytí polí.

Pro uživatele to představuje vnímání horšího výkonu a špatný uživatelský dojem, jelikož uživatelé vidí nejprve celý dokument s následným opětovným vykreslením, přičemž skrývá pole jeden po druhém Client scriptem spuštěným v prohlížeči.

Příklad: Catalog UI Policy “LAN_form hide extras”

... a mnoho dalších.

Obrázek 12: Příklad velkého množství nastavených UI Policies



Name	Read only	Mandatory	Visible
ups_klist	Leave alone	Leave alone	False
in_house_wiring_available	Leave alone	Leave alone	False
attach_inhouse	Leave alone	Leave alone	False
change_requires	Leave alone	Leave alone	False
select_supply	Leave alone	Leave alone	False
req_netw_order_order	Leave alone	Leave alone	False
add_rack	Leave alone	Leave alone	False
ups_quote	Leave alone	Leave alone	False
inhouse_wire	Leave alone	Leave alone	False
del_test_ser	Leave alone	Leave alone	False
order_date	Leave alone	Leave alone	False
rack_space	Leave alone	Leave alone	False
add_rackid	Leave alone	Leave alone	False
pre_rack_ups_install	Leave alone	Leave alone	False
ups_req	Leave alone	Leave alone	False
rack_quote	Leave alone	Leave alone	False
update_infoblox	Leave alone	Leave alone	False
ordering_date	Leave alone	Leave alone	False
choose_num	Leave alone	Leave alone	False
req_ups_req	Leave alone	Leave alone	False

zdroj: ServiceNow instance společnosti A

- Oblast dopadu – ovladatelnost, výkonnost, použitelnost
- Kritičnost – vysoká
- Doporučení – vzít v úvahu použití formulářových náhledů (Form Views)

Pro stejná pole je definováno několik katalogových UI Policy akcí

Pro některá pole existuje více katalogových UI Policy, které mají stejné pořadí, nelze tedy předvídat jejich pořadí spuštění. To má vliv na výkonnost na straně klienta.

Příklad takových polí: pri_int, wan_reject, mser_site, pri_con, ser_type a 50+ dalších.

- Oblast dopadu – ovladatelnost, výkonnost, škálovatelnost
- Kritičnost – střední
- Doporučení – podrobněji definovat pravidla a podmínky nebo změnit pořadí tak, aby každá UI Policy byla jedinečná.

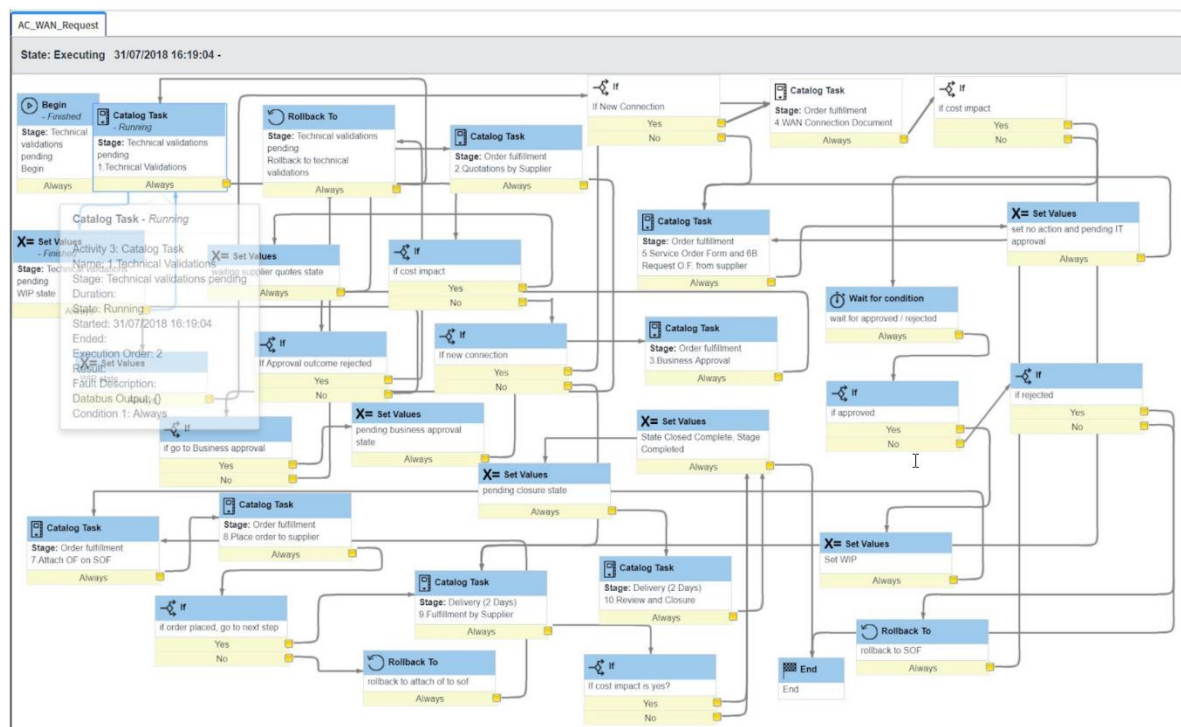
4.3.2.8. Workflow

Uspořádání workflow

Pracovní postupy mají neformátované uspořádání, takže je obtížné, ne-li nemožné jim porozumět, řešit problémy či dále vyvíjet a sledovat stav ticketu.

Příklad: Workflow “WAN_Request”

Obrázek 13: Příklad nepřehledného workflow



zdroj: ServiceNow instance společnosti A.

- Oblast dopadu – ovladatelnost
- Kritičnost – střední
- Doporučení – zlepšit uspořádání workflow

Mnoho katalogových úloh ve workflow

Pracovní postupy generují řadu katalogových úloh, které má řešitel splnit. Jedna osoba musí uzavřít více úkolů, aby hlavní ticket pokračoval v toku.

- Oblast dopadu – použitelnost
- Kritičnost – střední
- Doporučení – přezkoumat tok obchodních procesů spolu s technickým workflow, aby bylo možné vidět příležitosti ke zjednodušení

4.3.3. Shrnutí analýzy

Řešení je implementováno se zaměřením na poskytnutí očekávaných funkcí, ale s velmi nízkým uvážením nákladů na budoucí možné změny nebo na maximalizaci výkonu ovlivňujícího koncové uživatele. Využití stávajících funkcí ServiceNow (které jsou k dispozici po dlouhou dobu) je překvapivě nízké. Je zde několik problémů, které, pokud se nebudou řešit, se zhorší. Může být dosaženo významných zlepšení.

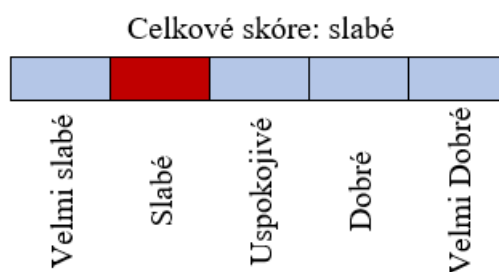
Při implementaci bylo použito mnoho customizovaného vývoje, který mohl být proveden za použití out-of-the-box funkcí. Výdaje na takový vývoj je mnohem vyšší, jelikož out-of-the-box funkce jsou rozkódovány a překonfigurovány. Také jakékoliv dodatečné vylepšení pro out-of-the-box funkce přidané ServiceNow v budoucnosti nemohou být použity, protože jsou využity customizované tabulky a kódy.

Otázka nákladů na budoucí upgrade je velkým problémem. Vlastní vývoj by mohl být způsob, jak se vyhnout konfliktům v průběhu upgradu, ale pouze pokud se provádí v rámci jednotlivých aplikací, které chrání vytvořené objekty, což zde není použito. Velké množství vlastních objektů mimo rámec aplikací znamená další riziko během upgradu. Pokud existuje strategie pro vývoj aplikace pomocí vlastního kódu pro minimalizaci možného dopadu upgradů, měla by být přehodnocena, protože náklady na tento vývoj jsou mnohem vyšší než náklady na upgrade, zejména u tak vyspělých oblastí, jako je request fulfilment.

Výkonnost platformy je těžce ovlivněna nedodržováním ServiceNow best practice a nezohlednění výkonu během vývoje. Tento špatný výkon neovlivňuje pouze serverovou stranu ServiceNow, ale také uživatelskou zkušenost.

Na základě analýzy je doporučeno znovu naimplementovat řešení od začátku pro společnost E. Přidaná hodnota bude v oblastech se zvýšeným potenciálem řídit vývoj obchodních procesů spolu se zjednodušením procesů technické implementace, zlepšením výkonu, snížením nákladů na údržbu a náklady na budoucí změny.

Tabulka 2: Celkové skóre



zdroj: [16]

Tabulka 3: Celkový přehled kritičnosti

Kritičnost	
Vysoká	7
Střední	10
Nízká	8

zdroj: [16]

4.4. Návrhy řešení

Následující návrhy vycházejí z výsledků výše uvedené technické analýzy. Společnost D nabízí dva možné scénáře.

4.4.1. Scénář A – Převod řešení do nové instance, oprava chyb a vylepšení

Scénář A zahrnuje přenesení řešení společnosti A do nové instance zákazníka E, opravu chyb a konfiguraci vylepšení. Zákazník musí rozhodnout, zda opravy a vylepšení budou provedeny před převodem na novou instanci, z čehož by mohla čerpat také společnost A a náklady na toto vylepšení by si tak společnosti mohly rozdělit. Tento scénář také zahrnuje opravu všech chyb s nízkou, střední i vysokou kritičností uvedených v technické analýze.

Scénář A obsahuje následující tři kroky:

1. Oprava chyb nalezených v ServiceNow platformě společnosti A

Tento krok zahrnuje opravu všech nalezených chyb, včetně opravy kódu naimplementované logiky pomocí best practices pro zlepšení efektivnosti a snížení zbytečné složitosti původního řešení. Dále zahrnuje reverzní inženýrství a novou dokumentaci celého řešení a také nasazení řešení napříč oběma prostředími společností A i E.

2. Převod řešení z instance společnosti A do instance společnosti E zahrnující opravy nástroje Network Request Management

V tomto kroku dochází k nasazení připraveného nástroje Network Request Management z kroku č. 1, dále přesun dat z tabulek instance společnosti A do instance společnosti E, validaci nasazené konfigurace na instanci společnosti E a samozřejmě opravu případných errorů.

3. Vylepšení současného řešení ServiceNow na základě požadavků zákazníka

V posledním kroku tohoto scénáře budou implementovány katalogové položky dle požadavků, definované role a zodpovědnosti a samozřejmě nastavená přístupová práva konkrétním rolím (uživatelům). Dále je samozřejmostí dodání související dokumentace, uživatelské testování, případné opravy chyb nalezených testováním a krátké období podpory ze strany dodavatele.

4.4.2. Scénář B – Nové řešení “na zelené louce”

Druhou navrženou variantou je implementace celého procesu tzv. „na zelené louce“, což znamená vývoj zmíněného procesu od začátku přímo na instanci ServiceNow společnosti E, včetně vyvarování se nalezených chyb a implementaci požadovaných vylepšení.

Tento scénář obsahuje následující kroky definující, jakým způsobem bude scénář probíhat:

1. Přezkoumání obchodních procesů a požadavků k jejich přizpůsobení pro společnost E
2. Solution design – použití méně customizace a více out-of-the box funkcionalit pro zlepšení administrace a aktualizace
3. Definování user stories – popisování funkcionalit aplikace z pohledu konkrétních uživatelů
4. Implementace nového řešení bez veškerých nalezených chyb v technické analýze
 - a. Refaktoring zavedené obchodní logiky s využitím best practices pro zlepšení efektivity
 - b. Zachování přístupových práv a omezení rolí
 - c. Snížení zbytečné složitosti implementovaného řešení
 - d. Nastavení stejných notifikací jako v instanci společnosti A, avšak podle best practices
 - e. Re-implementace současného zastaralého CMS portálu do ServiceNow out-of-the-box řešení – Service Portal
5. Převod souvisejících dat z tabulek v instanci společnosti A do instance společnosti E
6. Technická dokumentace – dobře strukturovaná dokumentace, která usnadňuje správu a podporu implementovaného nástroje
7. Příprava testovacích scénářů pro uživatele
8. Nasazení, oprava chyb a období podpory – možné dřívější nasazení procesu díky implementaci od začátku

4.5. Finanční odhady pro jednotlivé scénáře

V následujících dvou podkapitolách jsou uvedeny cenové nabídky pro společnost E na základě navrhovaných scénářů. Ceny jsou rozděleny dle nároků na zaměstnance společnosti D.

Na projektu zpravidla pracuje ServiceNow Consultant (junior), ServiceNow Expert (senior), Architect (principal) a Project manager (lead). Pro každého z uvedených zaměstnanců je uveden odhadovaný počet man days (MDs), které představují jeden pracovní den daného člověka, a cena za jeden tento den u daného zaměstnance.

4.5.1. Scénář A

Při stanovení cenové nabídky pro scénář A byly vzaty v úvahu následující předpoklady:

- Společnost A bude chtít profitovat z provedených vylepšení a to znamená, že
 - zažádá současného poskytovatele o klon produkční instance na developerskou a testovací před začátkem fixace,
 - připraví regresní testy, aby ověřila, že fixace neporušila současné funkce,
 - provede tyto regresní testy.
- Pokud společnost A nebude chtít profitovat z fixace chyb, celé řešení bude nejprve přeneseno na instanci společnosti E a opravy budou prováděny až tam.
- Společnost E bude nasazovat řešení mezi instancemi (developerská, testovací, produkční), pokud však společnosti A či E budou toto požadovat od současného poskytovatele, nabídka bude levnější o 1 MD na konzultanta.

Pro tento scénář je cena rozdělena do tří kroků, které byly definovány v kapitole 5.1. Celková odhadovaná cena je 60 850 EUR s celkovým počtem 102.7 MDs.

Tabulka 4: Finanční náklady na scénář A

Steps	ServiceNow Consultant		ServiceNow Expert (Senior Consultant)		Architect (Principal Consultant)		Project Manager (Lead Consultant)	
	MDs	Price [€500/MD]	MDs	Price [€600/MD]	MDs	Price [€1000/MD]	MDs	Price [€800/MD]
Step 1	38.4	19 200	15.6	9360	2.4	2400	10.8	8640
Total [€]/MDs of Step 1	39 600 EUR for 67.2 MDs							
Step 2	14.5	7 250	7	4 200	0.5	500	4.5	3 600
Total [€]/MDs of Step 2	15 550 EUR for 26.5 MDs							
Step 3	5	2 500	1	600	1	1 000	2	1 600
Total [€]/MDs of Step 3	5 700 EUR for 9 MDS							
TOTAL [€]	60 850 EUR for 102.7 MDs							

zdroj: [17]

4.5.2. Scénář B

Stanovení cenové nabídky pro scénář B se řídí následujícími předpoklady:

- Společnost E požádá současného poskytovatele o klon produkční instance na developerskou a testovací před začátkem nového vývoje ze strany společnosti D.
- Společnost D bude provádět nasazení řešení mezi instancemi, pokud však toto společnost E požaduje od současného poskytovatele, cena bude levnější od 1 MD na konzultanta.

Celková odhadovaná cena za scénář B je 54 230 EUR s celkovým počtem 90.7 MDs. Tento scénář je tedy levnější než scénář A a zároveň společností D doporučený.

To je také důvod, proč si **společnost E zvolila scénář B.**

Tabulka 5: Finanční náklady na scénář B

Deliverable / Role	ServiceNow Consultant		ServiceNow Expert (Senior Consultant)		Architect (Principal Consultant)		Project Manager (Lead Consultant)	
	MDs	Price [€500/MD]	MDs	Price [€600/MD]	MDs	Price [€1000/MD]	MDs	Price [€800/MD]
New implementation	51.7	25 850	18.8	11 280	4.7	4 700	15.5	12 400
TOTAL [€]	54 230 EUR for 90.7 MDs							

zdroj: [17]

4.6. Požadavky zákazníka

4.6.1. Katalogové položky

Společnost A doposud využívala 4 katalogové položky pro objednávání síťových služeb, přičemž každá služba představovala konkrétní položku v systému:

- Request for LAN / WLAN Service
- Request for Internet Connection
- Request for WAN Connection
- Request for Managed Service

Vzhledem k tomu, že požadavky na Internet Connection, WAN Connection a Managed Service využívají podobný postup při jejich zpracování, požaduje společnost E sloučení těchto tří položek do jedné s názvem **Managed Service Request** a vytvoření jednotného workflow pro tento proces.

Položka Request for LAN / WLAN Service využívá odlišný proces, nicméně požadavek společnosti E je přejmenovat tuto položku na **Network Infrastructure Request** a vytvořit workflow pro tuto položku.

Katalogové položky byly také velmi složité a obsahovaly spoustu proměnných, které musel žadatel, případně poté řešitel vyplňovat. Průměrně každá položka obsahovala 40-50 proměnných a dále různý počet sad proměnných, což způsobilo velkou nepřehlednost. Zákazník tedy požaduje zpřehlednění a zredukování počet proměnných na nezbytný. Proměnné, které mají být využity, byly definovány.

4.6.2. Workflow

Ve společnosti A byly implementovány pracovní postupy k jednotlivým položkám (Příloha č. 1–4), které však byly příliš složité a nepřehledné. Požadavek zákazníka je takový, aby pro sloučené položky Managed Service Request bylo vytvořeno jednotné workflow a druhé pro položku Network Infrastructure Request, které bylo také nevyhovující.

Původní workflow byla vytvořena na bázi katalogových tasků, které vytvářely pod úkoly pro daný požadavek a každý task řešila určitá odpovědná osoba. Požadavek zákazníka bylo celkově zjednodušit a zpřehlednit tato workflow. Společnost D tedy zákazníkovi

doporučila vypustit katalogové tasky a pouze daný požadavek přiřazovat v určitých fázích odpovědné řešitelské skupině.

Pro položku **Network Infrastructure Request** bylo navrženo nové workflow (Příloha č. 5) s definovanými fázemi:

1. Design & Validation
2. Quotation
3. IT Approval
4. Procurement
5. Delivery
6. Closed

Pro položku **Managed Service Request** bylo navrženo nové workflow (Příloha č. 6) s těmito definovanými fázemi:

1. Preparation
2. Quotation
3. IT Approval
4. Documentation
5. Final Approval
6. Place Order
7. Delivery
8. Closed

4.6.3. Možnost zrušení požadavku

Zákazník požaduje možnost zrušení požadavku během jeho celého životního cyklu. Tento požadavek se ze systému nesmaže, jeho stav se však změní na Closed Incomplete a v systému zůstane viditelný. Tato možnost by měla být přístupná pouze řešitelům požadavku. Zároveň se má odeslat notifikace zadavateli, že jeho požadavek byl zrušen.

4.6.4. Schvalovací proces

Každý požadavek musí být schválen odpovědnou osobou. Zákazník má definovány skupiny řešitelů a úrovně schvalování, přičemž každá úroveň odpovídá cenové hranici.

Obě nové katalogové položky mají definované schvalovací procesy a jejich úrovně rozdílně. Požadavek zákazníka je implementovat logiku schvalování dle následující specifikace:

- Položka **Network Infrastructure Request** má pouze IT schvalování, přičemž při celkové ceně služby do 5000 EUR včetně, se požadavek přiřazuje pouze první úrovni schvalovatelů a pokud celková cena služby přesahuje cenu 5000 EUR, požadavek se po schválení na první úrovni posouvá ještě ke druhé úrovni schvalovatelů.
- Položka **Managed Service Request** má kromě IT schvalování ještě finální schvalování. V tomto případě se v IT schvalování nebere v potaz cena, pouze technická specifikace, a proto je zde pouze jedna úroveň schvalovatelů. Ti však mají mít možnost požadavek nejen schválit či zamítnout, ale také vrátit na začátek k přepracování. Ve finálním schvalování již hraje roli cena, kdy každá úroveň má svoji schvalovací skupinu a daný požadavek se přiřazuje konkrétní skupině dle následující tabulky.

Tabulka 6: Úrovně schvalování

Cena (EUR)	Skupina
<= 5000	první úroveň schvalovatelů
>5000<=30000	druhá úroveň schvalovatelů
>30000	třetí úroveň schvalovatelů

zdroj: vlastní

Zároveň zákazník požaduje, aby se po 7denní neaktivitě skupiny schvalovatelů, kterým je požadavek momentálně přiřazen, zaslalo e-mailové upozornění příslušné skupině, že požadavek čeká na její schválení.

4.6.5. Předpokládané doručení

Zákazník chce mít možnost zadat do pole „Estimated delivery“ datum předpokládaného doručení, a to kdykoliv před fází Delivery u obou položek. Zadání data by mělo být

povinné právě ve fázi, která předchází fázi Delivery. Poté již toto pole nebude možné měnit.

4.6.6. Průzkum spokojenosti

Zákazník požaduje, aby se po uzavření požadavku odeslal žadateli e-mail s odkazem na dotazník spokojenosti. Tento dotazník by měl obsahovat hodnocení pomocí emotikon, přičemž jejich bodové ohodnocení je na škále od 1 (velmi nespokojen) do 5 (velmi spokojen) a také pole s možností dodatečných komentářů.

4.7. Implementace optimalizovaného řešení

4.7.1. Katalogové položky

Dle požadavků zákazníka byly vytvořeny dvě katalogové položky, které společně tvoří kategorii s názvem “Network Connectivity”. Veškerá konfigurace byla při implementaci omezena pouze na danou kategorii, aby nedošlo k ovlivnění jiných aplikací a konfigurací, které již systém obsahuje.

Pro každou katalogovou položku byl vytvořen set proměnných, které se zobrazují na katalogovém formuláři a dále další proměnné, které se průběžně zobrazují při zpracovávání požadavku. Každá proměnná má unikátní název, který se dle best practices píše malými písmeny s podtržítka a to proto, aby mohl být snadno použit v případě nutnosti skriptování (např. `eit_site`). Koncovému uživateli se pak zobrazuje tzv. štítek, který je více uživatelsky přívětivý (např. Site).

4.7.2. Network Infrastructure Request

Set proměnných s názvem „EIT – NIR Variable set“ obsahuje následující proměnné s daným typem, který určuje, jak se bude proměnná zobrazovat, zda vyžaduje volbu z možností, či zda odkazuje na jinou tabulku, ze které je třeba vybrat záznam atd.

- **Site (`eit_site`)** – referenční pole, které vyžaduje výběr z tabulky `u_site`, kde jsou uloženy všechny Site. Zadavatel vybere ze seznamu danou Site, pro kterou chce požadavek vytvořit. Toto pole je povinné, tedy označené červenou hvězdičkou.
- **Local contact (`eit_local_contact`)** – referenční pole, které odkazuje na tabulku `sys_user`, kde jsou uloženi všichni uživatelé systému. Pro toto pole se vybírá

primární kontakt, který souvisí se zvolenou Site. Pro snazší zadávání požadavku byl vytvořen Client script, který automaticky vyplňuje toto pole příslušným kontaktem poté, co zadavatel vybere požadovanou Site. Toto pole je povinné, tedy označené červenou hvězdičkou.

- **Procurement (eit_nir_procurement)** – zaškrťovací políčko, které zadavatel zaklikne v případě, že je vyžadován nákup nové přípojky. V případě, že se jedná pouze o úpravu současné přípojky, políčko nechá prázdné. V závislosti na volbě tohoto pole dochází k rozdílnému postupu při zpracování požadavku.
- **Requested by Date (eit_requested_by_date)** – políčko pro výběr požadovaného data doručení. Pro toto pole byl vytvořen Client script, který hlídá, aby zadavatel nevybral datum v minulosti. V případě, že tak učiní, pod políčkem se zobrazí chybová hláška a pole zůstane prázdné.
- **Requested type (eit_requested_type)** – pole s možností výběru, pro kterou službu je požadavek vytvářen. Možnosti výběru jsou LAN, Wireless LAN či Both (oboje). Toto pole je povinné, tedy označené červenou hvězdičkou.
- **Proposed Solution (eit_proposed_solution)** – textové pole, do kterého zadavatel zadá navrhované řešení. Toto pole je povinné, tedy označené červenou hvězdičkou.
- **Comments (eit_more_comments)** – textové pole, kam zadavatel může zadat další dodatečné informace či komentáře.

Tyto proměnné mají nastavené pořadí, ve kterém se mají na formuláři zobrazovat a pomocné proměnné, které oddělují rozložení do sloupců. Výsledný katalogový formulář vypadá následovně.

Obrázek 14: Výsledný katalogový formulář pro Network Infrastructure Request

Service Catalog > Network Connectivity > Network Infrastructure Request

Service Request for LAN/WLAN or both

* Site

* Local contact

► More information

Procurement

* Proposed Solution

Comments

Requested by Date

* Requested type

-- None --

Order this Item

Delivery time 1 Day

Order Now

zdroj: ServiceNow instance společnosti E.

Další proměnné, které se zobrazují v průběhu zpracování požadavku budou detailněji vysvětleny během popisu procesu. Jejich názvy a typy jsou následující:

- Validated (eit_validated) – zaškrťovací pole
- Quote (eit_quote_number) – jednořádkové textové pole
- Have you attached the Contract in the Related list? (eit_quoted) – pole s možností výběru
- Procured (eit_procured) – zaškrťovací pole
- Delivered On (eit_delivered_on) – datumové pole
- Completed (eit_completed) – datumové pole

4.7.3. Managed Service Request

Pro tuto položku byl vytvořen set proměnných s názvem “EIT – MS Variable set”, které se zobrazují na formuláři. Některé z nich jsou totožné s proměnnými pro první položku pouze s rozdílem unikátního názvu. Tyto položky jsou následující:

- Site (eit_ms_site)
- Local contact (eit_ms_local_contact)
- Requested by date (eit_ms_req_by_date)
- Requested Type (eit_ms_type) - zde je rozdíl v možnostech výběru, kdy zadavatel požadavku vybírá WAN Connection, Internet Connection nebo Managed Service
- Proposed Solution (eit_ms_proposed_solution)

- Comments (eit_ms_comments)

Další položky zahrnuté do tohoto setu proměnných se zobrazují v závislosti na výběru možností v poli Requested Type a jsou následující:

- **Action (eit_ms_action)** – pole s možností výběru New nebo Modify, přičemž zadavatel vybírá, zda se jedná o novou službu, nebo změnu stávající. Toto pole je povinné.
- **Reason for new connection (eit_ms_reason_connect)** – textové pole, kde zadavatel specifikuje důvod nového připojení. Toto pole je povinné.
- **Reason for new Managed Service (eit_ms_reason_ms)** – textové pole, kde zadavatel specifikuje důvod nové služby Managed Service. Toto pole je povinné.
- **Internet Connection (eit_ms_int_connection)** – referenční pole odkazující na tabulku současných internetových služeb, ze kterých zadavatel vybírá v případě, že chce upravovat existující připojení. Toto pole je povinné.
- **WAN Connection (eit_ms_wan_connection)** - referenční pole odkazující na tabulku současných WAN služeb, ze kterých zadavatel vybírá v případě, že chce upravovat existující připojení. Toto pole je povinné.
- **Managed Service (eit_ms_managed_service)** - referenční pole odkazující na tabulku současných Managed Service služeb, ze kterých zadavatel vybírá v případě, že chce upravovat existující službu. Toto pole je povinné.
- **Service Type (eit_ms_service_type)** – pole s možností výběru ze 14 možností, ze kterých zadavatel vybírá požadovaný typ služby v případě volby Managed Service. Toto pole je povinné.
- **Requested modification type (eit_ms_req_type)** – pole s možností výběru ze 4 možností (Upgrade, Downgrade, Decommission, Other). Toto pole je povinné.
- **Please specify (eit_ms_pls_specify)** – textové pole, kam zadavatel specifikuje, o jakou se jedná službu či typ, v případě možnosti Other. Toto pole je povinné.

Pro lepší přehlednost, kdy se které pole má zobrazit, byla vytvořena následující tabulka kombinací polí a jejich příslušných akcí. Tato logika byla nastavena pomocí UI Policies.

Tabulka 7: Kombinace polí a jejich akce

Pole	Vybrané možnosti	Akce
Requested Type + Action	WAN Connection + New	Objeví se nové pole "Reason for new connection" a políčko Cost Impact se automaticky zaškrtně.
Requested Type + Action	Internet Connection + New	Objeví se nové pole "Reason for new connection" a políčko Cost Impact se automaticky zaškrtně.
Requested Type + Action	Managed Services + New	Objeví se nová pole "Reason for new Managed Service" a také "Service Type" a políčko Cost Impact se automaticky zaškrtně.
Requested Type + Action	WAN Connection + Modify	Objeví se referenční pole "WAN Connection", pro výběr stávajícího připojení a dále se objeví pole "Requested modification type".
Requested Type + Action	Internet Connection + Modify	Objeví se referenční pole "Internet Connection", pro výběr stávajícího připojení a dále se objeví pole "Requested modification type".
Requested Type + Action	Managed Services + Modify	Objeví se referenční pole "Managed Service", pro výběr stávající služby a dále se objeví pole "Service Type". V tomto případě je pole Service Type automaticky doplněno na základě vybrané stávající služby a není možné ho měnit.
Requested modification type	Other	Objeví se nové pole "Please specify".
Service Type	Other	Objeví se nové pole "Please specify".

zdroj: vlastní

Mimo set proměnných byly definovány další proměnné, které se zobrazují v průběhu zpracování požadavku. Níže jsou popsány jejich typy, detailněji budou vysvětleny během popisu procesu.

- RFP sent (eit_ms_rfp_sent) – zaškrťovací políčko
- Quote (eit_ms_quote_number) – textové pole
- Have you attached the Contract in the Related list? (eit_ms_quoted) – výběr z možností
- Documentation created (eit_ms_doc_created) – zaškrťovací políčko
- Order Placed (eit_ms_order_placed) – zaškrťovací políčko

- Delivered on (eit_ms_delivered_on) – datumové pole
- Completed (eit_ms_completed) – datumové pole

Stejně jako předchozí u předchozí položky, i tady mají proměnné nastavené pořadí, ve kterém se mají na formuláři zobrazovat a pomocné proměnné, které oddělují rozložení do sloupců. Výsledný katalogový formulář v jeho základním zobrazení vypadá následovně. Další pole se zobrazují dle výše uvedené tabulky.

Obrázek 15: Výsledný katalogový formulář pro Managed Service Request

zdroj: ServiceNow instance společnosti E.

4.7.4. Requested Item

Jakmile zadavatel vyplní a odešle formulář ke zpracování, vytvoří se požadavek s unikátním číslem. Tento požadavek se nazývá „Requested Item“ a číslo vždy začíná předvolbou RITM. Na formuláři jsou zobrazeny základní informace o daném požadavku, např. jméno zadavatele, datum vytvoření, skupina, které je požadavek přiřazen, v jaké je požadavek fázi atd.

Dále jsou na formuláři viditelné veškeré údaje vyplněné zadavatelem při tvorbě požadavku. Tyto údaje jsou nastaveny pouze ke čtení, aby je nebylo možné měnit.

Na následujících obrázcích jsou znázorněny formuláře pro každou z položek s vyplněnými testovacími údaji. Oba požadavky jsou v první fázi.

Obrázek 16: Formulář požadavku pro Network Infrastructure Request

Requested Item
RITM9001824 [Network Request view]

Update Cancel Request Save Delete ↑ ↓

Number	RITM9001824	Stage	Design & Validation
Requested for	[User]	State	Work in Progress
Item	Network Infrastructure Request	Opened	2019-01-02 12:52:21
Assignment group	EIT - NRM WL Net Specialists	Opened by	[User]
		Estimated Delivery	[Date]
Short description	Service Request for LAN/WLAN or both		

Variables

* Site	TUR008	Requested by Date	[Date]
* Local contact	[User]	* Requested type	Wireless LAN
Procurement	<input type="checkbox"/>		
* Proposed Solution	test		
Comments	[Text Area]		
Validated	<input type="checkbox"/>		

zdroj: ServiceNow instance společnosti E.

Obrázek 17: Formulář požadavku pro Managed Service Request

The screenshot displays the ServiceNow interface for a Managed Service Request. The main form contains the following fields:

- Number: RITM9001580
- Requested for: Request user 1 ManagedService
- Item: Managed Services Request
- Assignment group: EIT - NRM MS Order Admins
- Stage: Preparation
- State: Work in Progress
- Opened: 2018-12-10 09:39:46
- Opened by: Request user 1 ManagedService
- Estimated Delivery: (empty)
- Short description: Service Request for a new / modify WAN / Internet Connection or Managed Service

Below the main form is a 'Variables' section with the following fields:

- * Requested by date: 2019-02-20 09:39:44
- * Site: SWE046
- * Local contact: (empty)
- * Requested Type: Managed Services
- * Action: Modify
- * Managed Service: NMS10000068
- * Service Type: Other
- * Please specify: test
- * Proposed Solution: test
- Comments: (empty)
- RFP sent

zdroj: ServiceNow instance společnosti E.

4.7.5. Workflow

Vytvořením požadavku se spouští workflow neboli pracovní postup, který určuje, jak se má požadavek zpracovat a provedením příslušné akce ho posouvá do další fáze. Jedná se především o automatické nastavování skupin řešitelů, akce, při kterých se čeká na zadání hodnoty do příslušného pole, či schvalovací akce. Nově vytvořená workflow pro obě položky (Příloha č. 7 a 8) již nepoužívají katalogové tasky. Požadavek je posouván do další fáze zadáním požadované hodnoty do příslušného políčka a přiřazuje se skupině řešitelů. V následujícím textu je popsán postup pro danou katalogovou položku a akce, které v jednotlivých fázích nastávají.

4.7.5.1. Network Infrastructure Request

1. Design & Validation

Do této fáze se požadavek automaticky dostane v okamžiku jeho vytvoření a přiřadí se skupině Network Specialists, které přijde upozornění prostřednictvím e-mailové zprávy, že jí by požadavek přiřazen a je vyžadováno jeho vyřešení. Na formuláři požadavku se v této fázi zobrazuje pole „Validated“, které řešitel zaškrtně, jakmile jsou provedeny příslušné kroky k posunutí požadavku do další fáze. Uložení změn se požadavek posouvá do fáze Quotation.

2. Quotation

V této fázi zůstává požadavek přiřazen skupině Network Specialists, která zde vyplní formulář kontraktu v záložce „Related list“. Kontrakt obsahuje informace o poskytovateli služby, o jakou se jedná Site, jaká je cena této služby a další související informace. Jakmile je kontrakt vyplněn a přiřazen k požadavku, řešitel vyplní číslo kvóty do pole „Quote number“ a potvrdí volbou „Yes“ na políčku „Have you attached the Contract in the Related list?“, že byl kontrakt vyplněn a přiřazen k požadavku. K jednomu požadavku může být přiřazeno více kontraktů. Políčko Validated z předchozí fáze je pomocí UI Policy nastaveno pouze ke čtení, aby již nebylo možné ho zpětně upravovat.

3. IT Approval

Ve schvalovací fázi se požadavek přiřazuje první skupině schvalovatelů, která představuje první úroveň, na kterou se požadavek dostává vždy. Workflow generuje schvalovací požadavek a systém odešle zprávu členům příslušné skupiny. Pro schválení požadavku stačí, když je schválen pouze jedním ze členů skupiny, to samé platí pro zamítnutí.

V případě schválení požadavku na první úrovni dochází ke kontrole cen zadaných v kontraktech. Pokud součet všech cen kontraktů přesahuje částku 5000 EUR, požadavek se přiřazuje druhé skupině schvalovatelů na druhou úroveň. Tam schvalovací proces probíhá stejným způsobem, jako v první úrovni schvalování. V případě, že tato částka je nižší, než 5000 EUR, požadavek se posouvá do další fáze. Pokud jeden ze schvalovatelů (ať už v první, nebo druhé úrovni) požadavek zamítne, automaticky se uzavírá jako nekompletní a ukončuje workflow.

Schválení / zamítnutí lze provést dvěma způsoby, a to buď přímo kliknutím na odkaz v e-mailové zprávě, nebo v systému u příslušného požadavku.

Zároveň je ve workflow nastavena aktivita, která po 7 dnech zkontroluje, zda byl požadavek schválen či nikoliv. Pokud nebyl, systém odešle e-mailovou notifikaci aktuální skupině schvalovatelů s informací, že požadavek čeká na jejich schválení. Pokud byl požadavek schválen, workflow pokračuje dále.

4. Procurement

Tato fáze nastává pouze v případě, že bylo zaškrtnuto políčko „Procurement“ při zadávání požadavku. Skupina, které je požadavek přiřazen v této fázi, je znovu Network Specialists, která je opět upozorněna e-mailovou zprávou. Na formuláři se zobrazuje pole „Procured“, které je třeba zaškrtnout pro posunutí do další fáze.

5. Delivery

Ve fázi doručování probíhá dodání služby ze strany dodavatele na příslušné Site. Na formuláři se zobrazují dvě datová pole, jedno pro zadání data doručení (Delivered) a druhé pro zadání data kompletnosti a uzavření požadavku (Completed).

6. Closed

Do této fáze se požadavek dostane v okamžiku jeho uzavření, a to buď formou řádného splnění procesu, kdy se požadavek uzavře se stavem “Closed Complete”, nebo formou stornování či zamítnutí požadavky, kdy jeho stav “Closed Incomplete”. Všechna pole na formuláři jsou nyní pouze ke čtení a nelze je již upravovat.

4.7.5.2. Managed Service Request

1. Preparation

Toto je první fáze požadavku pro Managed Service Request, která se automaticky nastaví při vytvoření požadavku. Skupina řešitelů pro tuto položku je „Order Admins“ a v této fázi je jim požadavek opět automaticky přiřazen a odesílá se jim také e-mailová notifikace. Na formuláři se objevuje políčko „RFP sent“, které musí jeden z řešitelské skupiny zaškrtnout pro posunutí požadavku do další fáze.

2. Quotation

Ve druhé fázi požadavku probíhá stejná akce jako v případě položky Network Infrastructure Request, tedy řešitelé ze skupiny „Order Admins“ vyplní kontrakt a číslo kontraktu na formuláři. Stejně tak musí vybrat možnost „Yes“ pro potvrzení, že byl kontrakt vytvořen a přiložen k požadavku. Pomocí UI Policy je políčko z předchozí fáze nastaveno pouze ke čtení.

3. IT Approval

Ve fázi IT schvalování dochází k vytvoření schvalovacího požadavku pro skupinu „IT Approvers“, kterým je odeslána e-mailová notifikace. Ti mají možnost požadavek buď schválit, zamítnout anebo vrátit k přepracování.

Pokud jeden ze schvalovatelů požadavek schválí, posouvá se do další fáze. Jestliže vybere možnost zamítnutí, požadavek se uzavírá jako nekompletní a workflow končí. Pokud však zvolí možnost „Refine“, požadavek se vrací do fáze Preparation, workflow se resetuje a začíná běžet od začátku.

I pro toto schvalování je nastaven časovač jako v případě položky Network Infrastructure Request, který funguje na stejném principu.

4. Documentation

V této fázi je požadavek opět přiřazen skupině řešitelů „Order Admins“ včetně odeslání e-mailové notifikace. Na formuláři se objevuje políčko „Documentation created“, které je třeba opět zaškrtnout pro posunutí požadavku do další fáze. Veškerá dříve vyplněná pole již není možné upravovat. Jakmile jsou provedeny potřebné kroky a řešitel toto pole zaškrtně, následuje fáze Final Approval.

5. Final Approval

Předtím, než je požadavek posunut do této fáze, workflow pomocí skriptu určí, které skupině schvalovatelů má požadavek přiřadit na základě ceny zadané ve fázi Quotation. Jak bylo definováno zákazníkem, při ceně do 5000 EUR včetně je požadavek přiřazen první skupině schvalovatelů, při ceně v rozmezí 5000 EUR a 30 000 EUR včetně je

požadavek přiřazen druhé skupině schvalovatelů a při ceně vyšší než 30 000 EUR se požadavek přiřadí třetí skupině schvalovatelů.

V této schvalovací fázi již lze požadavek pouze schválit či zamítnout, opět stačí, když požadovanou akci provede pouze jeden člen z příslušné skupiny. Schválením se požadavek posouvá do další fáze, zamítnutím se požadavek uzavírá jako nekompletní a workflow končí.

Stejně jako v ostatních schvalovacích fázích, i zde funguje aktivita, která hlídá schválení požadavku a případně zasílá upozornění o čekajícím požadavku.

6. Place Order

Požadavek se přiřazuje zpět skupině „Order Admins“, kteří v této fázi zaškrtnou nově zobrazené pole „Order placed“, aby mohl být požadavek dále zpracován. Samozřejmostí je nemožnost upravovat dříve vyplněná pole. Jakmile řešitel provede příslušné kroky, postupuje požadavek do fáze Delivery.

7. Delivery

Stejně jako pro první položku, i zde se zobrazují dvě datumová pole – Delivered a Completed. Pro správné zpracování a uzavření požadavku je nutné vyplnit obě tato pole.

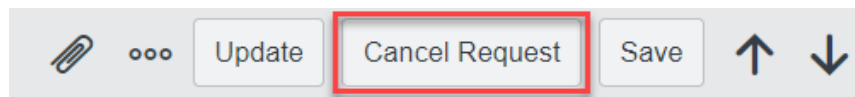
8. Closed

I tato fáze je stejná, jako pro první položku, tedy stav požadavku je buď „Closed Complete“ nebo „Closed Incomplete“ a všechna pole jsou nastavena pouze ke čtení a nelze je tedy měnit.

4.7.6. Zrušení požadavku

Na základě požadavku zákazníka, bylo vytvořeno tlačítko na formuláři pomocí UI Action, které má název „Cancel Request“ a umožňuje skupině řešitelů stornovat jakýkoliv požadavek z kategorie Network Connectivity, který je v jakékoliv fázi s výjimkou fáze Closed. Kliknutím na toto tlačítko se systém uživatele zeptá, zda si opravdu přeje zrušit požadavek. Výběrem možnosti OK se požadavek posune do fáze Closed se stavem Closed Incomplete a workflow se zruší.

Obrázek 18: Tlačítko pro zrušení požadavku



zdroj: ServiceNow instance společnosti E.

Obrázek 19: Okno pro zrušení požadavku

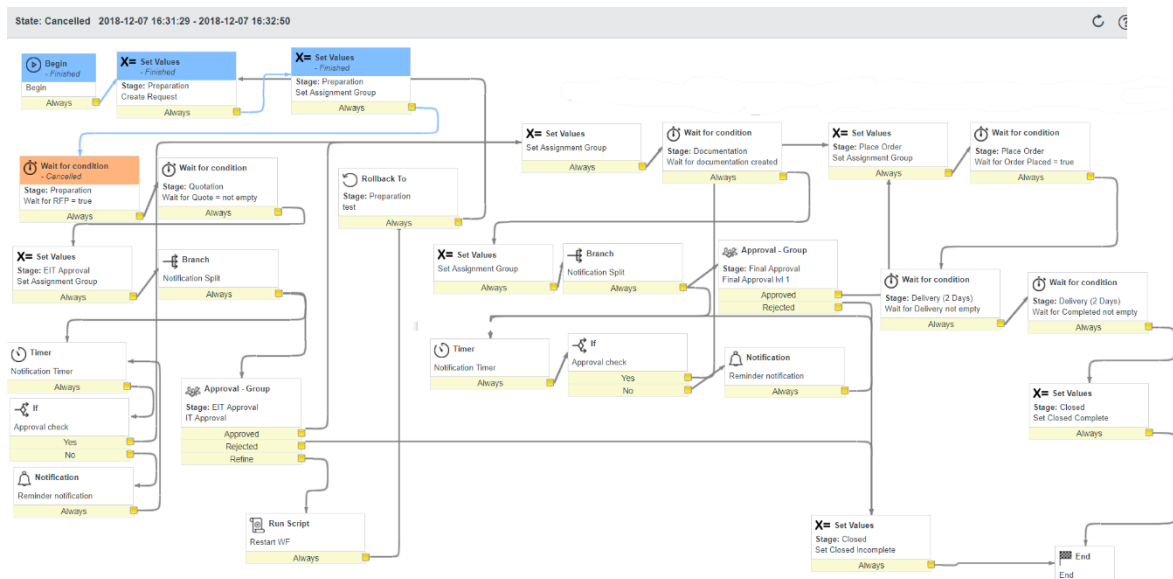
Web ██████.service-now.com říká

Click 'OK' to confirm cancellation of this request, else click 'Cancel' to abort the cancellation.



zdroj: ServiceNow instance společnosti E.

Obrázek 20: Stornované workflow v případě zrušení požadavku

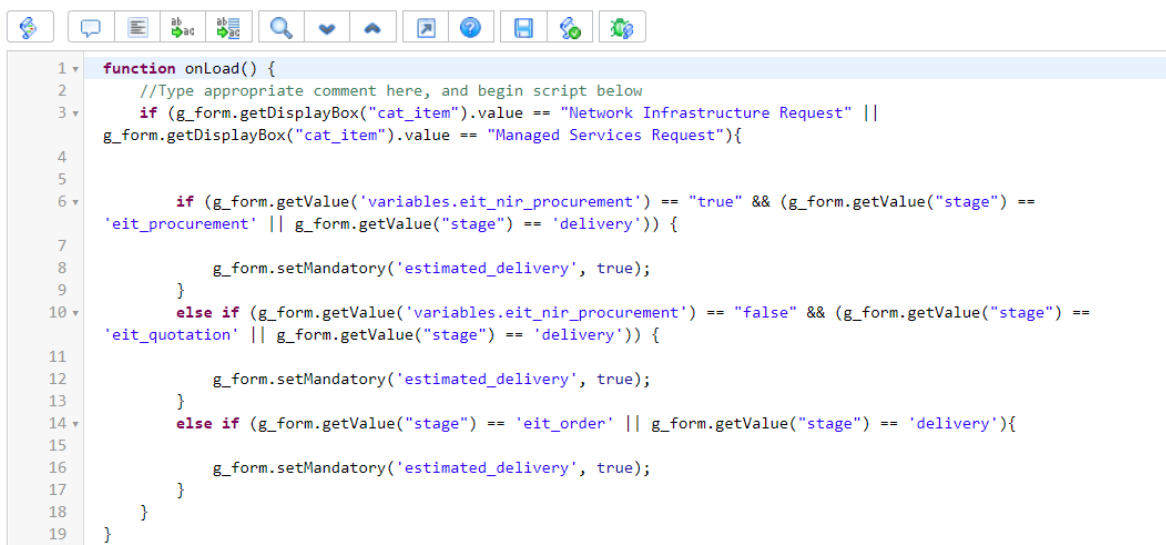


zdroj: ServiceNow instance společnosti E.

4.7.7. Předpokládané doručení

Pro políčko s názvem „Estimated delivery“ byl vytvořen onLoad() Client script, který pro položku **Network Infrastructure Request** nastavuje toto pole povinné v případě, že je ve fázi Procurement (pokud tato možnost byla vybrána při tvorbě požadavku) anebo ve fázi Quotation. Pro položku **Managed Service Request** nastavuje toto pole povinné ve fázi Place Order.

Obrázek 21: Client Script pro pole Estimated Delivery



```
1 function onLoad() {
2     //Type appropriate comment here, and begin script below
3     if (g_form.getDisplayBox("cat_item").value == "Network Infrastructure Request" ||
4         g_form.getDisplayBox("cat_item").value == "Managed Services Request"){
5
6         if (g_form.getValue('variables.eit_nir_procurement') == "true" && (g_form.getValue("stage") ==
7             'eit_procurement' || g_form.getValue("stage") == 'delivery')) {
8             g_form.setMandatory('estimated_delivery', true);
9         }
10        else if (g_form.getValue('variables.eit_nir_procurement') == "false" && (g_form.getValue("stage") ==
11            'eit_quotation' || g_form.getValue("stage") == 'delivery')) {
12            g_form.setMandatory('estimated_delivery', true);
13        }
14        else if (g_form.getValue("stage") == 'eit_order' || g_form.getValue("stage") == 'delivery'){
15            g_form.setMandatory('estimated_delivery', true);
16        }
17    }
18 }
19 }
```

zdroj: ServiceNow instance společnosti E.

4.7.8. Průzkum spokojenosti

Pro potřeby dotazníkového průzkumu spokojenosti byla vytvořena šablona s emotikony se škálou od 1 do 5 a zároveň možnost dopsat textové ohodnocení tak, jak bylo požadováno zákazníkem. Tento dotazník se zasílá zadavateli požadavku při jeho řádném dokončení prostřednictvím e-mailové zprávy, která obsahuje odkaz na příslušný dotazník a také informaci, o jaký požadavek se jedná. Odpovědi těchto dotazníků se ukládají nejen do tabulky odpovědí, ale také byl vytvořen report, který zobrazuje odpovědi pomocí grafu.

Obrázek 22: Průzkum spokojenosti

Short Customer Satisfaction Survey using Smiley faces

How satisfied are you with the provided service?

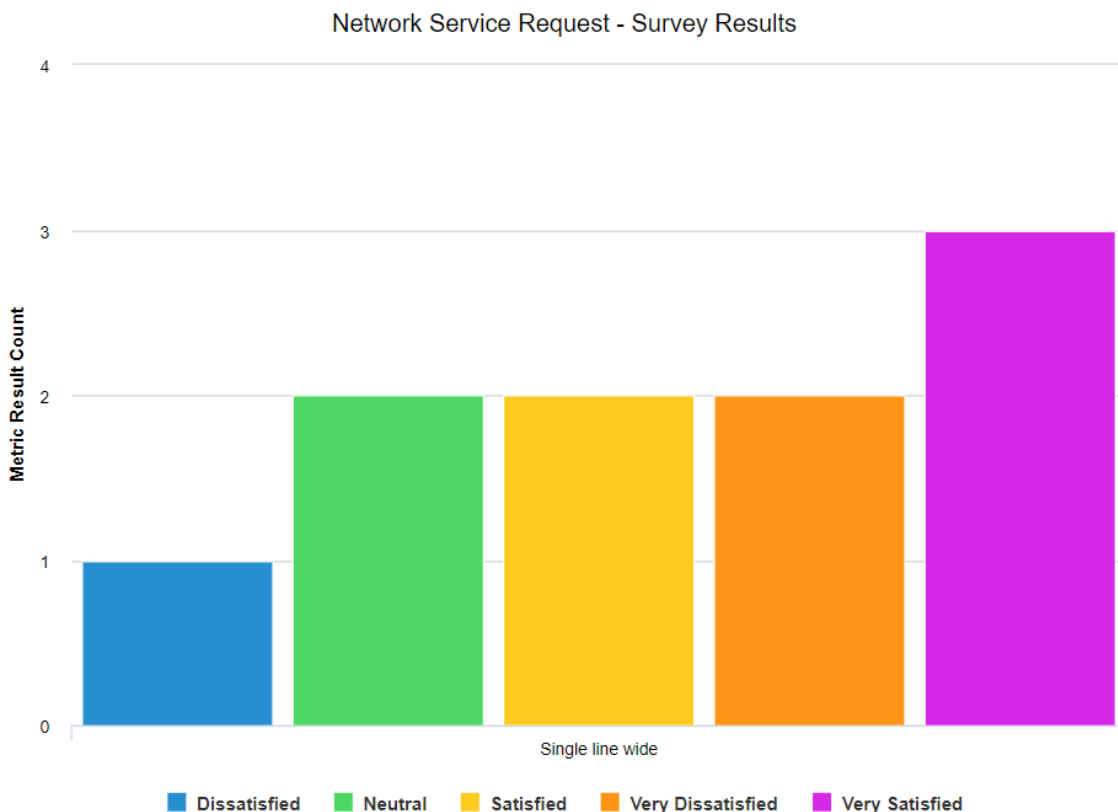
Very Dissatisfied Dissatisfied Neutral Satisfied Very Satisfied

Please provide us your experience with a comment.

Submit Save Cancel

zdroj: ServiceNow instance společnosti E.

Obrázek 23: Graf výsledků dotazníkového průzkumu spokojenosti



zdroj: ServiceNow instance společnosti E.

5. Vyhodnocení finančních nákladů projektu

Během implementace nového procesu narazila společnost D na několik problémů, které způsobily navýšení původních navrhovaných nákladů o 14 500 EUR (27,5 MDs). V níže uvedené tabulce jsou tyto náklady rozepsány. V těchto dodatečných nákladech je uvedena

také položka Hyper Care, což znamená podpora ze strany dodavatele během prvního týdne od nasazení finálního řešení do produkční instance společnosti E.

Tabulka 8: Dodatečné náklady na vývoj

Deliverable / Role	Service Consultant		Service Expert (Senior Consultant)		Project Manager		Hyper Care	
	MDs	Price [€500/MD]	MDs	Price[€600/MD]	MDs	Price[€800/MD]	MDs	Price[€500/MD]
	15	7500	3	1800	1,5	1200	8	4000
	10 500 EUR						4000 EUR	
TOTAL	14 500 EUR							

zdroj: [17]

Mezi hlavní důvody, které způsobily navýšení nákladů patří zejména:

- Komplikace v rámci první synchronizace se současným poskytovatelem ohledně vývoje. Společnost D očekávala vývoj na čistém systému, bohužel docházelo ke konfliktům při úpravách konfigurace se současným poskytovatelem.
- Společnosti D nebyly zajištěny přístupy do produkční instance, a tudíž bylo složitější kontrolovat nasazený vývoj a jeho chování.
- Během vývoje bylo nutné upravit původní datový model, což způsobilo větší časové náklady.
- Dodaná data z produkční instance od původního dodavatele nebyly v očekávané kvalitě. Bylo nutné data upravovat, filtrovat a čistit, aby byla použitelná pro potřeby testování. Oproti dat z testovací instance, obsahují tato data jinak formátované záznamy.

Společnost E tedy za kompletně nové řešení zaplatila **68 730 EUR** v délce trvání **118.2 MDs**.

6. Závěr

Cílem diplomové práce bylo na základě vymezených teoretických východisek a provedené technické analýzy navrhnout optimální řešení pro správu požadavků na síťové připojení v systému ServiceNow a na základě sběru požadavků zákazníka vybrané optimální řešení implementovat.

Analýzou stávajícího řešení byly zjištěny zásadní nedostatky, které měly mimo jiné velký vliv na výkonnost platformy a možnost jejího bezproblémového upgradu. Ve stávajícím řešení bylo využito příliš mnoho vlastního vývoje, který lze snadno nahradit využitím nativních funkcí systému a tím tak systém znatelně „odlehčit“. Velkým problémem bylo také zjištění, že doporučené best practices nebyly dodržovány, a proto nebylo řešení optimální a systém dostatečně efektivní.

Společnosti E tedy byly navrženy dva způsoby, kterými lze dosáhnout optimalizace. První zahrnoval opravu veškerých zjištěných nedostatků na původním řešení a převedení opraveného řešení do nového prostředí, z čehož by mohla profitovat i společnost A. Druhou navrženou variantou byla implementace „na zelené louce“, tedy vývoj přímo na instanci společnosti E. Společnost si tedy vybrala variantu E, která byla zároveň i méně nákladná.

Na základě shromážděných požadavků zákazníka pak bylo optimalizované řešení implementováno za využití ServiceNow best practices, pomocí konfigurací uvedených v teoretické části a samozřejmě bez nalezených nedostatků z technické analýzy. Proces byl celkově zjednodušen a automatizován pomocí optimalizovaných workflow, které jsou přiloženy v příloze této práce.

V této práci je popsána pouze část z celého projektu a to ta, kterou jsem reálně vypracovala sama. Celý projekt zahrnuje mnoho dalších aktivit, ať už se jedná o datovou strukturu implementovaného procesu, migraci veškerých dat společnosti, nastavení přístupových práv, tvorbu souvisejících konfigurací, anebo třeba samotný release na produkční instanci.

Finální finanční náklady na projekt jsou oproti odhadované částce vyšší o 14 500 EUR z důvodu komplikací s původním poskytovatelem, komplikací s migrací dat a dalších důvodů uvedených kapitole 11.

7. Seznam použitých zdrojů

1. ŠILEROVÁ, Edita a Klára HENNYEYOVÁ. Informační systémy v podnikové praxi. První vydání. Praha: Powerprint, 2016. ISBN 978-80-87994-78-8
2. Wikisofia, *Hierarchie Data -> Informace -> Znalost* [online], 2013. [cit. 2019-03-20], Dostupné z WWW:
https://wikisofia.cz/index.php/Hierarchie_Data_%E2%86%92_Informace_%E2%86%92_Znalost
3. Andromedia, *Systém* [online], 2008 [cit. 2019-03-20], Dostupné z WWW:
<http://www.andromedia.cz/andragogicky-slovník/system>
4. BUCKSTEEG, Martin. *ITIL 2011*. Brno: Computer Press, 2012. ISBN 978-80-251-3732-1
5. BMC, *ITIL Processes & Best practices* [online], 2019. [cit. 2019-03-09], Dostupné z WWW: <https://www.bmc.com/guides/itil-introduction.html>
6. System Online, *Historie a základní principy cloud computingu* [online], 2015. [cit. 2019-03-10], Dostupné z WWW:
<https://www.systemonline.cz/virtualizace/historie-a-zakladni-principy-cloud-computingu.htm>
7. VELTE, Anthony T, Toby J VELTE a Robert C ELSENPETER. *Cloud Computing: praktický průvodce*. Přeložil Jakub GONER. Brno: Computer Press, 2011. ISBN 978-80-251-3333-0
8. ServiceNow, *All products and solutions* [online], 2018. [cit. 2019-01-19]. Dostupné z WWW: <https://www.servicenow.com/products-by-category.html>
9. ServiceNow Documentation, *Business Rules* [online], 2018. [cit. 2019-01-20], Dostupné z WWW: https://docs.servicenow.com/bundle/london-application-development/page/script/business-rules/concept/c_BusinessRules.html
10. ServiceNow Documentation, *How Business Rules work* [online], 2018. [cit. 2019-01-20], Dostupné z WWW: <https://docs.servicenow.com/bundle/london->

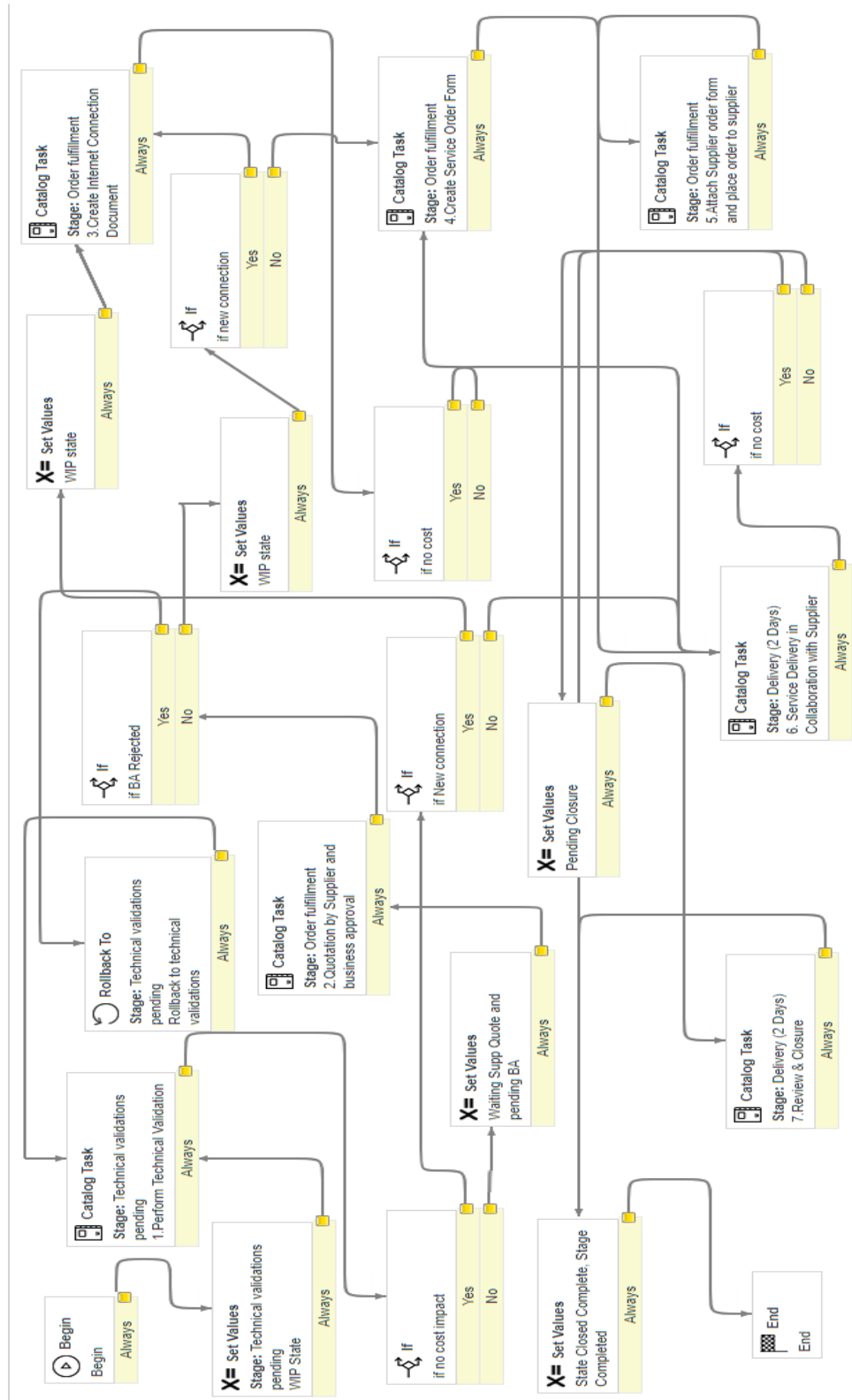
application-development/page/script/business-rules/reference/r_HowBusinessRulesWork.html

11. ServiceNow Documentation, *UI Policies* [online], 2018. [cit. 2019-01-21],
Dostupné z WWW: https://docs.servicenow.com/bundle/london-platform-administration/page/administer/form-administration/task/t_CreateAUIPolicy.html
12. ServiceNow Documentation, *UI Actions* [online], 2018. [cit. 2019-01-21],
Dostupné z WWW: https://docs.servicenow.com/bundle/london-platform-administration/page/administer/list-administration/concept/c_UIActions.html
13. ServiceNow Documentation, *Client Scripts* [online], 2018. [cit. 2019-02-02],
Dostupné z WWW: <https://docs.servicenow.com/bundle/london-application-development/page/script/client-scripts/concept/client-scripts.html>
14. ServiceNow Documentation, *Script Includes* [online], 2018. [cit. 2019-02-02],
Dostupné z WWW: https://docs.servicenow.com/bundle/london-application-development/page/script/server-scripting/concept/c_ScriptIncludes.html
15. ServiceNow Developer, *Technical Best Practices* [online], 2018. [cit. 2019-02-03],
Dostupné z WWW:
https://developer.servicenow.com/app.do#!/catlist/technical_best_practices?v=london
16. Network Service Request, *Technical Review*, interní dokument společnosti D, 2018
17. Network Request management offer, interní dokument společnosti D, 2018

8. Přílohy

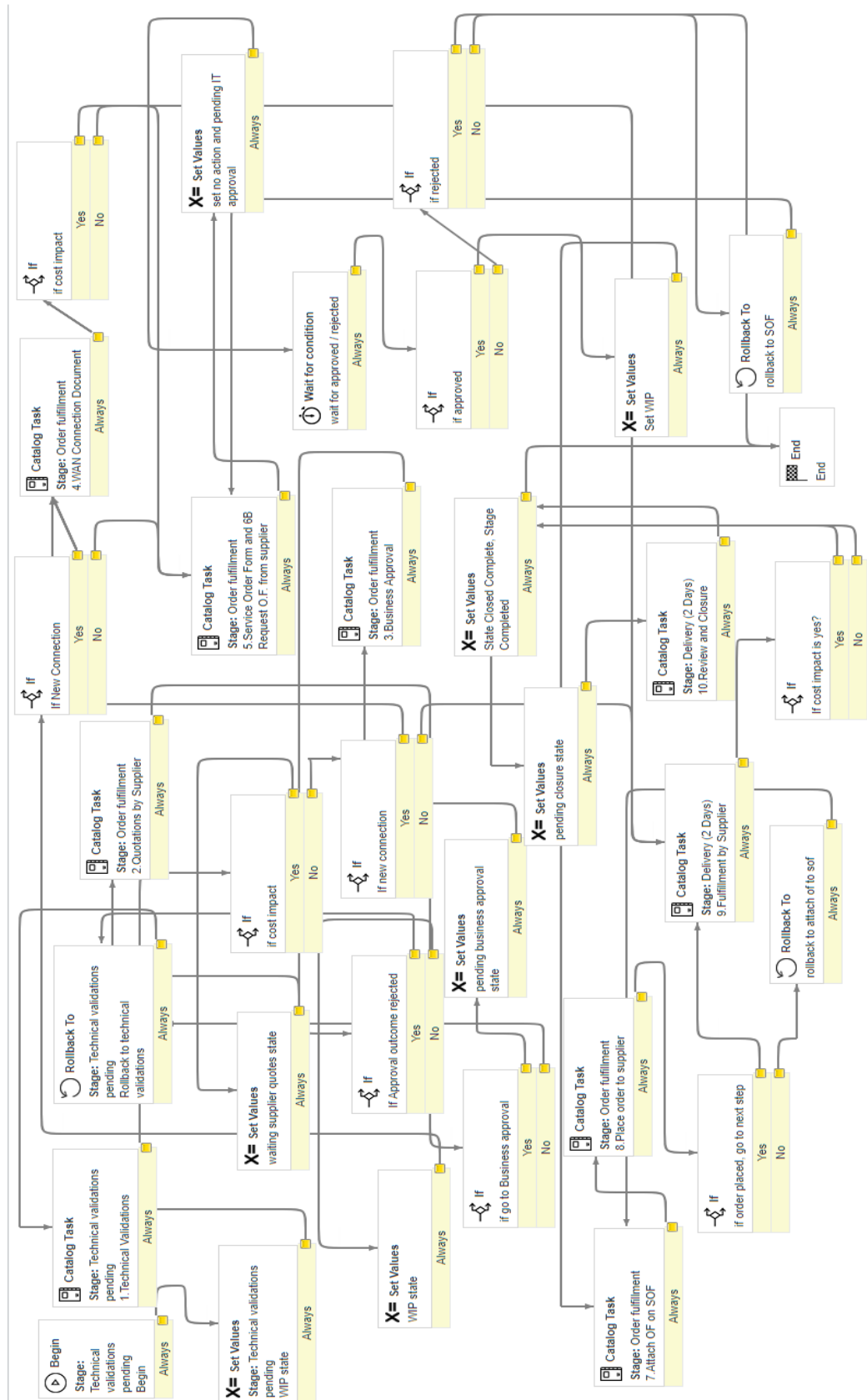
Příloha 1: Workflow pro Internet Connection (Společnost A)

WORKFLOW PRO INTERNET CONNECTION (SPOLEČNOST A)



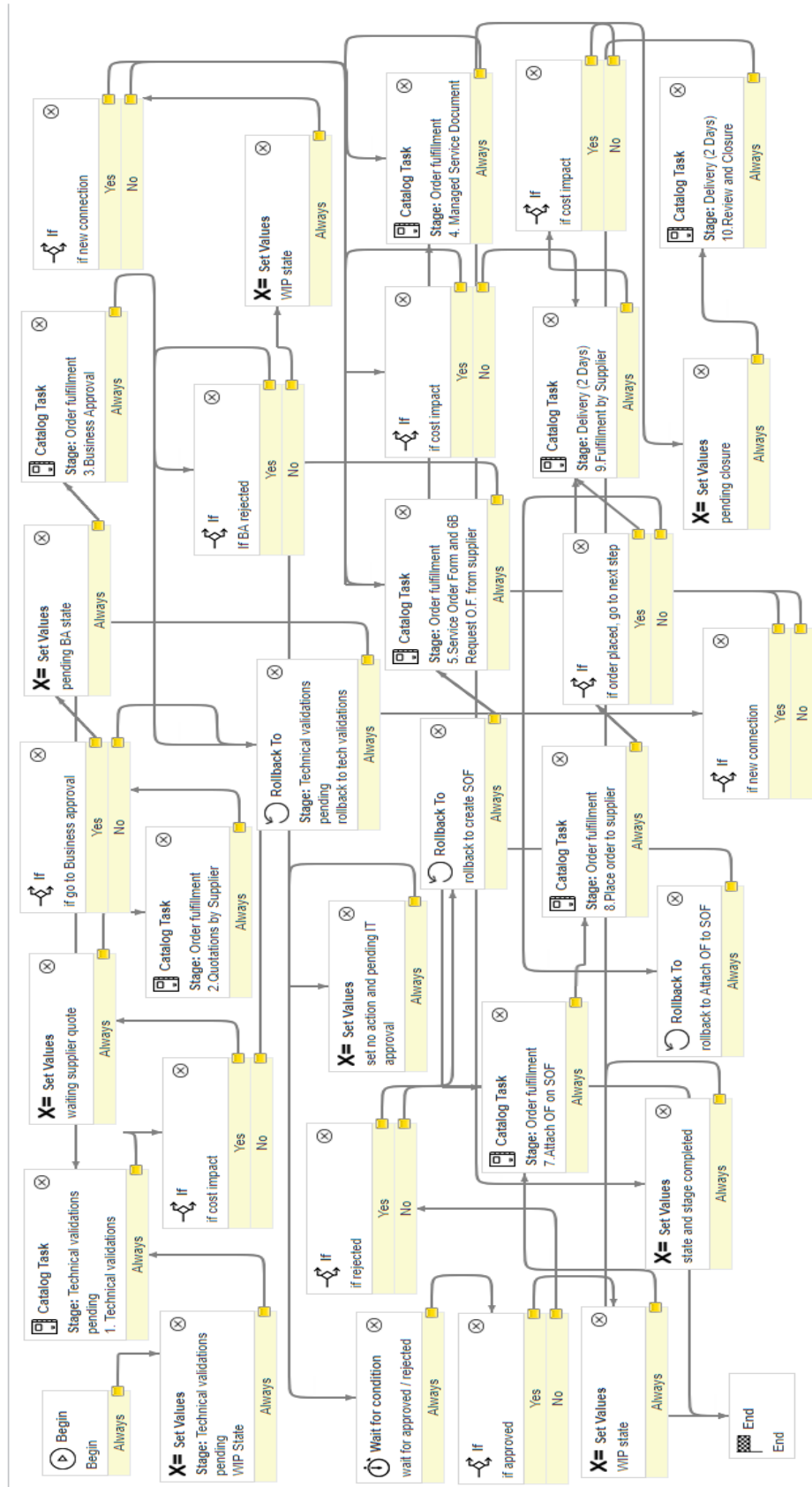
Příloha 2: Workflow pro WAN Request (Společnost A)

WORKFLOW PRO WAN REQUEST (SPOLEČNOST A)



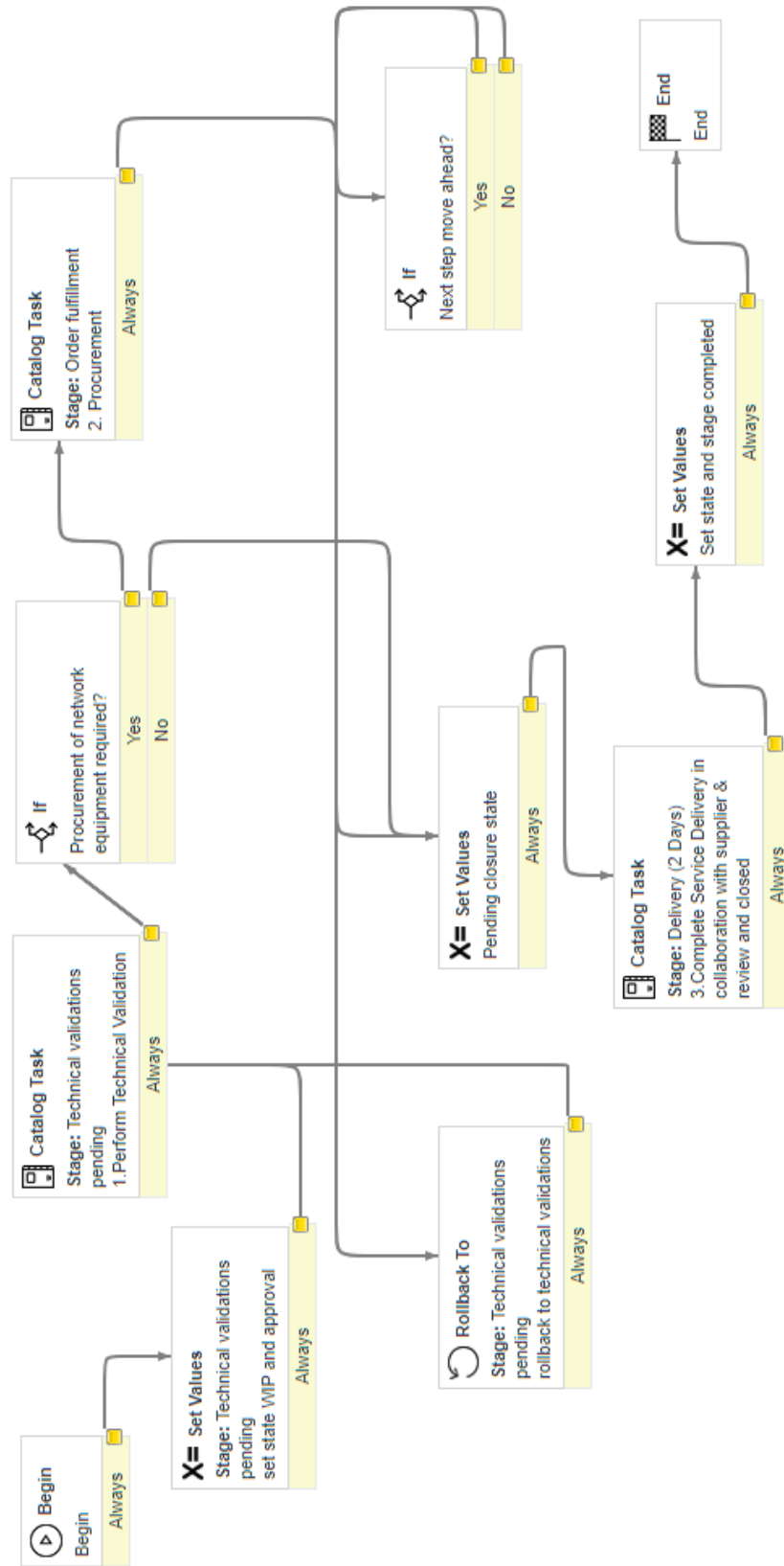
Příloha 3: Workflow pro Managed Service Request (Společnost A)

WORKFLOW PRO MANAGED SERVICE REQUEST (SPOLEČNOST A)

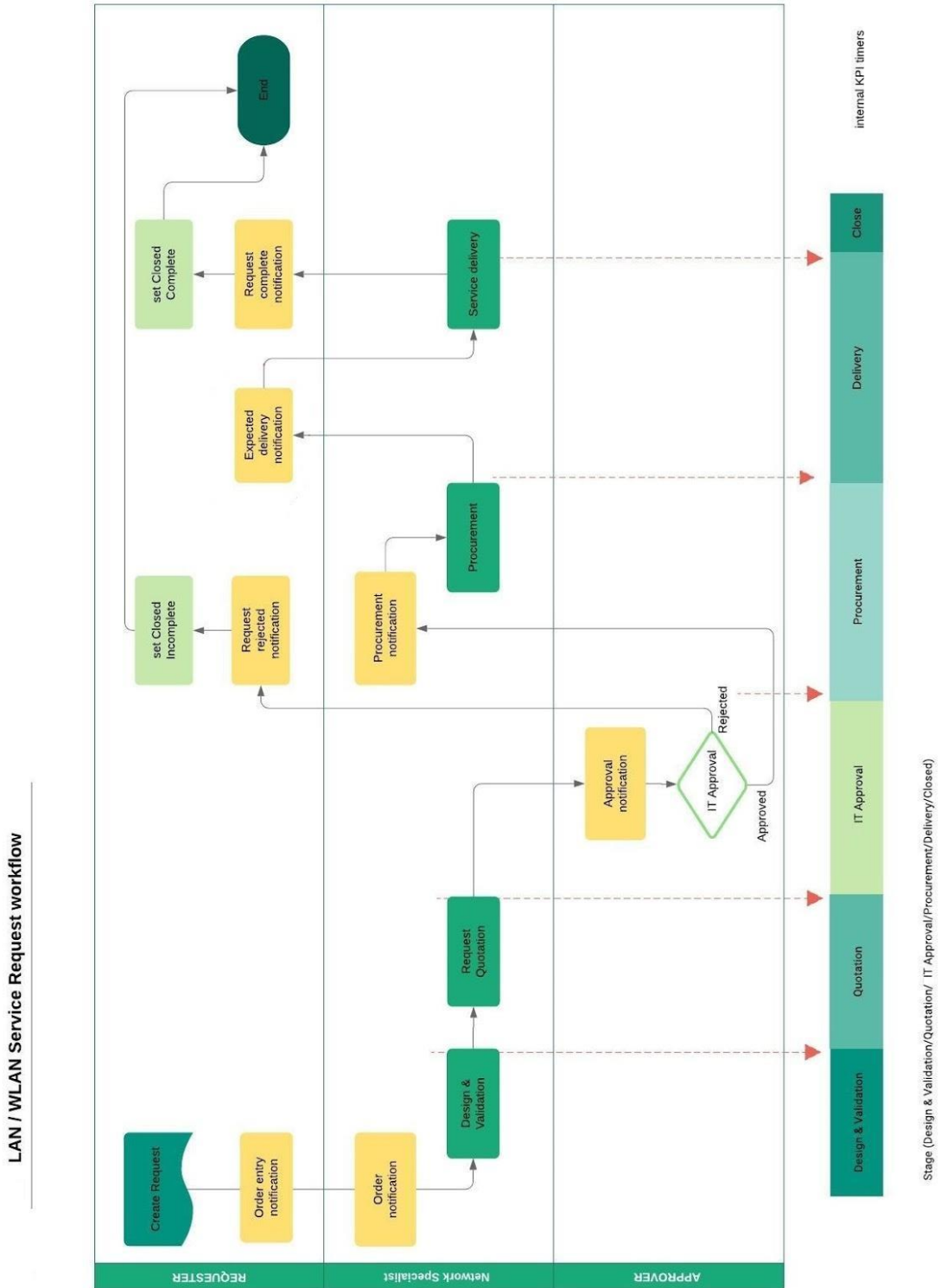


WORKFLOW PRO LAN / WLAN REQUEST (SPOLEČNOST A)

Příloha 4: Workflow pro LAN / WLAN Request (Společnost A)

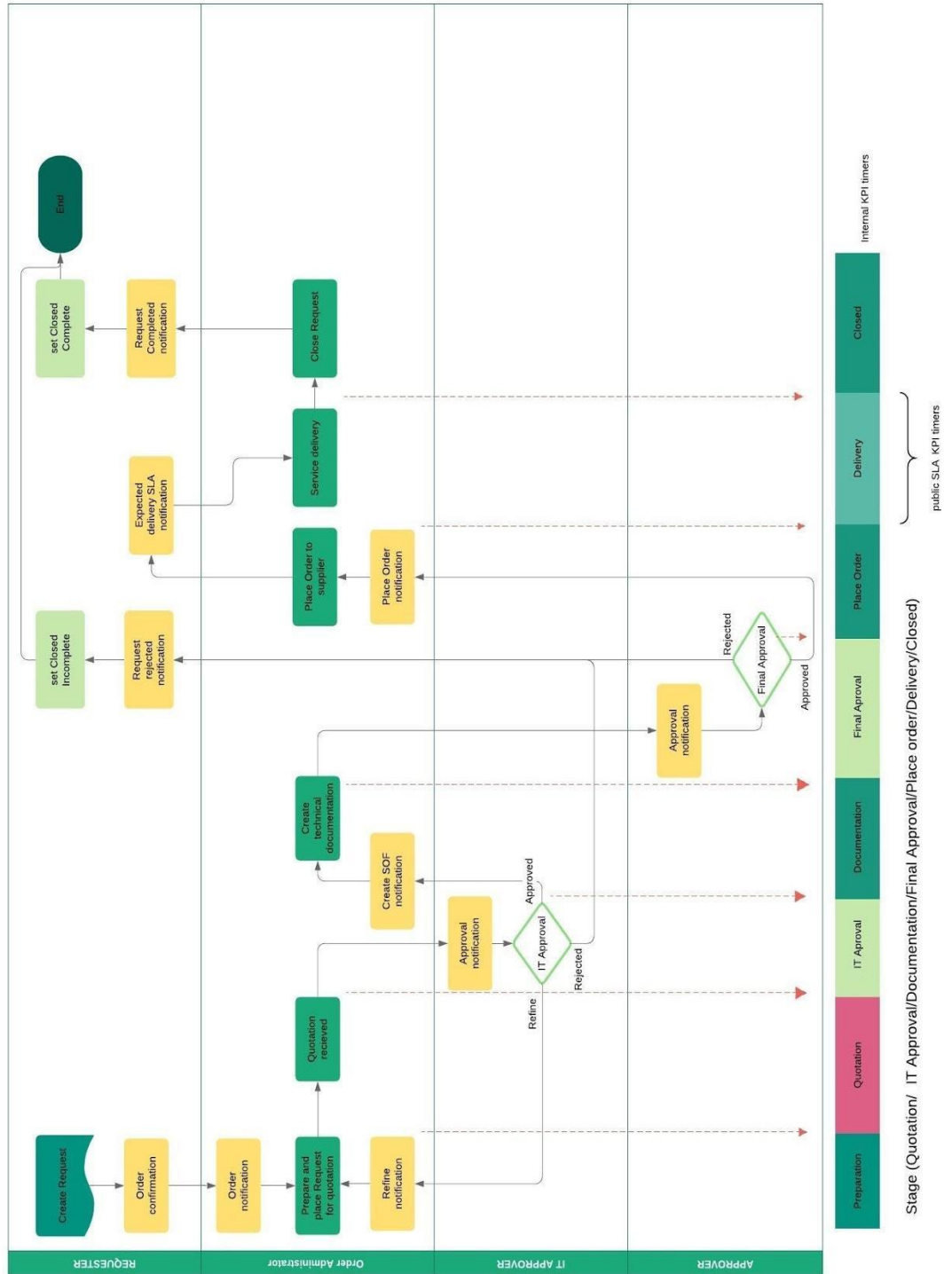


Příloha 5: Návrh nového workflow pro LAN / WLAN Service Request = Network Infrastructure Request



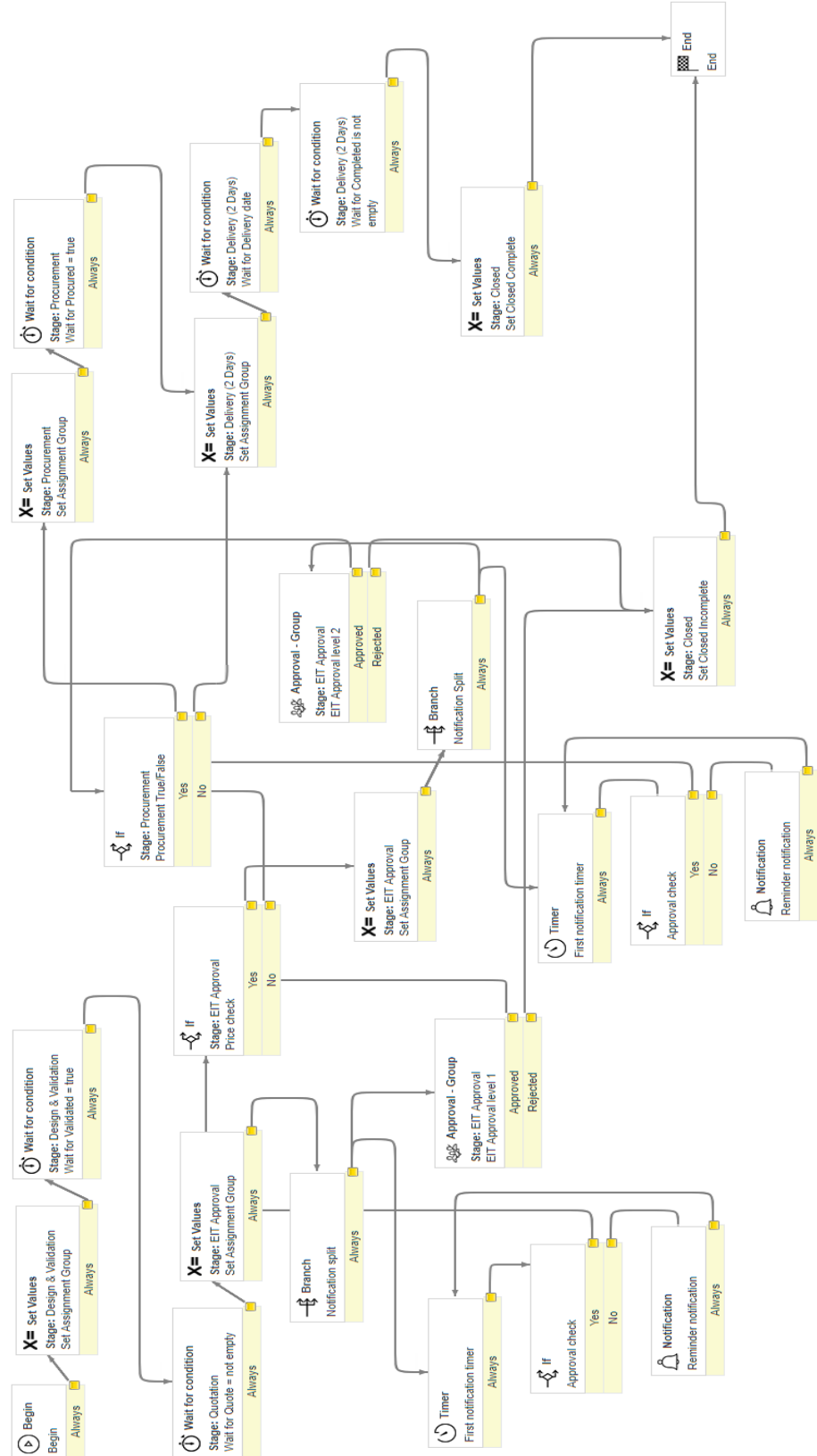
Příloha 6: Návrh nového workflow pro Managed Service Request

Managed Service Request workflow



Příloha 7: Implementované workflow pro Network Infrastructure Request (Společnost E)

WORKFLOW PRO NETWORK INFRASTRUCTURE REQUEST (SPOLEČNOST E)



Příloha 8: Implementované workflow pro Managed Service Request (Společnost E)

WORKFLOW PRO MANAGED SERVICE REQUEST (SPOLEČNOST E)

