



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA PODNIKATELSKÁ

FACULTY OF BUSINESS AND MANAGEMENT

ÚSTAV INFORMATIKY

INSTITUTE OF INFORMATICS

ANALÝZA A IMPLEMENTAČNÍ MODEL AGILNÍHO A PROJEKTOVÉHO ŘÍZENÍ V SOFTWAREM VÝVOJI

ANALYSIS AND IMPLEMENTATION MODEL OF AGILE AND PROJECT MANAGEMENT IN SOFTWARE
DEVELOPMENT

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Martin Sikora

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Lenka Smolíková, Ph.D.

BRNO 2016

ZADÁNÍ DIPLOMOVÉ PRÁCE

Sikora Martin, Bc.

Informační management (6209T015)

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách, Studijním a zkušebním řádem VUT v Brně a Směrnicí děkana pro realizaci bakalářských a magisterských studijních programů zadává diplomovou práci s názvem:

Analýza a implementační model agilního a projektového řízení v softwarovém vývoji

v anglickém jazyce:

Analysis and Implementation Model of Agile and Project Management in Software Development

Pokyny pro vypracování:

Úvod

Cíle práce, metody a postupy zpracování

Teoretická východiska práce

Analýza současného stavu

Návrh řešení a přínos návrhů řešení

Závěr

Seznam použité literatury

Seznam odborné literatury:

CHRISSIS, M. B. CMMI for development: guidelines for process integration and product improvement. 3. vyd. Upper Saddle River: Addison-Wesley, 2011. ISBN 978-0-321-71150-2.

LARSON, E. W. a C. F. GRAY. Project management: the managerial process. 5. vyd. New York: McGraw-Hill Irwin, 2011. ISBN 978-0-07-340334-2.

RUBIN, K. S. Essential Scrum: a practical guide to the most popular agile process. Upper Saddle River, NJ: Addison-Wesley, 2012. ISBN 978-0-13-704329-3.

SCHWALBE, K. Řízení projektů v IT: kompletní průvodce. 1. vyd. Brno: Computer Press, 2011. ISBN 978-80-251-2882-4.

ŠOCHOVÁ, Z. a E. KUNCE. Agilní metody řízení projektů. 1. vyd. Brno: Computer Press, 2014. ISBN 978-80-251-4194-6.

Vedoucí diplomové práce: Ing. Lenka Smolíková, Ph.D.

Termín odevzdání diplomové práce je stanoven časovým plánem akademického roku 2015/2016.

L.S.

doc. RNDr. Bedřich Půža, CSc.
Ředitel ústavu

doc. Ing. et Ing. Stanislav Škapa, Ph.D.
Děkan fakulty

V Brně, dne 29.2.2016

Abstrakt

Diplomová práce se zaměřuje na vytvoření implementačního modelu agilního a projektového řízení v softwarovém vývoji. Analyzuje tyto dva přístupy ve vývojovém týmu v mezinárodní společnosti a představuje návrh řešení jejich využití pro zdokonalení procesů ve vybraných procesních oblastech modelu CMMI.

Abstract

The diploma thesis deals with an establishment of the agile and project management implementation model in a software development. Thesis analyzes these approaches in a software development team of an international organization and presents a solution of their usages in a process improvement defined by the CMMI model in selected processes areas.

Klíčová slova

CMMI, SCRUM, agilní vývoj, projektový management, softwarový vývoj

Keywords

CMMI, SCRUM, agile development, project management, software development

Bibliografická citace práce

SIKORA, M. *Analýza a implementační model agilního a projektového řízení v softwarovém vývoji*. Brno: Vysoké učení technické v Brně, Fakulta podnikatelská, 2016. 108 s. Vedoucí diplomové práce Ing. Lenka Smolíková, Ph.D..

Čestné prohlášení

Prohlašuji, že předložená diplomová práce je původní a zpracoval jsem ji samostatně. Prohlašuji, že citace použitých pramenů je úplná, že jsem ve své práci neporušil autorská práva (ve smyslu Zákona č. 121/2000 Sb., o právu autorském a o právech souvisejících s právem autorským).

V Brně dne 25. května 2016

.....

podpis studenta

Poděkování

Tímto bych chtěl poděkovat paní Ing. Lence Smolíkové, Ph.D za vedení, připomínky a cenné rady při tvorbě diplomové práce.

Obsah

ÚVOD.....	11
1 CÍLE PRÁCE, METODY A POSTUPY ZPRACOVÁNÍ.....	13
2 TEORETICKÁ VÝCHODISKA.....	14
2.1 Projektový management.....	14
2.1.1 Projekt.....	14
2.1.2 Trojimperativ	15
2.1.3 Cíl projektu	16
2.1.4 Životní cyklus projektu.....	17
2.1.5 Stakeholders (zainteresované strany).....	18
2.1.6 Mezinárodní standardy projektového řízení	20
2.2 Agilní přístup k projektovému řízení	22
2.2.1 Agile manifesto.....	22
2.2.2 Extreme Programming.....	24
2.2.3 SCRUM	25
2.3 CMMI.....	32
2.3.1 Komponenty procesních oblastí.....	32
2.3.2 Procesní oblasti	34
2.3.3 Specifické a generické cíle	34
2.3.4 Specifické a generické praktiky.....	34
2.3.5 Úrovně způsobilosti a vyspělosti	34
3 ANALÝZA SOUČASNÉHO STAVU	36
3.1 Představení společnosti	36
3.1.1 Honeywell ve světě.....	36

3.1.2	Honeywell v České republice	37
3.1.3	Základní údaje.....	38
3.1.4	FSS.....	39
3.2	Strategická analýza společnosti.....	41
3.2.1	Analýza okolí SLEPT	41
3.2.2	Analýza 7S.....	43
3.2.3	Analýza SWOT	45
3.3	Metodiky a rámce.....	46
3.3.1	Kaizen	46
3.3.2	Six Sigma.....	47
3.3.3	IPDS.....	48
3.3.4	SCRUM	51
3.3.5	CMMI	62
3.4	Závěr analýzy	63
4	NÁVRH ŘEŠENÍ A PŘÍNOS NÁVRHU ŘEŠENÍ	64
4.1	CMMI procesní oblasti	64
4.1.1	Requirements Management (REQM)	64
4.1.2	Project Planning (PP).....	68
4.1.3	Project Monitoring and Control (PMC).....	78
4.1.4	Integrated Project Management (IPM)	86
4.1.5	Risk Management (RSKM)	91
4.2	Výsledný model	96
4.3	Způsob realizace řešení	99
4.4	Ekonomické zhodnocení	100

4.5	Přínos vlastního řešení	101
5	ZÁVĚR	103
6	SEZNAM POUŽITÉ LITERATURY	105

ÚVOD

Obecně se v globálních korporacích dostáváme do situací, kde se střetávají různé metodiky, rámce a standardy. Společnosti implementují tyto metodiky, rámce či standardy za účelem zvýšení kvality svých produktů nebo služeb. Tato práce se zaměřuje na prostředí mezinárodní korporace, která vychází z tradiční vodopádové metody projektového řízení, přičemž se rozhodla svým směřováním zaměřit i na metody agilní. Konkrétně se tato práce zaměřuje na tým, který již aplikoval agilní vývoj v podobě metodiky SCRUM. Organizace, ve které tento tým působí, se rozhodla aplikovat model CMMI, který slouží ke zkvalitnění procesů a jejich výstupů.

Tato práce si dává nelehký úkol skloubení těchto přístupů. Jejím cílem je tedy nalezení vzájemné synergie SCRUM metodologie, požadavků modelu CMMI a zároveň vlastního projektového rámce IPDS. Konkrétní výstupy jsou nejen analýza těchto metodologií (rámců), ale zejména praktický přesah v podobě doporučení a zároveň návrhu implementace hybridního modelu pro vybrané procesní oblasti modelu CMMI z kategorie projektového managementu, konkrétně oblasti Requirements Management, Project Planning, Project Monitoring and Control, Integrated Project Management a Risk Management.

Zkoumaný subjekt diplomové práce představuje vývojový tým Flight Support Services (dále jen FSS) společnosti Honeywell, který se zabývá vývojem mobilních aplikací. Jak už bylo zmíněno, tento tým pro svou činnost následuje skloubení agilní metody SCRUM a klasického projektového přístupu vycházejícího z celopodnikového procesu IPDS.

Diplomová práce je rozdělena na tři části. Teoretická část se věnuje projektovému managementu, agilnímu přístupu k projektovému managementu, převážně nejvíce používané metodice SCRUM a modelu CMMI. Analytická část obsahuje popis společnosti Honeywell, a strategickou analýzu její pobočky v Brně. Dále je popsán vývojový tým FSS a jeho činnost ve společnosti Honeywell. Výrazná část analýzy je věnována vývojovému procesu, metodě SCRUM a stavu modelu CMMI v týmu FSS. Závěrem analytické části je shrnutí hlavních problematických stránek. Praktická část má pevně stanovenou strukturu. Každá specifická praktika z vybraných procesních oblastí obsahuje návrh řešení a důležité výstupy. Dále je

uvedena tabulka představující výsledný hybridní model a ekonomické zhodnocení, založené na ohodnocení celkové náročnosti realizace navrženého řešení. Kapitola vlastního řešení je zakončena přínosem celé práce.

1 CÍLE PRÁCE, METODY A POSTUPY ZPRACOVÁNÍ

Cíle práce

Cílem této diplomové práce je zanalyzovat současný stav použití agilní metodologie SCRUM a projektového řízení v rámci modelu kvality CMMI a na jejím základě navrhnout implementační model hybridního fungování v softwarovém týmu. Zkoumané procesní oblasti modelu CMMI jsou Requirements Management, Project Planning, Project Monitoring and Control, Integrated Project Management a Risk Management.

Dílčím cílem je stanovení ekonomického zhodnocení založeného na odhadu náročnosti realizace navrhovaného řešení.

Metody a postupy zpracování

Práce je rozdělena na tři základní kapitoly. První část obsahuje teoretické poznatky nezbytné pro pochopení problematiky. Konkrétně je zde uvedena část týkající se projektového managementu, kde jsou uvedeny jeho základní pojmy. Dále agilní metoda SCRUM, popisující její základní filozofii, role, které zde vystupují, ceremonie a důležité artefakty. Poslední část je věnována modelu CMMI a popisu jeho komponent.

Druhou část se věnuje analýze zkoumaného subjektu, kterým je vývojový tým společnosti Honeywell. Při vypracování této části byla použita metoda sběru dat, kde tyto informace byly čerpány z projektové dokumentace a na základě diskuze s managementem týmu. Analýza se zabývá procesem vývoje produktů ve zkoumaném subjektu a na použití agilní metody SCRUM.

Návrh vlastního řešení byl vypracován podle vybraných procesních oblastí modelu CMMI s využitím agilní metody SCRUM a metod projektového managementu. Nezbytnou částí vlastního návrhu je ekonomické zhodnocení, které se zakládá na vytvořeném odhadu náročnosti realizace řešení.

2 TEORETICKÁ VÝCHODISKA

Začátek teoretické části diplomové práce je zaměřen na charakteristiku projektového managementu. Popisuje jeho základní pojmy, tedy co je projekt samotný, trojimperativ projektu, jeho cíl a životní cyklus. Závěr části projektového managementu je zaměřena na mezinárodní standardy projektového řízení, které jsou IPMA, PMBOOK a PRINCE2.

Druhá část teoretické části se zaměřuje na agilní přístup projektového řízení, kde je popsána filozofie tohoto přístupu. Tato část obsahuje popis dvou nejpobulárnějších metod agilního přístupu – Extreme Programming a SCRUM.

V poslední části je stručně popsán model CMMI a jeho komponenty, které slouží ke zvýšení úrovně podnikových procesů.

2.1 Projektový management

Jedním z velmi důležitých diskutovaných témat současné doby je téma projektového managementu. Všechny organizace, nehladě na jejich velikost, neustále implementují nové procesy, produkty či služby. Projektový management se zabývá právě tím, jak řídit aktivity nezbytné k úspěšnému splnění těchto úkolů (1).

Stavby jako velká pyramida v Gíze nebo velká Čínská zeď nemohly být dokončeny bez využití určitého typu podobnému přístupům projektového managementu. Základní principy projektového řízení se začaly formulovat začátkem dvacátého století, jejich vývoj následně akceleroval v období druhé světové války. Řízení projektů v té době se velkou mírou lišilo od současného moderního projektového řízení. Kdy v pozdějších letech k vývoji projektového managementu výrazně přispěl rozvoj IT (1).

2.1.1 Projekt

Pro pochopení projektového řízení je potřeba na samotném začátku definovat projekt samotný. Institut projektového managementu definuje projekt jako „časově omezené úsilí vynaložené na vytvoření unikátního produktu, služby nebo výstupu“ (2).

Pro odlišení projektu od různých dalších úsilí organizace mohou posloužit charakteristiky projektu (2):

- Stanovený cíl
- Dočasný
- Kombinace různých oddělení a zdrojů
- Unikátní
- Ohraničen časem, náklady a rozsahem

První charakteristikou je stanovený **cíl**, ať už se projekt týká výstavby nové budovy či vývojem softwaru, vždy je stanoven účel, proč je projekt uskutečněn. Projekt je **dočasný**, má stanoven začátek a jeho konec. Realizace projektu často překračuje hranice oddělení a využití jedné specializace, proto je důležitá **kombinace** různých oddělení, lidí a zdrojů. Projekt je **unikátní**, výstupem projektu je něco, co ještě nebylo vykonáno – pro představu například přistání sondy na planetě Mars. I obyčejná konstrukce budovy, která se skládá z rutinních postupů, vyžaduje určité modifikace, které ji dělají unikátní. Poslední charakteristiku nazýváme projektovým trojimperativem, ten je popsán v následující kapitole (2).

2.1.2 Trojimperativ

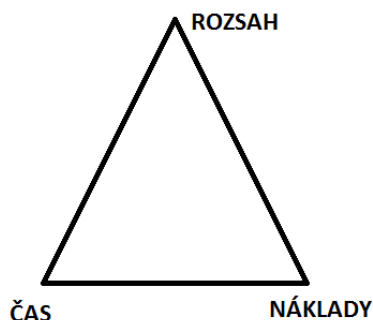
Pojem trojimperativ projektu představuje omezení v podobě času, nákladů a rozsahu. Důležitým faktem je provázanost těchto tří prvků (3).

K dosažení úspěchu projektu musí projektový manažer tyto tři veličiny respektovat a uvědomovat si jejich závislosti. Musí uvažovat o následujícím (3):

- **Rozsah** – co je výstupem projektu?
- **Čas** - jak dlouho projekt bude trvat? Kdy bude zákazníkovi produkt doručen?
- **Náklady** - kolik bude realizace projektu vyžadovat zdrojů? Jaký je rozpočet projektu?

Projektový manažer se musí rozhodnout, které aspekty trojimperativu jsou pro projekt nejdůležitější. Pokud je pro projekt prioritní dodržet časový plán, lze to dosáhnout změnou původního plánového rozsahu či změnou nákladů. Pokud naopak je prioritní rozsah projektu, bude muset projektový manažer přizpůsobit harmonogram či rozpočet. A pokud jsou

nejdůležitější náklady spojené s realizací projektu, lze ovlivnit zbylé dva aspekty. Změna jedné veličiny ovlivňuje zbylé dvě (3).



Obrázek č. 1: Trojimperativ projektu, Zdroj: Vlastní zpracování

2.1.3 Cíl projektu

Jedním z klíčových faktorů projektu je stanovený cíl. Jak tento cíl definujeme, výrazně ovlivňuje dosažení úspěchu projektu. Špatně definovaný cíl může mít za následek nesoulad výsledku projektu s tím, co požadovala některá ze zainteresovaných stran (4).

Definovat cíl správně není jednoduchá záležitost. Nelze pouze technicky popsat požadovaný výsledek. Je potřeba, aby si různé strany porozuměly v tom, co má být na konci realizace projektu výsledkem, k čemu má výsledek sloužit a za jakých podmínek by tento cíl měl být dosažen (4).

Každý stanovený cíl by měl splňovat požadavky techniky SMART. Tato technika je pomůckou pro správné definování cíle. Cíl by podle této techniky měl splňovat následující (4):

- **S** – specifický (specific) – potřebujeme vědět CO
- **M** – měřitelný (measurable) – potřebujeme určit, zda jsme cíle dosáhli
- **A** – akceptovaný (agreed) – zainteresované strany vědí, o co jde, shodli se
- **R** – realistický (realistic) – dosažení stanoveného cíle je reálné
- **T** – termínovaný (timed) – kdy má být cíle dosaženo

2.1.4 Životní cyklus projektu

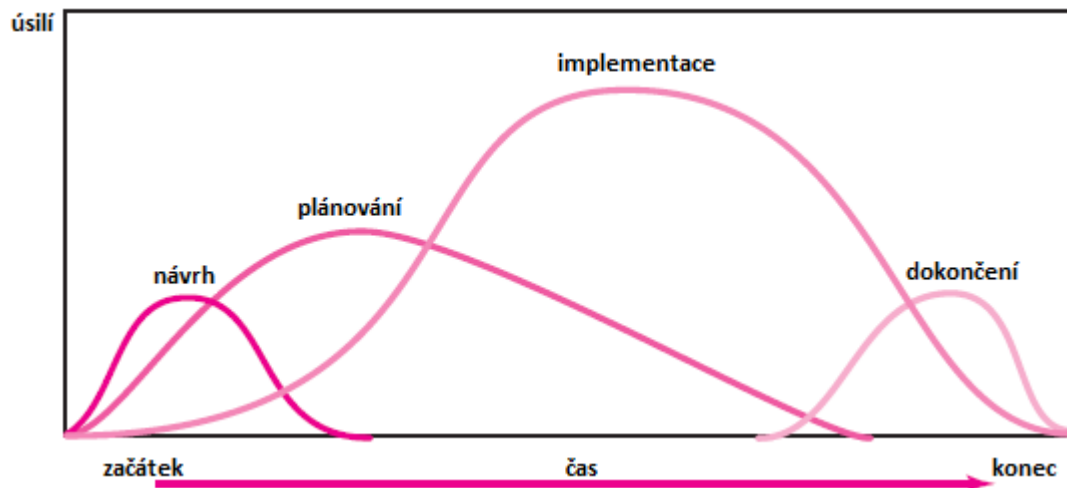
Projekty jsou provázány velkou mírou nejistoty, proto je potřeba jejich rozdělení na jednotlivé. Životní cyklus projektu je definován jednotlivými projektovými fázemi

Definice jednotlivých fází představuje práci, kterou je potřeba v jednotlivých fázích udělat. Jaké jsou její výstupy, kdy bude jejich dodávka a kdo se těchto fází účastní spolu s definicí, jak tyto aktivity budou kontrolovány a schvalovány (3).

Jednotlivé fáze životního cyklu závisí na typu projektu a oblasti, kde vznikají. V tradičním přístupu projektového managementu jsou definovány tyto obecné fáze (3):

- Návrh
- Plánování
- Implementace
- Dokončení

Ve fázi **návrhu** stanovujeme specifikaci projektu, jeho cíle a také je sestavován projektový tým spolu s určením hlavních zodpovědností. V **plánovací** fázi definujeme podrobnější plán realizace projektu, jeho časové určení, odhady nákladů a rozpočet. Fáze **implementace** zahrnuje konkrétní realizační činnosti. Postup práce, využití rozpočtu je měřeno a reportováno zainteresovaným stranám. Poslední fáze **dokončení** projektu je neméně důležitá jako tři předchozí. Tato fáze zahrnuje tři hlavní činnosti, předání výsledků a jejich schválení zainteresovanými stranami, uvolnění využívaných zdrojů a zpětné zhodnocení projektu (tzv. lessons learned) (2).



Obrázek č. 2: Životní cyklus projektu, Zdroj: Upraveno dle (2, str. 7)

Stupeň nejistoty je v počáteční fázi životního cyklu projektu největší, zároveň tato fáze vyžaduje nejnížší potřebu zdrojů. V tomto období jsou zainteresované strany schopné co nejvíce ovlivnit finální vlastnosti produktu, služeb či výsledků projektu. Cena za změnu těchto charakteristik stoupá s postupem projektu do dalších fází životního cyklu. S postupem projektu do dalších fází však také klesá míra nejistoty, kdy s postupem času jsou informace o cílech a požadavcích jasnější. O prvních dvou fázích se hovoří v souvislosti s **proveditelností projektu**. Fáze implementace a dokončení spojujeme s **akvizicí projektu** (3).

2.1.5 Stakeholders (zainteresované strany)

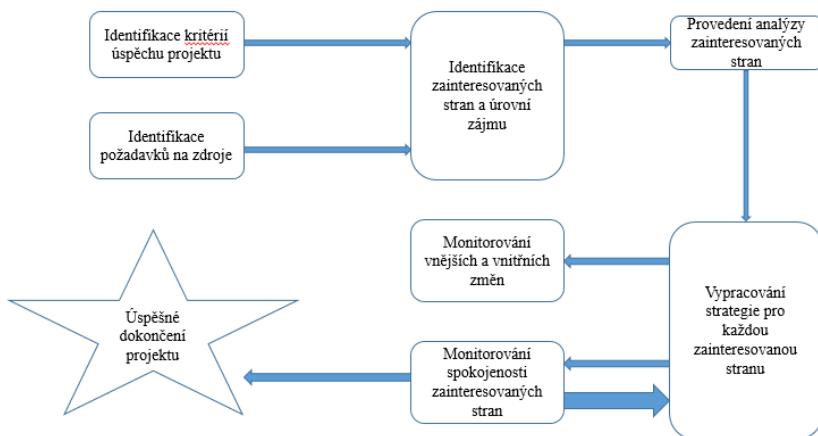
Zainteresovanou stranou v projektu je osoba/organizace, která se aktivně zapojuje do projektu, nebo jejíž zájmy mohou být pozitivně/negativně ovlivněny realizací projektu, příp. jeho výsledkem. Často také může ovlivnit průběh projektu či jeho výsledky (4).

Rozdělení zainteresovaných stran podle rolí se může lišit na typu projektu, například pokud se jedná o interní projekt (zavedení nové výrobní linky) nebo externí projekt (výstavba

budovy pro zákazníka). Členění podle zastávané role může vypadat například následovně (4):

- **Zadavatel (vlastník) projektu** – má zájem na realizaci projektu, na změně
- **Zákazník (uživatel) projektu** – hájí zájem osob, které budou pracovat s výstupy
- **Sponzor projektu** – autorita potřebná k rozhodování o základních aspektech projektu
- **Realizátor (dodavatel) projektu** – zastává zájmy zhotovitelů
- **Investor projektu** – představuje zájem vlastníka finančních nebo jiných zdrojů
- **Dotčené strany** – ostatní, které se projekt nějakým způsobem dotýká

Zainteresoovaných stran může být na projektu mnoho, každá může mít různý zájem, pohled či vliv na úspěšnosti projektu. Proto je potřeba všechny zainteresované strany správně identifikovat a klasifikovat na základě míry vlivu a zájmu na projektu a následně přiřadit prioritu, podle které budou tyto zájmy naplňovány. Snižuje se tak riziko neúspěšnosti projektu. Tuhle aktivitu představuje **řízení zainteresovaných stran**. Řízení zainteresovaných stran se stará o správnou identifikaci klíčových zájmových skupin, následnou analýzu těchto zainteresovaných stran a vytvoření přístupu, strategie k jednotlivým zainteresovaným stranám (4).



Obrázek č. 3: Proces řízení zainteresovaných stran, Zdroj: Upraveno dle (4, str. 51)

Otázky kladené v jednotlivých krocích řízení zainteresovaných stran mohou být (4):

- **Identifikace** - Kdo chce, aby projekt uspěl/neuspěl? Kdo bude ze změny těžit a koho to naopak zničí? S kým a bez koho tato změna nebude možná?
- **Analýza** - Jaké mají jednotlivé strany od projektu očekávání? Jaký mají zájem a vliv? Které zainteresované strany jsou primární, a které naopak sekundární?
- **Strategie** - Jak budeme jednotlivé zainteresované strany informovat? Které strany zapojíme do projektu? Jak budeme komunikovat?

2.1.6 Mezinárodní standardy projektového řízení

Stejně jako u jiných profesí, tak i u projektového managementu vznikly na základě neustálého růstu počtu a složitostí projektů určité standardy profesionálního chování. Standardy projektového managementu, však na rozdíl od standardů jiných profesí, neobsahují různé vyhlášky či normy, ale spíše doporučení, určitou filozofii, nabídku osvědčených metod, atd. Tato skutečnost vznikla na základě velkého množství různých proměnných, které se při řízení projektu vyskytují, a které je obtížné měřit. Tyto standardy byly vytvořeny na základě nejlepších zkušeností mnoha významných manažerů (4).

I proto je potřeba brát tyto standardy jako inspiraci než tvrdý zákon. Standardů projektového řízení existuje více, téměř všechny však vycházejí z podobné základní filozofie, používají obdobné metoda a názvosloví. Většinou jde jen o jiný úhel pohledu na tutéž oblast. V současné době k hlavním standardům patří PMI, IPMA, PRINCE2®. Rozdíly v těchto standardech představují podklady, na kterých byly vytvořeny a také způsob zpracování. Všechny výše uvedené standardy poskytují a propagují možnost certifikace projektových manažerů (4).

PMBoK®

Project Management Body of Knowledge, tento standard vytváří a udržuje Project Management Institute, PMI. Standard vznikl na základě standardů US army v sedmdesátých letech 20. století. Tyto standardy byly jednoduše aplikovatelné a rozšiřitelné i do sektoru komerčních projektů. PMBoK přistupuje k problematice projektového řízení procesním

pojetím. Definuje pět hlavních rodin procesů, devět oblastí znalostí, jednotlivé procesy a jejich vzájemné vazby. Jsou zde definovány vstupy a výstupy spolu s nástroji transformace u každého procesu a procesních kroků. Tento standard je typický pro společnosti zabývající se především IT a jsou vlastněny americkým kapitálem (4).

PRINCE2®

APM Group Ltd. spravuje britský standard PROjects IN Controlled Environments, metodiku vlastní OGC. Jedná se o další procesní pojetí projektového managementu, které vzniklo na základě zadání britského ministerstva průmyslu a obchodu. V důsledku velkého množství IT projektů, u kterých bylo potřeba zajistit dodržování kvality, harmonogramu, rozpočtů a dosahování cílů byla vytvořena metodika, ze které se stal standard. Dodržení těchto metod bylo povinné pro získání státní zakázky. Přestože byl tento standard vytvořen převážně pro IT prostředí, je v současné době použitelný obecně (4).

IPMA

Standard vznikl v 60. letech v Evropě na základě státních norem několika evropských států.

Na rozdíl od dvou předchozích standardů projektového managementu, které k projektovému managementu přistupují procesně, je standard vytvářený a spravovaný profesní organizací International Project Management Association kompetenční. Standard se tedy zaměřuje místo na přesnou podobu definovaných procesů a jejich konkrétní aplikaci, tak na schopnosti a dovednosti, tedy kompetence, projektových manažérů a členů projektových týmů. Standard nediktuje procesy, ale vnáší do projektového řízení určitou dávku kreativity a důraz na vlastní názor spolu s využitím doporučených procesních kroků. Základní filozofie je obdobná jako u předchozích dvou standardů. Problematika standardu je rozdělena do tří základních kompetenčních oblastí – technické kompetence, behaviorální kompetence a kontextové kompetence (4).

2.2 Agilní přístup k projektovému řízení

Vedle klasických přístupů k řízení projektů se v poslední době velmi často mluví o přístupu k řízení projektů agilními metodami, ačkoliv tyto metody nejsou žádnou novinkou. Tyto metody představují souhrn základních principů, kterými by se měl úspěšný projekt řídit, a aplikují je na komplexní prostředí, jakým je vývoj softwarových produktů. Agilní přístup k projektovému řízení však lze využívat nejen v tomto oboru. Agilní metody představují iterativní proces, který klade důraz na týmové spolupráci, otevřené komunikaci, zapojení zákazníka a připravenosti na změnu (5).

V této kapitole budou představeny konkrétní metody agilního přístupu, které jsou aktuálně nejvíce využívány v praxi. Hlavní důraz bude kladen na metodu SCRUM.

2.2.1 Agile manifesto

V roce 2001 vznikl tzv. „Manifest agilního vývoje software“, který byl vytvořen a podepsán několika reprezentanty agilních přístupů. Tento manifest představuje jakousi filosofii agilního přístupu. Zaměřuje se na čtyři oblasti, kdy hodnoty na levé straně mají vyšší váhu než hodnoty na pravé straně (6).

Obsah manifesta je následující (6):

- **Jednotlivci a interakce** před procesy a nástroji
- **Fungující software** před vyčerpávající dokumentací
- **Spolupráce se zákazníkem** před vyjednáváním o smlouvě
- **Reagování na změny** před dodržováním plánu

Původní verze v anglickém podání je k nalezení na <http://www.agilemanifesto.org/>

Jednotlivci a interakce před procesy a nástroji

Všeobecně známým faktem je, že skupiny individuálně pracujících jednotlivců dosahují menších pracovních výsledků, než týmy s vysokou mírou interakce a spolupráce. Tým, který pracuje v silně kolaborativním prostředí, kde si vytváří své nástroje, které mu dopomáhají k vytvoření kvalitního softwaru v krátkém čase, má větší šanci na úspěch než tým, který

pracuje v přesně definovaném procesním prostředí s nejdražšími nástroji, ale bez zaměření na interakci a spolupráci (7).

Manifest však neříká, že by se procesy neměly vyskytovat, a že týmy by měly pracovat zcela bez nástrojů. Rozdíl je v tom, že týmy by si měly tyto nástroje sami vybírat a používat jen ty, které jim opravdu pomáhají v dosažení kvalitního výsledku (7).

Fungující software před vyčerpávající dokumentací

Ačkoli je dokumentace vysoce důležitá, neměla by převážet nad vlastním produktem. Dokumentace v agilním prostředí by měla sloužit pouze jako reference pro oblasti, které nejsou intuitivní a snadno pochopitelné. To samé platí i pro interní dokumentaci, obsahující například znalosti architektury. Tato dokumentace by měla být stručná a měla by postihnout klíčové informace (7).

Poměr mezi námahou a časem, který investujeme do psaní dokumentace, by měl odpovídat hodnotě, kterou její zákazníci z této dokumentace načerpají (7).

Spolupráce se zákazníkem před vyjednáváním o smlouvě

Dohady a smluvní jednání jsou ve většině případů velmi komplikovaná, za účasti právníků se obě strany snaží vytvořit precizní smlouvu. Agilní přístup však preferuje rozumnou spolupráci se zákazníkem před snahou dosáhnout dokonalých smluv (8).

Cílem komunikace se zákazníkem je vytvoření spolupracujícího týmu, který zajistí úspěch oběma stranám. Smlouvy jsou důležité, neměly by však nahradit spolupráci a komunikaci (7).

Reagování na změny před dodržováním plánu

Ačkoliv jsou projektové plány velmi důležité, v agilním prostředí by však měly sloužit pouze jako vodítko a neměly by řídit životy spolupracujících firem. Agilní přístup předpokládá, že zákazníkovo očekávání a požadavky na produkt se během vývoje mohou měnit. Spoluprací a adaptací na tyto změny, může být zákazníkovi dodán produkt, který opravdu vyžaduje (7).

2.2.2 Extreme Programming

První z představených metod agilního přístupu je Extreme Programming. Tato metoda je velmi oblíbená, je však v poslední době stále častěji nahrazována metodou SCRUM. Extreme Programming zastává hodnoty agilního přístupu a uvádí je do praxe. Vývojáři, kteří neustále komunikují se zákazníkem, na základě zpětné vazby upravují produkt dle požadavků i v pozdějších fázích vývoje. Základem vývoje je jednoduchý a čistý design a testování již od samého začátku. Funkční produkt je dodáván po malých částech, v co nejdříve možném čase (7).

Hlavní hodnoty Extreme Programming jsou (7):

- Jednoduchost
- Komunikace
- Zpětná vazba
- Respekt
- Odvaha

Jednoduchost je myšlena soustředěnost týmu pouze na to, co je v daný moment nejdůležitější. Tým se nezabývá prací navíc, nepředvídá, co by zákazník v budoucnu mohl vyžadovat. Pomocí časté **komunikace** v rámci týmu, tým společně pracuje na návrhu řešení, implementaci a samotném testování produktu. Společně se snaží najít řešení problémů, které je blokují. Tým produkuje malé části softwaru v krátkých iteracích. Tyto části jsou prezentovány zákazníkovi a ten poskytuje **zpětnou vazbu**. Na základě této zpětné vazby jsou případné změny implementovány dle požadavků co nejrychleji. Vzájemný **respekt** je jedním z nejdůležitějších stavebních kamenů dobře fungujícího týmu. Tým také respektuje přání a potřeby zákazníka a zákazník naopak respektuje technické znalosti a background týmu. **Odvaha** představuje odvahu každého člena týmu přijmout zodpovědnost, zviditelnit problémy, nazývat věci pravými jmény, přijmout změnu a dělat věci jinak než doposud (7).

Projekt začíná sběrem požadavků od uživatele a definováním User Stories. Na začátku projektu se doporučuje testování tzv. „spike solutions“, tedy testování architektury a designu, v oblastech, u kterých je větší předpoklad výskytu rizik. Následující aktivitou je Release Planning, kterého se účastní všechny strany tedy zákazník, manažeři a vývojový tým.

Hlavním cílem této aktivity je dohoda na release plánu. Následuje iterace, která trvá většinou 1-3 týdny. Během této iterace jsou vybrány User Stories, které představují pro zákazníka přínos maximální obchodní hodnoty. Tyto User Stories jsou dekomponovány na drobné technické úkoly a ohodnoceny pracností. Součástí implementace jsou také testy (7).

2.2.3 SCRUM

Jedna z nejvíce používaných metod agilního přístupu. Nepředstavuje proces nebo přímo techniku, ale spíše procesní rámec používaný k řízení komplexních projektů pro vývoj produktů. SCRUM jako takový obsahuje Scrum tým a jeho role, související ceremonie, artefakty a pravidla. Všechny tyto části jsou vzájemně provázány a hrajou důležitou roli k dosažení úspěchu projektu (9).

SCRUM klade důraz na znalosti založené na zkušenostech a na schopnosti rozhodovat se na základě známých souvislostí. K posílení předvídatelnosti a lepší kontrole rizik se používá iterační a inkrementální přístup k vývoji produktu. Stojí na třech základních pilířích: transparentnost, kontrola, adaptace (9).

SCRUM role

Product Owner

Product Owner představuje týmu cíl a vizi projektu. Je zodpovědný za to, že projekt maximalizuje hodnotu investice. Product Owner je také zodpovědný za řízení Product Backlogu, což zahrnuje tyto úkoly (9):

- Jasný popis položek Product Backlogu
- Seřazení položek k nejlepšímu dosažení cíle
- Product Backlog je viditelný všem, maximalizuje vypovídající hodnotu
- Tým chápe zadávanou práci

Product Owner má nejvyšší autoritu co se projektu týče (9).

Scrum Master

Scrum Master pomáhá týmu v efektivním využívání a pochopení metodiky SCRUM. Chrání tým před okolními vjemy. Scrum Master není nadřazená pozice členům týmu, funguje spíše jako coach ke zvýšení efektivity členů týmů. Schopný Scrum Master by měl mít tyto schopnosti - odpovědný, skromný, kooperativní, angažovaný, vlivný, zběhlý (10).

Hlavní povinnosti Scrum Mastera jsou následující (9):

- Facilitátor
- Pomáhá týmu v pochopení a dodržení metodiky SCRUM
- Dopomáhá týmu k „self-organized“ a „cross functional“
- Dopomáhá týmu k dodání produktu s nejvyšší kvalitou
- Odstraňuje překážky, které zabraňují týmu pokračovat v efektivní práci
- Pomáhá organizaci v přijetí metodiky SCRUM

Vývojový tým

Skládá se z profesionálů ve svém oboru, kteří jsou zodpovědní za dodání inkrementu potencionálně hotového výrobku na konci každého sprintu. Vývojový tým je uzpůsoben a organizací podporován k organizaci a řízení své vlastní práce, která přispívá k výslednému produktu. Tým se snaží pracovat co nejefektivněji. Pro vývojový tým platí následující charakteristiky (9):

- „Self-organized“ – pouze tým rozhoduje jak transformovat položku Product Backlogu na reálný přírůstek funkcionality
- „Cross functional“ – v týmu se nacházejí všechny dovednosti potřebné pro vytvoření inkrementu produktu
- Zodpovědnost – tým je zodpovědný za dodání přírůstku, ke kterému se celý tým zavázal

SCRUM artefakty

Sprint

Jedním z nejdůležitějších pojmů metodiky SCRUM je sprint, který představuje časově uzavřený úsek, ne delší než měsíc, za který je dodán použitelný přírůstek produktu v odpovídající kvalitě. Sprint začíná hned po ukončení sprintu předešlého. Během sprintu nesmí docházet ke změnám, které by ohrozily cíl aktuálního sprintu. Sprint můžeme chápat jako projekt, který nemá délku trvání delší jednoho měsíce. Každý sprint je stanoven cíl, co má být výstupem sprintu a také plán jak tohoto cíle dosáhnout. Sprint je ukončen Product Ownerem, kterému je představen výsledek sprintu (9).

Epic

Představuje velký funkční blok, udržuje globální kontext a drží jednotlivé User Stories pohromadě. Při počátečním definování Product Backlogu se začíná právě s Epicem (7).

User Story

User Story je možno chápat jako uživatelský popis toho, co by systém měl dělat z uživatelského pohledu. Je popsána pomocí jednoduchých vět, formát User Story často vypadá následovně „*Jako **ROLE** chci **CÍL** tak, aby **UŽITEK**.*“ Role představuje definici, koho se daná User Story týká. Uživatel může vystupovat v různých rolích (editor, fakturant, účetní, atd.). Nejnutnější částí popisu User Story je cíl, tedy definice toho, co má systém vykonávat za činnost. Poslední část se týká definice užitku, co splněním cíle dosáhneme (8).

Každá User Story by měla splňovat kritéria INVEST, tedy **I**ndependent – jednotlivé User Stories musí být na sobě nezávislé, **N**egotiable - vysvětlitelná, **V**aluable – má hodnotu pro uživatele, **E**stimable - ohodnotitelná, **S**mall – malá, lze ji dokončit nejdéle za polovinu jednoho sprintu, **T**estable - testovatelná. Pokud daná User Story kritéria nesplňuje, nemusí být jasné, co je vlastně jejím obsahem a jaký má přínos. User Story je zaměřena na obchodní hodnotu. Každá User Story by také měla mít uvedena akceptační kritéria, seznam podmínek, které musí být splněny, aby mohl Product Owner User Story akceptovat (7).

Product Backlog

Product Backlog představuje seznam všeho relevantního, co je potřeba k vývoji produktu. Product Owner je zodpovědný za udržování Product Backlogu a také za prioritizaci jeho položek. Product Backlog je dynamický a v průběhu vývoje se neustále mění. Tento seznam obsahuje potřebné vlastnosti produktu, funkcionalitu, požadavky, vylepšení a také opravy (9).

Product Backlog položky jsou většinou ve formě User Story, obsahují popis, ale i odhad komplexity/náročnosti v relativních jednotkách. Tuto náročnost odhaduje tým. Jak už bylo zmíněno, Product Backlog je seřazen dle priority. Kde na horních položkách jsou User Stories, které mají největší prioritu, jsou kvalitně popsány a připraveny pro akceptaci vývojového týmu pro přiřazení do sprintu. Product Backlog také obsahuje položky, které ještě nejsou detailně popsány, ale poskytují hrubou představu o tom, co bude potřeba dodat (7).

Sprint Backlog

Sprint Backlog představuje výstup z ceremonie Sprint Planning, je to seznam ohodnocených, seřazených User Stories, získaných z Product Backlogu. Představuje veškerou práci, ke které se tým zavázal, a která bude realizována v nadcházejícím sprintu (9).

SCRUM ceremonie

Backlog Grooming

Pokud se tým nachází v prostředí se složitou architekturou a mnoha závislostmi je vhodné pořádat Backlog Grooming. Tento meeting je nejčastěji pořádán během sprintu a věnuje se k obecnému porozumění prioritních User Stories. Product Owner spolu s týmem prochází prioritní User Stories a spolu si ujasňují, co vlastně představují. Na tomto meetingu také dochází k ohodnocení User Stories jejich náročností, pro tuto činnost se často využívá estimační metoda planning poker. Tento meeting slouží jako prekvizita pro Sprint Planning (8).

Sprint Planning

Sprint Planning je jednou z nejdůležitějších částí sprintu. Tato ceremonie slouží k naplánování následujícího sprintu, tedy určení úkolů, které budou v dalším sprintu řešeny a stanovení cíle sprintu (8).

Tento meeting se skládá z následujících kroků (8):

- Definice cíle sprintu
- Vytvoření Sprint Backlogu

V prvním kroku Product Owner navrhne cíle sprintu, tedy definuje, co od daného sprintu očekává a co má být výstupem na jeho konci. Následně Product Owner představuje jednotlivé User Stories, které budou řešeny ve sprintu. Tým vybírá User Stories, které je schopen ve sprintu dokončit – tím vytváří Sprint Backlog. Scrum Master má za úkol celý meeting moderovat a také dopomáhá týmu získat přesné specifikace jednotlivých User Stories a jejich akceptační kritéria. Scrum Master dále pomáhá týmu ujasnit, zda vybrané User Stories je schopen doručit a zda některý člen z týmu nemá pocit, že má nedostatek práce či naopak příliš mnoho (7).

Součástí tohoto meetingu je také rozbití jednotlivých User Stories na jednotlivé úkoly, které jsou nezbytné pro jejich splnění. Ke konci meetingu se tým zaváže Product Ownerovi za splnění sestaveného Sprint Backlogu. Tento Sprint Backlog je v průběhu sprintu neměnný, Product Owner musí tedy vždy dobře zvážit priority (7).

Daily Scrum

Nejkratší ceremonie sprintu slouží k synchronizaci členů týmu ohledně aktivit, které budou následovat v dalších 24 hodinách. Každý člen týmu odpoví na následující tři otázky (9):

- Co jsem dělal včera pro dosažení cíle sprintu?
- Co budu dělat dnes?
- Jsou zde nějaké překážky, které mi brání?

Vzhledem k tomu, že většina pracovníků je do značné míry nezávislá, je smyslem tohoto meetingu hlavně zajištění informovanosti všech členů týmu. Dále je zde vyčleněn prostor

k identifikaci vzniklých problémů, jejich následnému řešení a k souhrnnému zhodnocení stavu sprintu. Je vhodné, aby tento meeting byl pořádán ve stejném čase a na stejném místě. Účast celého týmu je vyžadována, Scrum Master zde hraje roli moderátora, Product Owner roli tichého pozorovatele. Doporučená délka tohoto meetingu je maximálně 15min (8).

Sprint Review (Sprint Demo)

Meeting, který se odehrává na samotném konci sprintu. Hlavním účelem je prezentace týmem dokončených User Stories zákazníkovi. Jako zákazník se v tomto kontextu uvažuje kdokoliv, kdo má na projektu nějaký zájem. Dokončená User Story znamená, že jsou splněna všechna stanovená akceptační kritéria a „definition of done“ – tedy. Tým na základě prezentace dokončených User Stories získává okamžitou zpětnou vazbu. Krátká část tohoto meetingu je věnována také zhodnocení právě dokončeného sprintu, kde tým porovnává původní závazek vůči skutečnému výsledku (7).

Sprint Retrospective

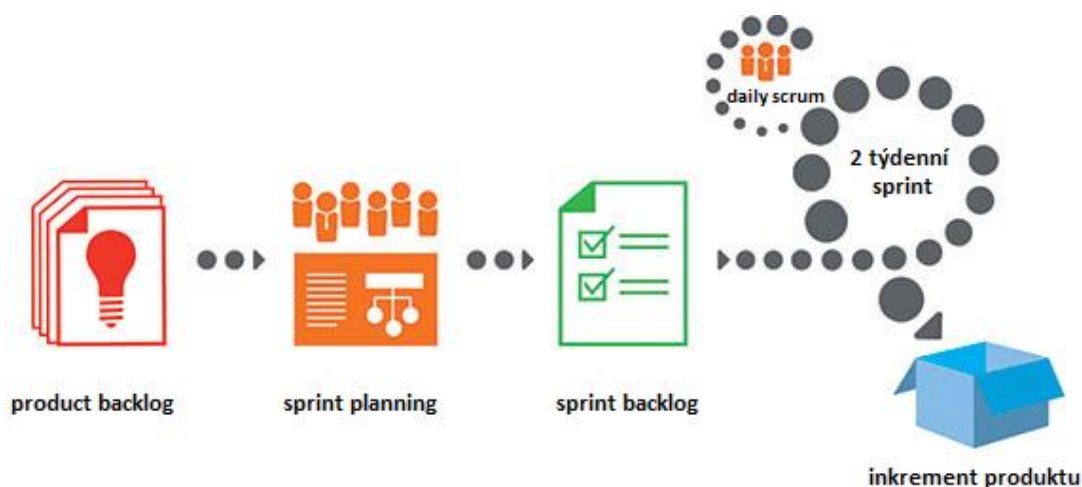
Tento čistě neformální meeting slouží k identifikaci problémů, které vznikly v právě dokončeném sprintu, co je potřeba zlepšit a jaké kroky je k tomuto zlepšení nutno provést. Je zde prostor pro hodnocení všeho, co se ve sprintu událo, technika plnění úkolů, přístup jednotlivých členů a podobně. Výstupem Sprint Retrospective by měl být seznam toho, co lze ponechat stejné (tedy toho, co funguje a není to nutné měnit), co je potřeba změnit (věci, které fungují principiálně správně, ale vyžadují určité korekce), co zrušit (skutečnosti, které se ukazují být negativními) a co naopak zavést (nové principy, technologie, procesy či postupy) (8).

Často se při retrospektivě používá vizualizace na tabuli ve tvaru hvězdice, kde každý člen přidává lístečky do již zmíněných kategorií (co ponechat stejné, co změnit, atd.) a následně se o těchto bodech vedou diskuze a celý tým ohodnocuje důležitost těchto bodů. Forma tohoto meetingu je různá, je v podstatě na moderátorovi, kterou praktiku využije (rybí kost, loď, metoda ESPV) (7).

Scrum shrnutí

Scrum tým se skládá z 5-7 lidí, v rámci daných pravidel se řídí sám, je tzv. „self-organized“. Tým je „cross functional“ a členové si vzájemně pomáhají a organizují se tak, jak je to v danou chvíli nejlepší. Ke každému týmu patří jeden Scrum Master a Product Owner. Celý tým je plně alokovaný a v ideálním případě sedí spolu ve stejné místnosti (7).

Před začátkem každého sprintu Product Owner na Sprint Planning meetingu představí jednotlivé User Stories, které má seřazený v Product Backlogu dle priority. Tyto User Stories jsou dále týmem rozbity na jednotlivé úkoly. Vybrané User Stories, které se tým zaváže doručit na konci sprintu, představují Sprint Backlog. Během sprintu tým reportuje svůj aktuální stav na Daily Scrum meetingu, jsou zde také identifikovány problémy, které brání týmu dokončit jednotlivé User Stories. Dokončené User Stories se prezentují zainteresovaným stranám na Sprint Review meetingu, kde je také zrevidován finální stav sprintu. Sprint Retrospective meeting slouží týmu ke zlepšování procesů či postupů a identifikaci best practices (7).



Obrázek č. 4: SCRUM, Zdroj: Upraveno dle (9)

2.3 CMMI

V současné době mají podnikatelské společnosti na výběr ze široké škály různých standardů, metodologií, modelů zralosti, které slouží ke zlepšování výrobních procesů, zkvalitnění nabízených služeb a dále. Většina těchto přístupů se však zaměřuje na specifickou část podnikání a k problémům, které tíží většinu společností přistupují nesystematicky. Tento přístup zaměřený pouze na specifickou část, nepomůže odstranit překážky a bariéry, které se mohou ve společnosti vyskytovat (11).

CMMI-DEV (Capability Maturity Model Integration for Development) nabízí možnost vyhnout se či přímo eliminovat vzniklé překážky a bariéry. Obsahuje nejlepší praktiky, které se týkají aktivit spojených s vývojem produktu či služby. Tyto praktiky pokrývají celý životní cyklus výrobku či služby od konceptu až po dodání a následnou údržbu. Důraz je kladen na práci, nutnou pro vytvoření a udržování produktu (11).

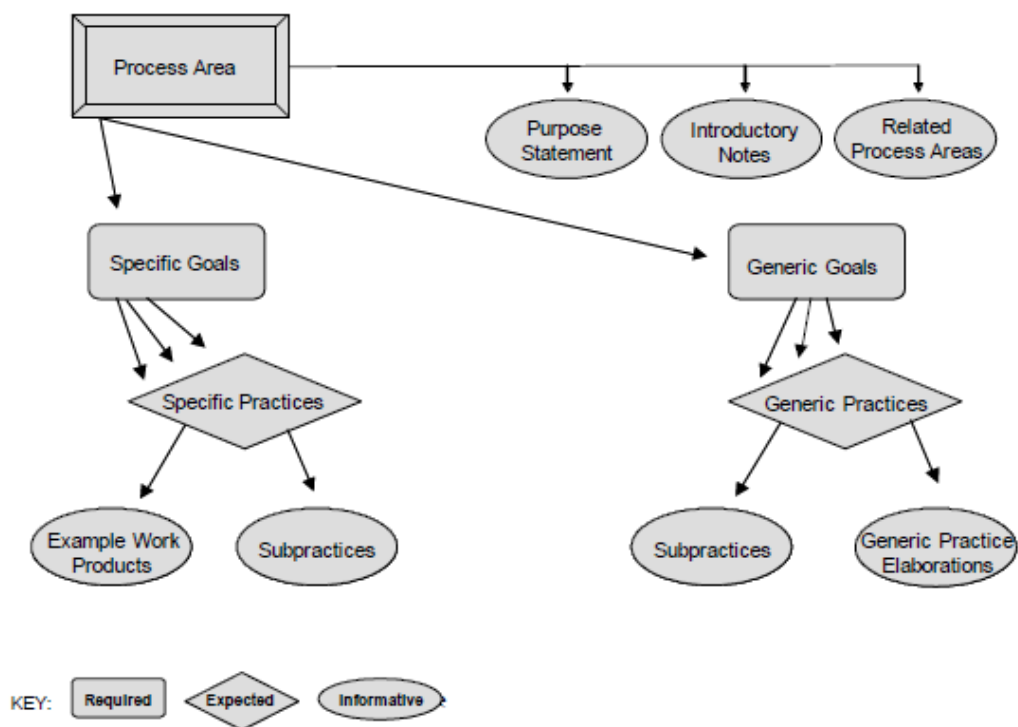
CMMI-DEV představuje referenční model, který pokrývá aktivity, které se vyskytují jak ve vývoji produktů, tak ve vývoji služeb. Tento referenční model je využíván společnostmi působících v oblastech letectví, bankovníctví, počítačového hardwaru, vývoje softwaru, automobilovém průmyslu, a dalších. CMMI-DEV pokrývá nejlepší praktiky v oblastech projektového managementu, procesního managementu, systémového inženýrství, hardwarového inženýrství a ostatních oblastech nutných k vývoji a údržbě (11).

CMMI-DEV obsahuje základní a také nezbytné elementy efektivních procesů. Základy těchto elementů jsou postaveny na konceptech vytvořených Crosbym, Demingem, Juranem a Humphreyem (11).

2.3.1 Komponenty procesních oblastí

Tato kapitola popisuje komponenty, které se nacházejí v každé procesní oblasti. Pochopení těchto komponentů je základem pro použití modelu CMMI-DEV. Komponenty jsou rozděleny do tří kategorií – **vyžadované (required)**, **očekávané (expected)** a **informativní (informative)**. **Vyžadované** komponenty jsou nepostradatelné pro dosažení vylepšení procesu v dané procesní oblasti. Tuto kategorii komponentů představují specifické a

generické cíle. Komponenty očekávané popisují aktivity, které jsou důležité k dosažení vyžadovaných komponentů. Očekávané komponenty představují specifické a generické praktiky. K uspokojení specifických a generických cílů musí být popsány praktiky, nebo jejich případné alternativy obsaženy v naplánovaných a implementovaných procesech organizace. **Informativní** komponenty slouží pro porozumění vyžadovaných a očekávaných komponentů. Tyto komponenty mohou být například ve formě příkladů, detailních vysvětlení, zdrojů a dále. Tyto komponenty jsou velmi důležité k pochopení jednotlivých cílů, praktik a nakonec celého modelu CMMI-DEV (11).



Obrázek č. 5: Komponenty procesních oblastí, Zdroj: Upraveno dle (11, str. 10)

2.3.2 Procesní oblasti

Procesní oblast (Process Area) představuje shluk souvisejících praktik, které, pokud jsou implementovány současně, vedou k uspokojení sady cílů, důležitých pro dosažení zlepšení v dané oblasti. CMMI-DEV obsahuje 22 procesních oblastí rozdělených do čtyř kategorií – Řízení procesů (Process Management), Řízení projektů (Project Management), Návrh a realizace (Engineering), Podpůrné procesy (Support). Každá procesní oblast obsahuje následující informativní komponenty – účel (Purpose Statement), úvodní poznámky (Introductory Notes) a příbuzné procesní oblasti (Related Process Areas) (12).

2.3.3 Specifické a generické cíle

Specifické cíle popisují unikátní charakteristiky, které musí být přítomny pro uspokojení procesní oblasti. Jak už bylo zmíněno, tento komponent patří do kategorie očekávaných komponentů (12).

Generické cíle naopak pokrývají vícero procesních oblastí. Tento komponent se nachází v kategorii vyžadovaných komponentů a je nezbytný k institucionalizaci procesů v dané procesní oblasti (11).

2.3.4 Specifické a generické praktiky

Specifické praktiky (Specific Practices) jsou v podstatě popisy aktivit důležitých k dosažení specifického cíle. Tyto praktiky patří mezi očekávané komponenty (11).

Generické praktiky, stejně jako generické cíle, jsou aplikovány na vícero procesních oblastí. Představuje popis aktivit důležitých k dosažení generického cíle (12).

2.3.5 Úrovně způsobilosti a vyspělosti

K implementaci modelu CMMI-DEV nabízí dvě reprezentace – stupňovitou a kontinuální. Stupňovitá reprezentace odpovídá úrovním vyspělosti a kontinuální reprezentace úrovním způsobilosti. Tyto úrovně slouží k ohodnocení organizace. Pro dosažení určité úrovně, musí

organizace splnit všechny cíle dané procesní oblasti, či určité skupiny procesních oblastí, které jsou určeny ke zdokonalení. Stupňovitá reprezentace používá úrovně vyspělosti jako charakteristiku celkového stavu procesů organizace, které se vztahují k celému modelu. Kontinuální naopak používá úrovně způsobilosti jako charakteristiku procesů v organizaci, které se vztahují k jednotlivým procesním oblastem. Jednotlivé úrovně popisují, v jakém stavu se nacházejí procesy v dané procesní oblasti či dané skupině procesních oblastí (11).

3 ANALÝZA SOUČASNÉHO STAVU

Kapitola analýzy současného stavu čerpá z teoretických poznatků, které byly uvedeny v předchozí kapitole této práce. Objektem analýzy je pobočka nadnárodní společnosti, která působí v různých oblastech inženýrství (letectví, doprava, automatizace, atd.). Konkrétně se jedná o pobočku mezinárodní společnosti Honeywell, která sídlí v České republice v Brně. Hlavním předmětem analýzy je vývojový tým této pobočky, který se zabývá vývojem mobilních aplikací pro letecký průmysl.

Součástí kapitoly je představení společnosti Honeywell ve světě a v České republice. Dále následuje představení hlavního subjektu diplomové práce – vývojovému týmu FSS. Další část se věnuje výčtem metodik, které jsou ve společnosti zavedeny a výrazně vybraný tým ovlivňují. Výrazná část je věnována metodě agilního přístupu k projektovému řízení SCRUM, která je v tomto týmu zavedena pro vývoj mobilních aplikací. Tato část obsahuje rozdělení týmů, jednotlivých SCRUM rolí, ceremonií a stav artefaktů. Dále je uveden seznam a popis jednotlivých nástrojů, sloužících pro podporu samotného projektového řízení a také vývoje samotného.

Poslední část analýzy je věnována stavu modelu CMMI v týmu FSS.

3.1 Představení společnosti

3.1.1 Honeywell ve světě

Honeywell International Inc. (dále jen jako Honeywell) je mezinárodní společností, jejíž kořeny sahají až do roku 1885. Pobočky má v 70 zemích světa a dohromady zaměstnává více než 129 000 zaměstnanců z celého světa. Sídlo společnosti se nachází v New Jersey, Spojených státech amerických. Společnost dosáhla v roce 2015 obratu 38 miliard dolarů. Honeywell je předním světovým dodavatelem a výrobcem širokého spektra vyspělých technologií. Společnost Honeywell svojí hlavní podnikatelskou činnost zaměřuje na tři segmenty – letectví, automatizace a řízení a speciální materiály a technologie (12).

V oblasti letectví je Honeywell hlavním dodavatelem letadlových motorů, integrované avioniky, systémových řešení a služeb a také široké škály různých produktů a služeb určených pro výrobce letadel, leteckých společností nebo pro armádu. Podnikatelská činnost společnosti Honeywell v oblasti automatizace a řízení se zabývá celosvětovým vývojem zařízení, softwaru a technologií, které jsou dodávány jako komplexní řešení pro běžné domácnosti či průmyslové budovy. Tyto produkty představují zabezpečovací systémy, systémy pro tepelnou regulaci v budovách, senzory a přístroje použitelné v medicíně. Činnost společnosti Honeywell v oblasti speciálních materiálů a technologií zahrnuje vývoj a výrobu moderních materiálů. Tato činnost klade důraz na vytváření technologií, které vedou ke snížení emisí, nebo které umožňují výrobu ekologických paliv (12).



Obrázek č. 6: Oficiální logo společnosti Honeywell, Zdroj:Převzato z (12)

3.1.2 Honeywell v České republice

V České republice Honeywell působí ve třech městech – Praha, Brno a Olomouc. Pobočka Honeywell v Praze zaměstnává v současné době kolem 700 zaměstnanců. Zaměstnanci v této pobočce reprezentují širokou škálu profesí v oblastech vědy a výzkumu, v leteckém odvětví a dále také v administrativě, nákupu, prodeje, v podnikových financích, zákaznické podpoře a v lidských zdrojích (13).

Pobočka Honeywell v Olomouci představuje výrobní závod, kde se vyrábějí a opravují statické plechové a žárové díly leteckých turbínových motorů z nerezavějící oceli a speciálních hliníkových, niklových, kobaltových a titanových slitin pro většinu motorů a energetických jednotek společnosti Honeywell. Tato výrobní pobočka Honeywell v Olomouci zaměstnává 1270 zaměstnanců (14).

V Brně Honeywell působí od roku 2003 jako vývojové centrum a je součástí globálního společenství vývojových center – Honeywell Technology Solutions (HTS). S počtem kolem 1500 zaměstnanců je také největší výzkumným a inovačním centrem ve Střední Evropě. V Brněnském výzkumném a inovačním centru se zabývají vývojem produktů a komplexních řešení v oblastech letecké techniky (simulace, vývoj a výzkum systémů řízení letu, displejů a navigačních systémů), automatizace a řízení (vývoj a testování produktů a řídicích systému pro použití v domácnostech i průmyslu), dopravní systémy (technický vývoj a testování turbodmychadel pro osobní i nákladní vozidla). Od roku 2001 se v Brně nachází výrobní závod, kde se vyrábí produkty v oblastech spalování, elektroniky, termoregulace a systémů s pitnou vodou (15).



Obrázek č. 7: Pobočka Honeywell v Brně, Zdroj: Převezato z (15)

3.1.3 Základní údaje

- **Obchodní název:** Honeywell, spol. s.r.o. – HTS CZ o.z.
- **Sídlo společnosti:** Tuřanka 100/1387, 627 00 Brno

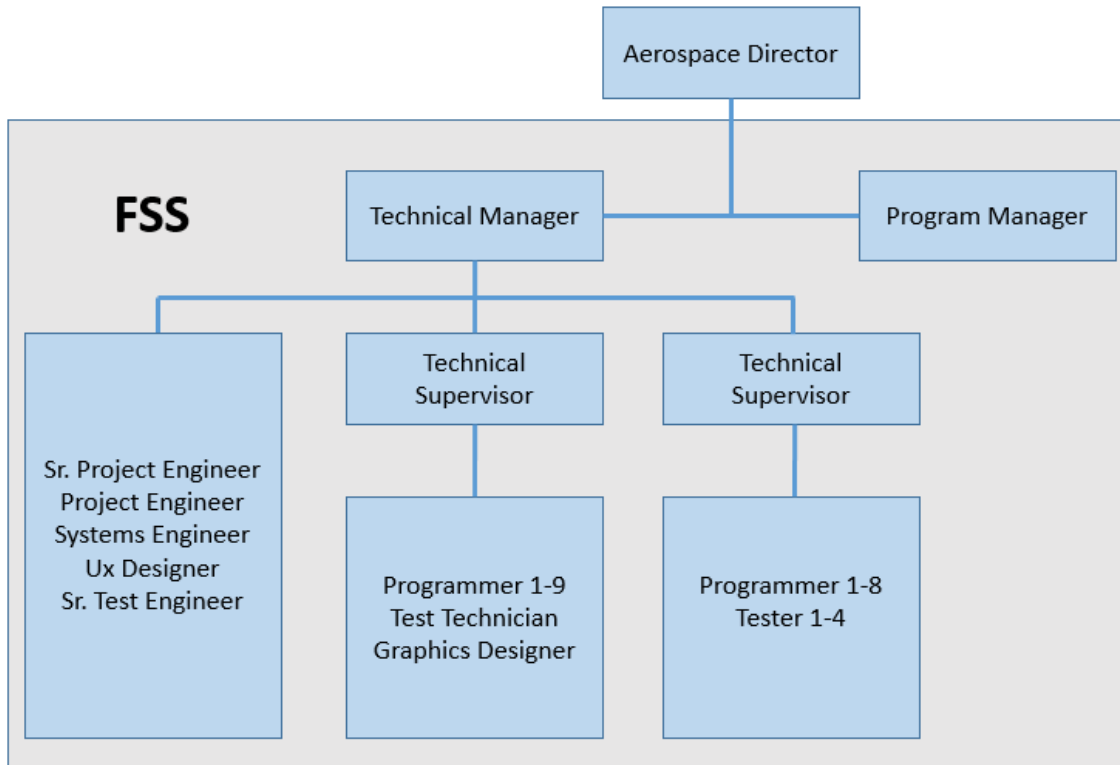
3.1.4 FSS

Předmětem analýzy a praktického řešení je v této diplomové práci vývojový tým FSS. FSS tým je součástí Honeywell Technology Solutions (HTS) v Brně. Hlavní činností týmu je vývoj softwarových aplikací určených pro mobilní zařízení. Tyto mobilní aplikace spadají do kategorie Electronic Flight Bag (EFB). Cílovými uživateli EFB aplikací jsou převážně piloti letadel. Těm mají aplikace sloužit k zjednodušení úkonů během letu (17).

Mobilní EFB aplikace, které tým FSS vyvíjí, jsou určeny pro piloty a slouží ke strategickému rozhodování během letu. Vzhledem k tomuto faktu, není vývoj těchto aplikací a následně i finální produkt zatížen různými regulacemi, certifikacemi a kontrolami. Na rozdíl od ostatních vývojových týmů vyvíjejících aplikace v Brně určené pro letecký průmysl vyvíjí FSS tým necertifikované aplikace (17).



Obrázek č. 8 Logo FSS, Zdroj: Upraveno dle (16)



Obrázek č. 9: Organizační struktura FSS týmu, Zdroj: Upraveno dle (16)

Na obrázku č. 9 lze vidět organizační strukturu týmu FSS. V tomto týmu jsou zastoupeny všechny důležité role pro vývoj softwaru.

Weather Information Service

Výsledný produkt jednoho z projektů, na kterém se FSS tým podílí. Tato mobilní aplikace je určena pro mobilní zařízení s operačními systémy iOS a Windows a slouží pilotům pro strategické rozhodování během letu. Aplikace poskytuje reálný, aktuální stav o počasí a také předpokládaný vývoj tohoto počasí v následujících hodinách. Pilot má k dispozici různé data o počasí, například oblasti, v kterých se vyskytují turbulence či aktuální stav počasí na cílovém letišti. Díky těmto informacím se může pilot vyhnout nebezpečným úsekům, které se vyskytují na jeho letové trase, tím se zvýší bezpečnost a také pohodlí cestujících. Tato

aplikace je určena pro zákazníky v sektoru ATR (Air Transport and Regional aviation) a BGA (Business and General Aviation) (17).



Obrázek č. 10: Weather Information Service, Zdroj: Upraveno dle (16)

3.2 Strategická analýza společnosti

Tato část práce je zaměřena na strategickou analýzu pobočky Honeywell, HTS CZ v České republice v Brně. Kapitola obsahuje analýzu okolí zmíněné pobočky (SLEPT), dále následuje analýza 7S, která analyzuje kritické faktory ve společnosti Honeywell. Jako poslední je uvedena analýza SWOT, která je zaměřena na interní analýzu týmu FSS.

3.2.1 Analýza okolí SLEPT

Tato část se věnuje analýze okolí společnosti Honeywell působící v Brně, analyzováno je 5 faktorů, které zasahují do činnosti této organizace v České republice.

Sociální faktory

Honeywell byl v roce 2016 vyhodnocen jako 3. nejžádanější zaměstnavatel v kategorii technických oborů v České republice. V Brně se nachází kvalitní vysokoškolské univerzity, jejichž absolventi mohou najít uplatnění ve společnosti Honeywell. Honeywell v Brně poptává nejvíce absolventů ze Strojírenské fakulty Vysokého učení v Brně a dále z řad absolventů informatiky, informačních technologií a také oborů managementu. Brno jako druhé největší město v České republice s krátkou vzdáleností do Prahy či Vídně, nabízí pro zaměstnance vhodné místo pro život, spolu s bohatým kulturním a sociálním životem (17).

Legislativní faktory

Z důvodu širokého portfolia produktů v mnoha různých technologických oblastech je společnost Honeywell zatížena velkým množstvím legislativních omezení. Vzhledem k mezinárodní působnosti nejsou tyto legislativní omezení pouze ze strany České republiky, ale i celosvětově. Velký důraz je kladen na to, zda vytvořený produkt neporušuje existující ochranné známky či patenty (17).

Ekonomické faktory

Současná dobrá ekonomická situace umožňuje společnosti Honeywell investovat do vývoje a výroby nových atraktivních produktů v různých oblastech, které mohou najít své místo na trhu. Díky vysoké diverzifikaci produktů společnosti Honeywell do různých technologických oblastí, je dopad, případného budoucího negativního vývoje ekonomické situace, na tuto společnost mírnější. Změny v ekonomické situaci v České republice mohou mít určitý vliv na pobočku Honeywell v Brně, tyto změny však výrazně neovlivní její tamější působení (17).

Politické faktory

Společnosti na celém světě jsou ovlivňovány politickými faktory daných zemí. Nejvýraznějším politickým faktorem, který ovlivňuje působení pobočky Honeywell v Brně je zahraniční politika současné a budoucí vlády České republiky (17).

Technologické faktory

Pro udržení konkurenceschopnosti společnosti musí Honeywell sledovat aktuální technologické trendy nejen v sektorech hlavní podnikatelské činnosti, ale také v oblastech, které na tyto sektory působí, nebo je ovlivňují. Na tento vývoj musí Honeywell reagovat. Celosvětové působení umožňuje vysoké sdílení technologických znalostí v rámci celé organizace (17).

3.2.2 Analýza 7S

Strategie

Pobočka Honeywell v Brně sdílí celopodnikovou strategii dodávat produkty a služby nejvyšší kvality, efektivitu, přinášející zákazníkům co nejvyšší hodnotu. Tato strategie se zaměřuje na využití User Experience pro dosažení souladu s potřebami zákazníků a také implementaci modelu CMMI, který slouží ke zkvalitnění procesů a tím ke kvalitnějším produktům (17).

Struktura

Zaměstnanci Honeywell v Brně jsou rozděleni do jednotlivých již zmíněných oblastí působnosti (Aerospace, ACS, atd.), kde se dále dělí do různých skupin. Z těchto skupin jsou tvořeny projektové týmy, které pracují na konkrétním projektu. Každá hlavní oblast má svého vedoucího, který se nachází pod prezidentem celé Brněnské pobočky. Součástí organizační struktury jsou také oddělení, které nespádají do hlavní činnosti působení společnosti Honeywell, ale tuto činnost jakýmsi způsobem podporují, tyto oddělení představují oddělení účetnictví, personální oddělení, finanční oddělení a další (17).

Systémy

V organizaci Honeywell je vysoké množství různých systémů pro podporu jednotlivých činností. Některé systémy jsou aplikovány na všechna oddělení stejně, existují zde však individuální systémy, které jsou z důvodů odlišnosti oddělení aplikovány pouze na některých místech (17).

Styl

Vztah nadřízeného k podřízenému je postaven na vzájemné důvěře. Pracovníci jsou nabádáni k otevřenosti, aby se předcházelo možným konfliktům. Všichni zaměstnanci si také každý rok stanovují své osobní cíle, které slouží ke zdokonalení jejich dovedností a rozvoj schopností. Pracovníci jsou také pravidelně informováni o vývoji a změnách ve společnosti Honeywell (17).

Spolupracovníci

Při výběru pracovníka je kladen důraz na jeho technické znalosti a určenou specializaci pro cílový tým. Velkou skupinu pracovníků tvoří absolventi z technologických univerzit. Jednotlivé skupiny či týmy pracovníků jsou umístěny v otevřených prostorech (openspace). S cílem stmelení kolektivu jsou pravidelně organizovány teambuildingové aktivity (17).

Sdílené hodnoty

Velký důraz je kladen na diverzitu, tolerantní a slušné chování pracovníků. Vize, cíl a hodnoty společnosti jsou pravidelně předávány formou velkých ceremonií zvaných Town Hall meetings (17).

Schopnosti

Jak už bylo zmíněno, zaměstnanec si každoročně za podpory svého liniového manažera stanovuje cíle, které mu pomáhají k rozvíjení jeho technických schopností a dovedností. K naplnění tohoto profesního rozvoje je k dispozici rozsáhlé spektrum školení a možností certifikace. Každý zaměstnanec má také možnost zlepšit své dovednosti v cizím jazyce, buď v rámci individuální výuky či návštěvou jazykové školy (17).

3.2.3 Analýza SWOT

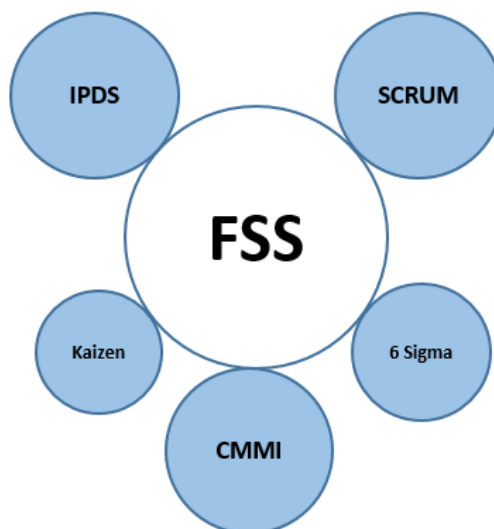
Dále následuje SWOT analýza, která se zaměřuje na silné a slabé stránky, příležitosti a hrozby vývojového týmu FSS v Brně (17).

Silné stránky <ul style="list-style-type: none">• Atraktivní projekty• Tým složen z expertů z různých oblastí• Spolupráce jednotlivých členů• Motivace členů podílet se na stanovení relevantních procesů• Spolupráce s významným výrobcem letadel• Vymezení z tradičního manažerského prostředí	Slabé stránky <ul style="list-style-type: none">• Model CMMI• Nedostatek lidských zdrojů• Automatické testování• Code Review• Provázanost mezi projekty• Continuous Integration• Zralost týmu
Příležitosti <ul style="list-style-type: none">• Zlepšení metodiky SCRUM pomocí externího trenéra• Úspěšná propagace produktů na EFB fóru• Spolupráce s velkou aerolinkou	Hrozby <ul style="list-style-type: none">• Neúspěch produktů• Neschválení Apple Review• Nedostatečná specifikace požadavků na produkt• Pokles motivace při neúspěchu produktu

Tabulka č. 1: SWOT analýza týmu FSS, Zdroj: Vlastní zpracování

3.3 Metodiky a rámce

V této kapitole se nachází analýza jednotlivých metodik a rámců, které výrazně ovlivňují působení týmu FSS.



Obrázek č. 11: Metodiky a rámce týmu FSS, Zdroj: Vlastní zpracování

3.3.1 Kaizen

Jeden ze základních procesů, který FSS tým následuje je proces neustálého zlepšování Kaizen. Kaizen představuje jednoduchý, malý krok ke zlepšení stávající situace. Kaizen volně v překladu z Japonštiny znamená „Změna k lepšímu“. Je to změna myšlení, kdy zaměstnanec kontinuálně přemýšlí a hledá způsob ke zdokonalení současného procesu či vykonávané práce. Kaizen má následující charakteristiky (16):

- Malý, inkrementální
- Minimální náklady, nízká hodnota rizika, kreativní zlepšení
- Jednoduchá implementace

V Honeywellu tento proces probíhá v několika krocích. V prvním kroku zaměstnanec při své každodenní pracovní činnosti identifikuje příležitost ke zlepšení - Kaizen. Tento Kaizen popíše a vloží do informačního systému k jeho akceptaci. Akceptaci vytvořeného Kaizenu

provádí project engineer nebo liniový manažer, po akceptaci je tento Kaizen implementován. Výsledky zlepšení zavedeného Kaizenu jsou po určité době zhodnoceny a je rozhodnuto o jeho využitelnosti i v ostatních týmech (16).

3.3.2 Six Sigma

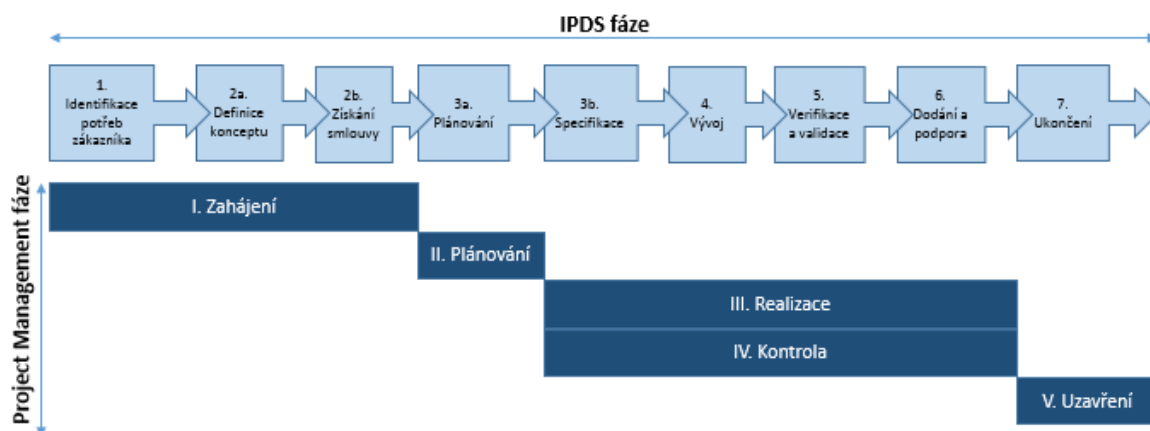
Six Sigma představuje ve společnosti Honeywell zavedenou strategii s cílem zlepšení procesů, produktů a služeb a jejich kvality a v neposlední řadě také ke snížení výrobních a administrativních nákladů. Six Sigma představuje základní zásady, kterými by se měl každý zaměstnanec společnosti Honeywell řídit (16):

- Eliminace pracovních kroků, které nepřinášejí hodnotu
- Snížení nároků na zdroje a podporu
- Snížení variací procesu
- Porozumění zákazníka a zaměření se na jeho uspokojení

Strategie Six Sigma je aplikována napříč celou organizací Honeywell a každý zaměstnanec je povinen se naučit přístupy a nástroje Six Sigmy. Povinná je certifikace Green Belt každého zaměstnance. Tato certifikace představuje vypracování týmového projektu s cílem identifikovat problém, najít a implementovat jeho řešení. Tento projektový tým má svého sponzora, který mu pomáhá s pochopením jednotlivých nástrojů Six Sigmy. Výsledek projektu je prezentován zainteresovaným stranám a následně je rozhodnuto o výsledku certifikace (16).

3.3.3 IPDS

IPDS (Integrated Product Delivery and Support) představuje integrovaný proces pro vývoj produktů napříč celou organizací Honeywell. Tento proces definuje životní cyklus programu a jeho jednotlivé fáze spolu s nezbytnými artefakty, které do těchto fází vstupují a vystupují (16).



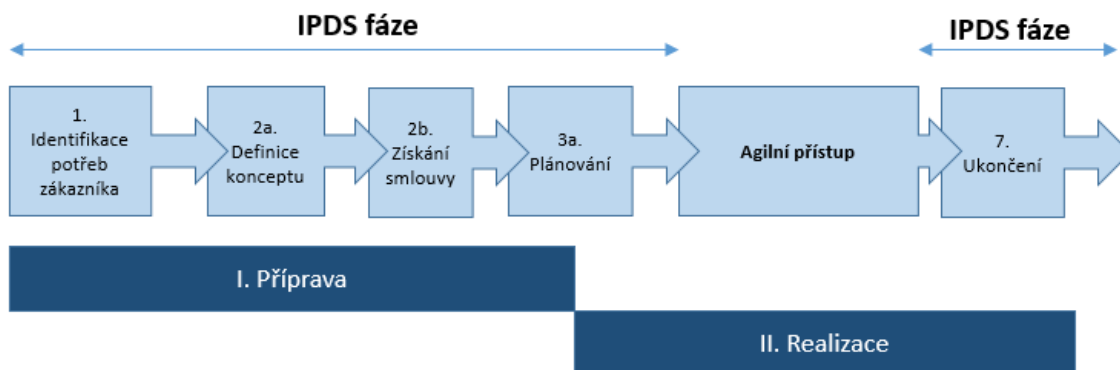
Obrázek č. 12: Fáze životního cyklu procesu IPDS, Zdroj: Upraveno dle (17)

Životní cyklus programu podle IPDS vychází z velké části z klasického pojetí životního cyklu projektu v projektového managementu. V obecném pohledu se zde vyskytují následující fáze: I. zahájení, II. plánování, III. a IV. realizace a kontrola, V. uzavření. Tyto fáze jsou v IPDS rozděleny na detailnější části (obrázek č. 12) : 1. identifikace potřeb zákazníka, 2a. definice konceptu, 2b. získání smlouvy, 3a. plánování, 3b. specifikace, 4. vývoj, 5. verifikace a validace, 6. dodání a podpora. 7. ukončení. Začátek jednotlivé IPDS fáze je možný až po schváleném ukončení fáze předcházející (mimo fázi č. 1). Úspěšnému dokončení jednotlivé fáze předchází splnění několika kroků. Každá fáze začíná samotným plánováním průběhu fáze, realizací potřebných činností a závěrečnou částí, v které se celá fáze hodnotí a schvaluje. Všechny fáze, procesy, činnosti a artefakty, které se vyskytují v IPDS jsou detailně popsány v dokumentech, s označením AP, AG (Aerospace Procedure, Aerospace Guideline). Pro schválení ukončení dané fáze je potřeba splnit ukončovací kritéria (Phase Exit Criteria). Tyto kritéria jsou ve většině případů ve formě artefaktů, které slouží

jako vstupy do fáze následující. Proces IPDS je stěžejním procesem pro všechny oddělení v sektoru Aerospace, vyjma oddělení, které jsou z použití tohoto procesu vyjmuty, či mají tento proces modifikován (16).

Proces IPDS představuje klasický vodopádový přístup k řízení projektu, kde jednotlivé fáze nemohou začít, dokud neskončí fáze předcházející. Tento princip je v rozporu s agilním přístupem, který tým FSS používá k vývoji svých produktů. Proto vznikla modifikace, která upravuje životní cyklus projektu v procesu IPDS a nabízí tak řešení pro týmy, které využívají agilní přístup. Tento proces však není dostatečně popsán a integrován do jednoho dokumentu. Část procesu popisuje **AG 5791**, k pochopení celého životního cyklu bylo potřeba diskuze s Project Engineerem. (17).

Celý životní cyklus je rozdělen na části (obrázek č. 13): I. příprava a II. realizace. V přípravné části najdeme fáze z klasického procesu IPDS - 1. identifikace potřeb zákazníka, 2a. definice konceptu, 2b. získání smlouvy, 3a. plánování. V části realizace nachází místo zvolená metodika agilního přístupu (16).



Obrázek č. 13: Životní cyklus procesu vývoje mobilních aplikací, Zdroj: Upraveno dle (16)

Jednotlivé fáze v části přípravy probíhají odděleně mimo FSS tým. Úkolem těchto činností je provést value analýzu. Na obecnější úrovni definovat co má být výsledkem tohoto

projekt, komu tento výstup bude sloužit, co to potenciálnímu zákazníkovi přinese, kolik to bude stát a jak dlouho projekt bude trvat (16).

Identifikace potřeb zákazníka

Identifikace tržních segmentů, kvantifikování požadavků zákazníka pro získání uceleného a přesného porozumění, definice hodnoty dané tržní příležitosti. Tuto fázi provádí oddělení M&PM (Marketing and Project Management) (16).

Definice konceptu

Definice funkčních požadavků pro vytvoření konceptu budoucího produktu. Dokončení business modelu a vytvoření obecného předběžného project management plánu (16).

Získání smlouvy

Finální definice funkčních požadavků na produkt. Stanovení nákladů a časového plánu vývoje produktu. Důležitou činností je vyjednávání o smlouvě s potenciálním zákazníkem a identifikace odchylek mezi nabídkou a poptávkou spolu s analýzou dopadů těchto odchylek na náklady a časový plán. Vzhledem k identifikovaným odchylkám je upraven project management plán (16).

Plánování

V této fázi se zavádí plán řízení rizik a stanovují se základní metriky pro sledování výkonnosti (rozsah projektu, časový plán, náklady). Hlavní činností této fáze je transformace informací z důležitých programových dokumentů (MRD – Market Requirements Document, SOW – Statement Of Work) do skupin aktivit, které jsou zachyceny v podobě WBS – Work Breakdown Structure, časového plánu a hlavních milníků programu (16).

Agilní přístup 4

Zde následuje fáze vývoje produktu s použitím agilního přístupu. FSS tým zavedl a současně využívá metodiku SCRUM, která je jednou z nejpoužívanějších metodik v agilním světě. Představení použití této metodiky ve FSS týmu se nachází v následující kapitole (16).

Uzavření

V této fázi dochází k uzavření projektu a jeho zhodnocení (16).

3.3.4 SCRUM

V této kapitole se nachází analýza současného použití metodiky SCRUM ve vývojovém týmu FSS. FSS tým tuto metodiku využívá již od samého počátku jeho vzniku.

Role

Celý FSS tým pracuje na několika projektech zároveň a je z tohoto důvodu rozdělen na tři SCRUM týmy. Každý tým má přidělen projekt, na který se zaměřuje s nejvyšší prioritou. Každý SCRUM tým se skládá z **Product Ownera**, **Scrum Mastera** a **vývojového týmu** (17).

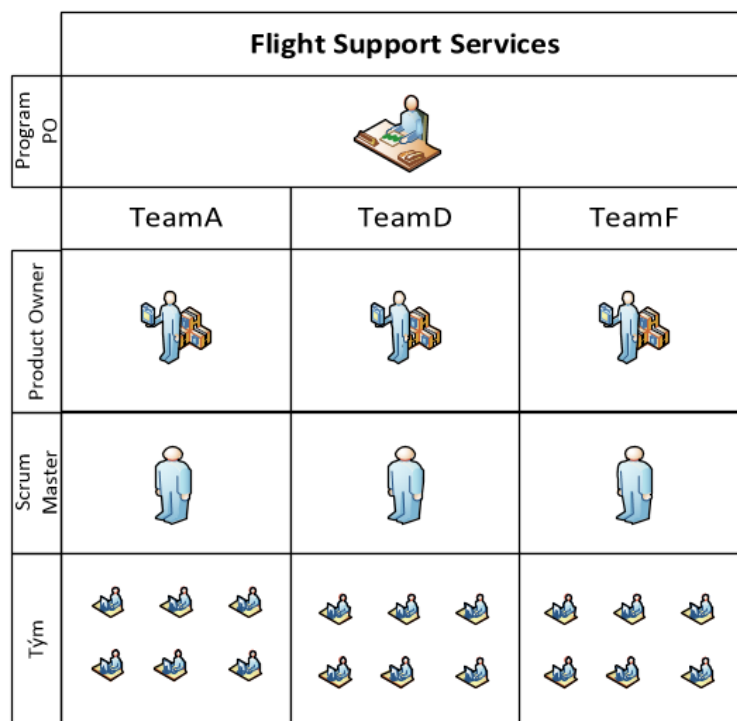
Vývojový tým je složen z odborníků různého zaměření. Tento tým tvoří z větší části zkušení programátoři, kteří svými dovednostmi dokáží pokrýt všechny úrovně vyvíjené aplikace – pozemní část, backend i frontend. Dále vývojový tým tvoří testéři, kteří jsou zodpovědní za testování vyvíjené aplikace, manuálně nebo automatickými testy. Poslední část vývojového týmu tvoří grafici, UX designéři či systémoví inženýři. Takto složené týmy jsou schopny dodat kompletní balík funkcionality svého vyvíjeného produktu (17).

SCRUM týmy mají přidělené Scrum Mastery, kteří jsou zodpovědní za coaching týmu, facilitaci jednotlivých ceremonií a odstraňování bariér, které se vývojovému týmu naskytnou. Scrum Masteri dohlížejí na správné pochopení a dodržování zavedené metodiky SCRUM (17).

Každý SCRUM tým má také svého Product Ownera, který naplňuje, spravuje a prioritizuje Product Backlog. Product Owneři komunikují se zákazníkem, ostatními stakeholdery a tvoří tak most mezi vývojovým týmem a businesssem (17).

Nad těmito třemi týmy stojí Program Manager, který zastává roli hlavního Product Ownera. Pro podporu administrativních či projektových záležitostí je týmům k dispozici liniový

Technical Manager a Project Engineer. Rozdělení FSS týmu na tři samostatné týmy z pohledu SCRUM lze vidět na obrázku č. 14 (16).



Obrázek č. 14: Struktura FSS týmu z pohledu SCRUMu, Zdroj: Upraveno dle (16)

Ceremonie

Zde následuje výpis jednotlivých ceremonií, které jsou zavedeny v týmu FSS. Ke klasickým ceremoniím, které představuje metodika SCRUM, jsou navíc zavedeny ceremonie, které se uskutečňují na odlišných úrovních managementu. Účastníci jsou rozdělení podle povinné (P) a nepovinné (N) účasti a podle toho kdo je realizátorem (R) ceremonie (16).

Název:	Level 1 Backlog Grooming		
Doba trvání:	4 hodiny	Frekvence:	kvartálně
Účastníci:	(R) Program Manager, (P) Product Owneři, (P) Technical Manager, (P) Project Engineer, (P) Stakeholders		
Popis:	Představení vize, představení požadavků na funkcionalitu, revize stávající kapacity týmu		

Tabulka č. 2: Level 1 Backlog Grooming, Zdroj: Upraveno dle (16)

Název:	Level 1 Program Planning		
Doba trvání:	2 hodiny	Frekvence:	kvartálně
Účastníci:	(R) Program Manager, (P) Product Owneři, (P) Technical Manager, (P) Project Engineer, (P) Stakeholders		
Popis:	Plánování založeno na stanovených estimacích.		

Tabulka č. 3: Level 1 Program Planning, Zdroj: Upraveno dle (16)

Název:	Level 2 Backlog Grooming		
Doba trvání:	2 hodiny	Frekvence:	každý sprint
Účastníci:	(R) Product Owner, (P) Scrum Master, (P) Tým		
Popis:	Představení vize, diskuze nad požadovanou funkcionalitou, estimace na obecné úrovni		

Tabulka č. 4: Level 2 Backlog Grooming, Zdroj: Upraveno dle (16)

Název:	Program Synchronization		
Doba trvání:	2 hodiny	Frekvence:	každý sprint
Účastníci:	(R) Program Manager, (N) Technical Manager, (P) Product Owner, (N) Scrum Master		
Popis:	Diskuze ohledně stavu jednotlivých zdrojů pro nadcházející sprint. Určení priority projektů. Distribuce práce ve SCRUM týmech.		

Tabulka č. 5: Program Synchronization, Zdroj: Upraveno dle (16)

Název:	Level 2 Sprint Planning		
Doba trvání:	2 hodiny	Frekvence:	každý sprint
Účastníci:	(P) Product Owner, (R) Scrum Master, (P) Tým		
Popis:	Představení a pochopení jednotlivých user stories pro následující sprint, definice cíle sprintu		

Tabulka č. 6: Level 2 Sprint Planning, Zdroj: Upraveno dle (16)

Název:	Level 3 Sprint Planning		
Doba trvání:	2 hodiny	Frekvence:	každý sprint
Účastníci:	(N) Product Owner, (R) Scrum Master, (P) Tým		
Popis:	Dekompozice jednotlivých user stories na menší úkoly, tým tvoří detailnější estimace jednotlivých úkolů, vytvoření sprint backlogu, estimace v jednotkách času do kapacity celého týmu		

Tabulka č. 7: Level 3 Sprint Planning, Zdroj: Upraveno dle (16)

Název:	Sprint Demo		
Doba trvání:	2 hodiny	Frekvence:	každý sprint
Účastníci:	(P) Product Owner, (R) Scrum Master, (P) Tým, (N) Stakeholders		
Popis:	Představení dokončené práce v daném sprintu, akceptace user stories product ownerem		

Tabulka č. 8: Sprint Demo, Zdroj: Upraveno dle (16)

Název:	Program Sprint Demo		
Doba trvání:	1 hodina	Frekvence:	každý sprint
Účastníci:	(R) Program Manager, (P) Product Owner, (P) Stakeholders		
Popis:	Představení hlavních nových funkcionalit na vyšší úrovni		

Tabulka č. 9: Program Sprint Demo, Upraveno dle (16)

Název:	Sprint Retrospective		
Doba trvání:	1 hodina	Frekvence:	každý sprint
Účastníci:	(N) Product Owner, (R) Scrum Master, (P) Tým		
Popis:	Rekapitulace ukončeného sprintu, identifikace zásadních problémů/úspěchů, návrhy na zlepšení procesů		

Tabulka č. 10: Sprint Retrospective, Zdroj: Upraveno dle (16)

Název:	Daily Scrum		
Doba trvání:	15 minut	Frekvence:	každý den
Účastníci:	(N) Product Owner, (R) Scrum Master, (P) Tým		
Popis:	Identifikace překážek, report stavu sprintu a rozpracovaných úkolů		

Tabulka č. 11: Daily Scrum, Zdroj: Upraveno dle (16)

Název:	Daily Scrums of Scrum		
Doba trvání:	15 minut	Frekvence:	každý den
Účastníci:	(R) Program Manager, (N) Technical Manager, (P) Product Owner, (N) Scrum Master		
Popis:	Rychlá synchronizace mezi týmy, reporting na vyšší úrovni		

Tabulka č. 12: Daily Scrums of Scrum, Zdroj: Upraveno dle (16)

Pořadí jednotlivých ceremonií na úrovni týmu, během každého sprintu, vychází z klasického pojetí metody SCRUM. Každý sprint začíná ceremonií Level 2 Sprint Planning a Level 3 Sprint Planning, kde Product Owner představí jednotlivé User Stories, které mají být pracovní náplní nadcházejícího sprintu, tyto User Stories jsou diskutovány, dokud nemá tým všechny potřebné informace. Product Owner dále představí hlavní cíl nadcházejícího sprintu. V další části tým rozbije User Stories na menší úkoly a provede detailní odhad časové náročnosti úkolů. Na konci této ceremonie vytvoří tým Sprint Backlog a zaváže se k jeho splnění (16).

V průběhu sprintu je každý den pořádán 15 minutový meeting Daily Scum za účelem získání přehledu o stavu práce a identifikaci případných překážek, které zabraňují plynulý průchod sprintem (16).

V polovině sprintu se uskuteční meeting, zvaný Level 2 Backlog Grooming, kde Product Owner prezentuje vizi na několik nadcházejících sprintů, dále je představena budoucí funkcionality, u které se týmem odhaduje její přibližná časová náročnost (16).

Sprint je zakončen ceremonií zvanou Sprint Demo, kde tým prezentuje dokončenou práci v rámci sprintu a Product Owner tuto práci podle splněných akceptačních kritérií a stanovených „definition of done“ akceptuje. Dokončení samotné User Story neznamena pouze její naprogramování programátorem, ale i její důkladné otestování (16).

Poslední ceremonií je Sprint Retrospective, kde tým diskutuje možné zlepšení procesu, rozebírá vzniklé problémy nebo také úspěchy (16).

Artefakty

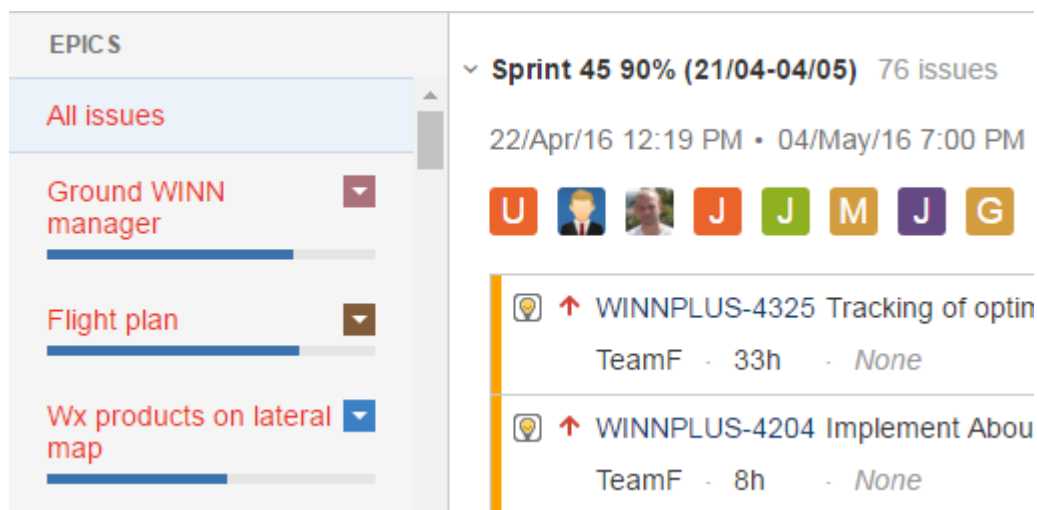
V této kapitole jsou uvedeny zavedené SCRUM artefakty.

Sprint

Délka sprintu je pro všechny SCRUM týmy ve FSS stejná. Už od samotného počátku zavedení agilního přístupu se pracuje ve dvoutýdenních sprintech. Týmy jsou schopné za takto stanovenou délku sprintu dodat potencionální přírůstek funkcionality (16).

Epic

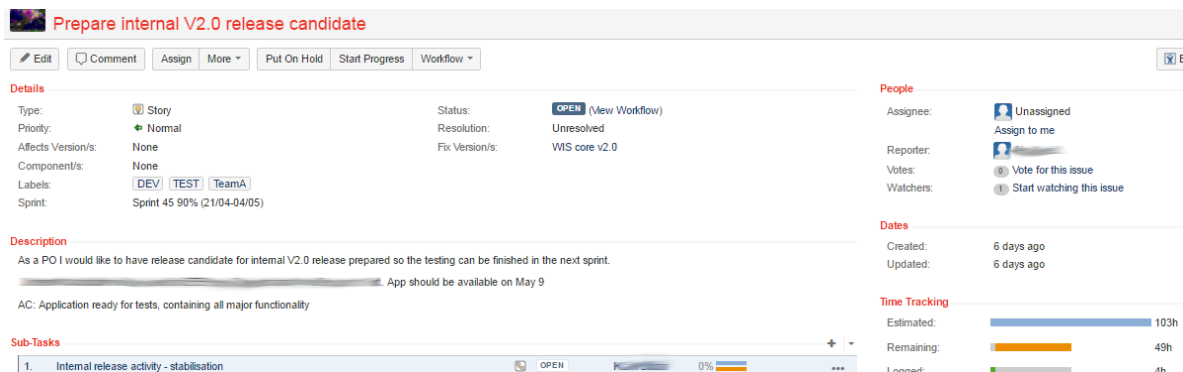
Na programové úrovni jsou stanoveny funkční požadavky na jednotlivé produkty, které vznikly z průzkumu trhu či z potřeb zákazníka. Tyto požadavky se transformují do Epicu – velký pracovní balík, představující celkovou funkcionalitu. Tento pracovní balík obsahuje dílčí úkoly, které jsou nezbytné pro jeho dokončení. Pro sledování a spravování Epiců (i dalších artefaktů) slouží FSS týmu systém podpory agilního vývoje - JIRA. Tento nástroj je vyvíjen společností Atlassian a je v Honeywellu hojně využíván napříč celou organizací. V tomto systému lze přehledně vidět, v jakém stádiu se daný Epic nachází, respektive kolik práce je potřeba udělat pro jeho dokončení (16).



Obrázek č. 15: Epic v systému JIRA, Zdroj: Vlastní zpracování

User Story

Jednotlivé User Story jsou v JIRA zaznamenány ve formátu: „*Jako **ROLE** chci **CÍL** tak, aby **UŽITEK**.*“. Během Sprint Planning se tato User Story rozbije na dílčí úkoly a časově se ohodnotí. User Story také obsahuje nezbytně nutné informace k jejímu dokončení a akceptační kritéria, podle kterých nakonec Product Owner posuzuje, zda byla práce splněna či naopak (16).

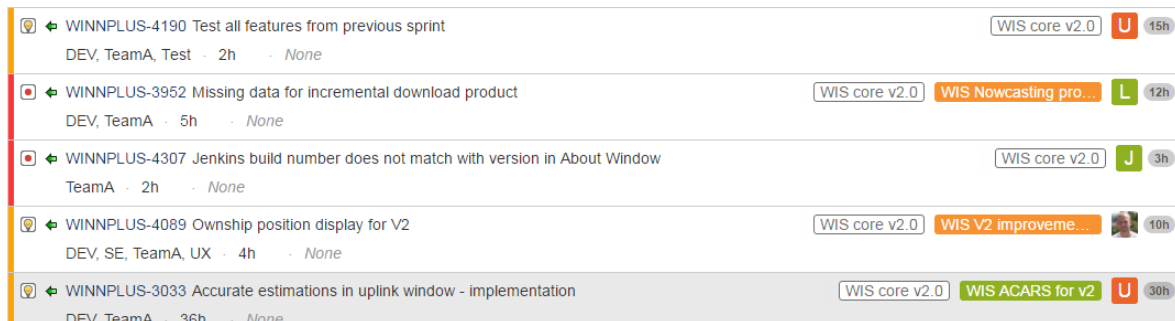


The screenshot shows a JIRA issue page for the User Story "Prepare internal V2.0 release candidate". The issue is in the "OPEN" status, with a priority of "Normal". It is assigned to "Unassigned" and has a reporter of "Assign to me". The description states: "As a PO I would like to have release candidate for internal V2.0 release prepared so the testing can be finished in the next sprint. App should be available on May 9". The sub-tasks section shows one task: "1. Internal release activity - stabilisation" with a progress bar at 0%. The time tracking section shows an estimated time of 103h, with 49h remaining and 4h logged.

Obrázek č. 16: User Story v systému JIRA, Zdroj: Vlastní zpracování

Product Backlog 5

Product Backlog obsahuje všechny User Stories, úkoly či identifikované bugy definovány nejen Product Ownerem, ale i celým týmem. Je však zodpovědností Product Ownera tento backlog spravovat a prioritizovat. Tento Product Backlog slouží jako vstup do Sprint Planning meetingu, kde jsou nad těmito úkoly (User Stories), vedeny diskuze a následně je vytvořen Sprint Backlog (16).



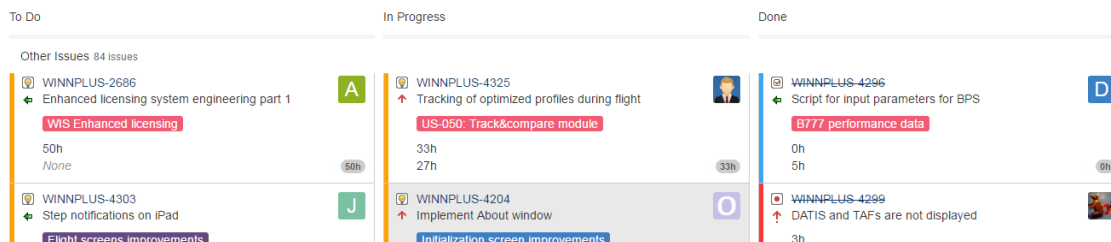
The screenshot shows a JIRA Product Backlog with five items. Each item includes a key, a title, assignee, team, time estimate, priority, and version. The items are:

- WINNPLUS-4190 Test all features from previous sprint (2h, None, WIS core v2.0, U, 15h)
- WINNPLUS-3952 Missing data for incremental download product (5h, None, WIS core v2.0, WIS Nowcasting pro..., L, 12h)
- WINNPLUS-4307 Jenkins build number does not match with version in About Window (2h, None, TeamA, WIS core v2.0, J, 3h)
- WINNPLUS-4089 Ownship position display for V2 (4h, None, DEV, SE, TeamA, UX, WIS core v2.0, WIS V2 improve..., 10h)
- WINNPLUS-3033 Accurate estimations in uplink window - implementation (36h, None, DEV, TeamA, WIS core v2.0, WIS ACARS for v2, U, 30h)

Obrázek č. 17: Product Backlog v systému JIRA, Zdroj: Vlastní zpracování

Sprint Backlog

Sprint Backlog, který je výstupem Sprint Planning meetingu, je během sprintu zobrazen na virtuální tabuli v systému JIRA. Tato tabule obsahuje tři stupně, ve kterých se pracovní balík může nacházet. První sloupec „To Do“ představuje stav, ve kterém se nachází práce, na které se ještě nezačalo pracovat. Stav „In Progress“ značí právě rozdělanou práci a stav „Done“ práci, která je hotova a čeká na akceptaci Product Ownerem. Každý člen týmu je zodpovědný za aktualizaci stavu práce, na které během sprintu pracuje. Členové týmu také aktualizují zbývající čas, který je přibližně potřeba k dokončení úkolu (16).



Obrázek č. 18: Sprint Backlog v systému JIRA, Zdroj: Vlastní zpracování

Agilní praktiky

Agilní praktiky slouží k podpoře agilního vývoje, jejich implementace pomáhá ke zvýšení kvality produktu či transparentnosti celého vývoje. V této části se nachází výčet agilních praktik, které jsou v týmu FSS využívány.

Coding guidelines

Coding Guidelines představují ucelený soubor doporučení, rad i povinností, kterými by se měl každý programátor v týmu FSS při tvorbě kódu řídit. Coding Guidelines neobsahují pouze pravidla jak formátovat kód či pojmenovávat funkce nebo třídy, ale i důležité pravidla popisující, které programovací techniky jsou preferovány, a kterým by se měl programátor vyvarovat. Takto psaný kód je pak přehlednější a lépe udržovatelný (17).

Continuous Integration

Continuous Integration můžeme chápat jako systém integrace v reálném čase. Skládá se z jednotlivých bloků, které na sebe navazují. Jejich společným cílem je vytvoření spustitelné aplikace. Tým FSS používá verzovací systém GIT, kde se nachází zdrojový kód aplikace. Programátoři do tohoto systému přidávají svou hotovou část kódu. Systém pak udržuje pořádek v jednotlivých verzích a eviduje historii změn. Další části Continuous Integration jsou automatické testy, které testují jednotlivé části kódu. Poslední částí je nástroj, který vytáhne zdrojový kód z repositáře, přeloží ho a spustí automatické testy. Výsledky tohoto procesu pak reportuje příslušným členům týmu. Tým FSS používá webový nástroj Jenkins pro pokrytí tohoto procesu. Jenkins tento proces spouští každý den v nočních hodinách a výsledky reportuje pomocí webového dashboardu. Tým má tak každý den k dispozici spustitelnou aplikaci, která obsahuje všechny změny předešlého dne, nebo je ihned informován o neúspěšném překladu aplikace z důvodu posledních změn (17).

Sdílený kód

Sdílený kód v týmu FSS představuje uložení veškerých zdrojových souborů ve verzovacím systému GIT, jsou k dispozici všem členům týmu a ti tak mohou změnit jakoukoliv část. Aby tato praktika fungovala je potřeba jí doplnit garancí za určité části kódu a vytvořením procesu Code Review. Aktuálně FSS tým nemá tyto dva kroky nastaveny. Každý člen tak může měnit jakoukoli část kódu bez kontroly a potvrzení jejího garanta (17).

Refaktoring

Refaktoring představuje úpravu stávajícího zdrojového kódu. Tato úprava spočívá ve vylepšování vnitřní struktury kódu pomocí malých změn. Při plánování sprintu vyhrazuje každý člen týmu FSS určitou část své kapacity na tuto aktivitu. Nejčastěji tento čas zahrnuje již do časového ohodnocení jednotlivých User Stories (17).

Application release

Kdy se bude aplikace vydávat je určeno Release Planningem, kde je stanoveno datum vydání a požadovaná funkcionálnita. Před samotným vydáním aplikace probíhá tzv. stabilizační sprint, kde je již požadovaná funkcionálnita dokončena a aplikace se pouze testuje, případně jsou opravovány nalezené bugy. V tomto sprintu také probíhá dokončení uživatelského manuálu a dalších nezbytných dokumentů. Na konci sprintu se release kandidát pošle na interní recenzi k UAT (User Acceptance Testing). V případě, že UAT akceptuje release kandidáta, je tento kandidát poslán na další nezbytnou recenzi, a to hlavním stakeholderům projektu. Pro akceptaci je potřeba písemného souhlasu s vydáním aplikace. V posledním kroku je akceptovaný release kandidát vydán na AppleStore v případě iOS verze, či na webové stránky produktu v případě verze pro operační systém Windows (16).

Podpora zákazníka

Tento proces má zajistit včasné odhalení případného problému vydané aplikace či podporu zákazníka, který tuto aplikaci využívá. Podpora zákazníka následuje metodu follow the sun. Jsou vyčleněny tři úrovně oddělení podpory (16).

- **Úroveň 1:** GDC Ops - USA 24/7 – Podpora zákazníka při stažení, instalaci, konfiguraci a používání aplikace. Řeší běžné problémy spojené s použitím aplikace. Problémy jsou evidovány v systému Salesforce
- **Úroveň 2:** CZ/Indie/USA 8/5 – Řešení komplexních problémů, problémů, které nedokáže vyřešit úroveň 1. Evidence v systému Salesforce
- **Úroveň 3:** CZ – vývojový tým 8/5 – Oprava bugů v aplikaci hlášených zákazníkem. Evidence v systému JIRA.

Monitorování a kontrola 4

Během agilního vývoje je potřeba sledovat stav projektu, pro tyto účely se pravidelně schází kontrolní skupina, která projednává (16):

- zdraví projektu
- rozpočet
- rizika
- stav zdrojů a jejich potřeba
- milníky

Softwarové nástroje

Tato kapitola obsahuje stručný pohled na softwarové nástroje, které jsou využívány v FSS týmu pro podporu vývoje či vývoj samotný.

JIRA

Jak už bylo zmíněno tento nástroj je v týmu FSS využíván jako kompletní podpora agilního přístupu k vývoji aplikací. Tento nástroj eviduje všechny User Stories projektu, spolu s jejich umístěním v Product Backlogu a zařazením do Epicu. Dále nabízí virtuální nástěnku Sprint Backlogu, která reflektuje práci v daném sprintu a její stav. Nástroj umožňuje několik možností reportingu, jako je například Burndown Chart, Velocity Chart a další (17).

Microsoft Sharepoint

Tato webová služba nabízí úložiště projektové dokumentace. Jsou zde uloženy důležité dokumenty vycházející z IPDS fází nebo dokumenty partnerů či zákazníků. Sharepoint nabízí také další webové nástroje, jako jsou například Meeting Minutes (strukturované zápisy z meetingů spolu se synchronizací stanovených úkolů do emailového klienta účastníka), nastavení workflow dokumentů, či dashboard pro reporting sledovaných metrik jednotlivých týmů (17).

Confluence Wiki

Webová služba používaná především vývojovým týmem. Obsahuje základní, stručnou dokumentaci technické stránky projektu (17).

TestLink

System pro podporu testování aplikace. Obsahuje jednotlivé testovací scénáře, testovací plány a velikou řadu testovacích reportů (17).

3.3.5 CMMI

Tým FSS následuje proces vývoje mobilních aplikací, který je modifikací procesu IPDS. Takto modifikovaný proces se skládá z prvních základních fází procesu IPDS, dále přechází do fáze vývoje, která je realizována pomocí metody SCRUM a následně je celý proces ukončen fází uzavření z procesu IPDS. Během celého projektu je následováno velké množství dalších podpůrných procesů, které podporují realizaci projektu. Dokument **AG 5791** slouží k popisu celého procesu vývoje mobilních aplikací, spolu s jeho podpůrnými procesy. Tento dokument je však v současné době **nedostatečný a zastaralý** (17).

Aktuální stav CMMI je neznámý. Vedení týmu nemá dostatečnou znalost modelu CMMI a není schopno určit, jaké jsou jeho požadavky, a které z využívaných procesů či metodik mohou pokrýt specifické praktiky uvedené v CMMI.

3.4 Závěr analýzy

V této podkapitole následuje stručné shrnutí výstupů z provedené analýzy vývojového týmu FSS:

- Modifikace procesu vývoje produktů, který vychází z klasického pojetí projektového managementu
- Pro fázi vývoje je aplikována metoda SCRUM, spolu s rozdělením týmů, jejich rolí, nastavení ceremonií, artefaktů a použití agilních praktik
- Existují podpůrné procesy, které jsou v týmu využívány
- Nedostatečná dokumentace těchto procesů a použité metodiky SCRUM
- Neznalost modelu CMMI a jeho požadavků
- Neschopnost popsat vzájemnou spolupráci zavedených metod, procesů pro pokrytí specifických praktik modelu CMMI

Z analýzy vyplývá, že tým FSS má dostatečně implementovanou agilní metodiku SCRUM, kterou doplňuje klasickými procesy z IPDS. Tato modifikace však není dostatečně zdokumentována. Z analýzy modelu CMMI, vyplývá, že způsob pokrytí specifických praktik není znám. Je úkolem této diplomové práce navrhnout řešení, jak zavedené metody a procesy mohou pokrýt tyto praktiky vzájemným působením a také navrhnout nové řešení v oblastech které nepokrývají.

4 NÁVRH ŘEŠENÍ A PŘÍNOS NÁVRHU ŘEŠENÍ

V části vlastního řešení se nacházejí jednotlivé procesní oblasti, jejich cíle a praktiky pro zaručení splnění těchto cílů. Každá specifická praktika obsahuje návrh řešení jak tuto praktiku pokrýt agilní metodou SCRUM, vybraným procesem IPDS, nebo zavedením zcela nové praktiky, respektive procesu. Konkrétně jsou uvedeny výstupy, které slouží jako artefakt pro posouzení splnění jednotlivé praktiky.

Stručný výstup vlastního řešení představuje tabulka jednotlivých procesních oblastí, cílů a jejich praktik spolu s vyznačením řešení způsobu pokrytí, buďto metodikou SCRUM nebo IPDS procesem, a zda je pokrytí této praktiky řádně zdokumentováno.

Součástí vlastního řešení je také vypracování ekonomického zhodnocení zavedení navrženého řešení. Poslední část je věnována přínosu práce a jejího reálného využití v praxi.

4.1 CMMI procesní oblasti

Tato kapitola obsahuje návrh vlastního řešení pokrytí jednotlivých specifických praktik, jejichž přítomnost je očekávaná pro splnění specifických cílů daných procesních oblastí. Kapitola je rozdělena na vybrané procesní oblasti, které uvádějí specifické cíle a jejich specifické praktiky. Každá specifická praktika obsahuje návrh řešení, konkrétní praktiku a její výstup.

4.1.1 Requirements Management (REQM)

Účelem této procesní oblasti je řízení požadavků produktu a jeho komponent. Zajištění provázanosti mezi těmito požadavky, projektovým plánem a produktem (11).

SG 1 Manage Requirements / Spravujte požadavky

Požadavky jsou řízeny a nekonzistence mezi těmito požadavky, projektovým plánem a produktem jsou identifikovány (11).

SP 1.1 Understand Requirements / Pochopte požadavky

Získejte porozumění produktových požadavků (11).

Tato praktika lze pokrýt metodou SCRUM. Je zodpovědností Product Owenera identifikovat relevantní zdroj informací pro definici produktových požadavků. Product Owner je také zodpovědný za propagování a vysvětlení těchto požadavků vývojovému týmu. Ve SCRUMu jsou požadavky v průběhu času detailně definovány, je však povinností Product Owenera zajistit všechny důležité informace jednotlivých User Stories nadcházejícího sprintu. Představení požadavků probíhá během ceremonie Sprint Planning, kde spolu vývojový tým a Product Owner diskutují požadavky jednotlivých User Stories, které mají být pracovní náplní nadcházejícího sprintu. Pro jednodušší pochopení co má být podle požadavků na konci sprintu dodáno, sestaví Product Owner a vývojový tým akceptační kritéria jednotlivých User Stories.

Konkrétní praktika

- Správa a definice Epiců/User Stories v Product Backlogu v systému JIRA
- Vysvětlení a pochopení požadavků na Level 2 Sprint Planning
- Sestavení akceptačních kritérií jednotlivých User Stories

Výstup

- Product Backlog v systému JIRA
- Popis a akceptační kritéria jednotlivých User Stories v systému JIRA
- Popis procesu v AG 5791

SP 1.2 Obtain Commitment to Requirements / Získej závazek k požadavkům

Získej závazek od účastníků projektu k produktovým požadavkům (11).

Další praktika, která je pokryta metodou SCRUM. V druhé části Sprint Planning vývojový tým vytváří Sprint Backlog, který představuje plánovanou práci nadcházejícího sprintu, ke které se vývojový tým zavazuje.

Konkrétní praktika

- Level 3 Sprint Planning

Výstup

- Sprint Backlog v systému JIRA
- Popis procesu v AG 5791

SP 1.3 Manage Requirements Changes / Spravujte změny požadavků

Spravujte změny požadavků, které se vyvíjejí během projektu (11).

Řešení

Změny požadavků v týmu FSS jsou řešeny procesem Change Management, změna musí být analyzována a potvrzena příslušnou autoritou. Všechny akceptované změny jsou Product Ownerem zaznamenány na úrovni Epiců či User Stories. Product Owner tyto změny požadavků diskutuje s týmem na Sprint Planning meetingu. Změny jsou promítány i do akceptačních kritérií jednotlivých User Stories.

Konkrétní praktika

- Change Management
- Správa a definice Epiců/User Stories v Product Backlogu v systému JIRA
- Vysvětlení a pochopení změn požadavků na Level 2 Sprint Planning

Výstup

- Product Backlog v systému JIRA
- Definice Epiců/User Stories v systému JIRA
- Popis procesu v AG 5791 s odkazem na příslušný proces

SP 1.4 Maintain Bidirectional Traceability of Requirements / Udržuj obousměrnou sledovatelnost požadavků

Udržuj obousměrnou sledovatelnost požadavků a výsledku práce (11).

Product Backlog má následující hierarchii – Epic > User Stories > úkoly. Za definici Epicu a jednotlivých User Stories, které vedou k jeho dokončení, je zodpovědný Product Owner. User Story je pak rozbita na jednotlivé úkoly nezbytné pro její dokončení. Tento úkon provádí vývojový tým. Důležité ceremonie pro tuto praktiku jsou Level 2 Backlog Grooming (představení Epicu), Level 2–3 Sprint Planning (představení User Stories, dekompozice na úkoly), Sprint Demo (představení výsledné práce a akceptace jednotlivých User Stories).

Konkrétní praktiky

- Správa Product Backlog
- Level 2 Backlog Grooming, Level 2-3 Sprint Planning, Sprint Demo

Výstup

- Product Backlog v systému JIRA
- Popis procesu v AG 5791

SP 1.5 Ensure Alignment Between Project Work and Requirements / Zajisti provázanost projektové práce a požadavků

Projektové plány a pracovní výsledky odpovídají požadavkům (11).

Product Owner přetváří požadavky projektu na pracovní bloky v podobě User Stories ve Sprint Backlogu. Je následně zodpovědný za identifikaci rozdílu mezi požadavky a skutečným výsledkem, na základě „Definition of Done“ a akceptačních kritérií jednotlivých User Stories. Tato činnost probíhá během Sprint Demo.

Konkrétní praktiky

- Přetváření požadavků do User Stories v Product Backlogu
- „Definition of Done“, akceptační kritéria User Stories
- Sprint Demo

Výstup

- User Stories v Product Backlogu v systému JIRA
- Popis „Definition of Done“ a akceptačních kritérií u User Stories v systému JIRA
- Popis procesu v AG 5791

4.1.2 Project Planning (PP)

Účelem procesní oblasti Project Planning je stanovení a udržování plánů, které definují projektové aktivity (11).

SG 1 Establish Estimates / Stanovte odhady náročnosti

Odhady náročnosti parametrů plánování projektu jsou stanoveny a udržovány (11).

SP 1.1 Estimate the Scope of the Project / Provďte odhad rozsahu projektu

Vytvořte WBS (Work Breakdown Structure) na nejvyšší úrovni pro stanovení odhadu rozsahu projektu (11).

Na programové úrovni je prvotní rozsah projektu vytvořen během fáze IPDS 3a (plánování). Je zde vytvořena WBS na nejvyšší úrovni spolu s přibližným časovým odhadem jejich položek.

Během vývojové fáze jsou tyto artefakty revidovány a aktualizovány během Level 1 Backlog Grooming a Level 1 Program Planning, kde jsou přítomni všichni stakeholderi a Product Owneri. Výstupem těchto ceremonií jsou přibližně časově ohodnocené Epicy v Product Backlogu. Které jsou pak následně dekomponovány na User Stories během Level 2 Backlog

Groomingu. Víceúrovňovou WBS představuje Product Backlog. Detailní estimace a stanovení nižší úrovně WBS pak probíhají na Level 3 Sprint Planningu.

Konkrétní praktika

- IPDS 3a fáze (plánování)
- Level 1 Backlog Grooming, Level 1 Program Planning
- „High Level“ odhad časové náročnosti Epicu
- Level 2 Backlog Grooming, Level 3 Sprint Planing
- Detailní odhad časové náročnosti User Story

Výstup

- „High Level“ WBS v Microsoft Projectt
- Product Backlog
- Odhad časové náročnosti Epicu/User Stories
- Popis procesu v AG 5791

SP 1.2 Establish Estimate of Work Product and Task Attributes / Stanovte estimace pracovního produktu a atributů úkolů

Stanovte estimace pracovního produktu a atributů úkolu (11).

„High Level“ WBS je stanovena během 3a IPDS fáze, kde jsou její jednotlivé části hrubě odhadnuty časovou náročností.

WBS je aktualizována a revidována pravidelně během Level 1 Backlog Grooming a Level 1 Program Planning.

Detailnější estimace jednotlivých položek WBS probíhá pravidelně na Level 2 Backlog Grooming a Level 2 Sprint Planning.

Konkrétní praktika

- IPDS 3a fáze (plánování)
- Level 1 Backlog Grooming, Level 1 Program Planning
- „High Level“ odhad časové náročnosti Epicu
- Level 2 Backlog Grooming, Level 3 Sprint Planing
- Detailní odhad časové náročnosti User Story

Výstup

- „High Level“ WBS v Microsoft Projectt
- Product Backlog
- Odhad časové náročnosti Epicu/User Stories
- Popis procesu v AG 5791

SP 1.3 Define Project Life Cycle Phases / Definujte fáze životního cyklu projektu

Projekt začíná částí přípravy, která obsahuje 4 prvotní fáze procesu IPDS, dále následuje část realizace, která je zastoupena metodou SCRUM. Zakončení projektu probíhá poslední fází z procesu IPDS - ukončení.

Konkrétní praktika

- Životní cyklus projektu vývoje mobilních aplikací
- SCRUM

Výstup

- Popis procesu v AG 5791

SP 1.4 Estimate Effort and Costs / Estimujte pracovní úsilí a náklady

Odhadněte pracovní úsilí a náklady práce a úkolů, založených na zdůvodněných estimacích (11).

V IPDS fázi 3a (plánování) jsou stanoveny „High Level“ nákladové a časové plány projektu. Tyto artefakty jsou revidovány a aktualizovány měsíčně Project Engineerem projektu a PP&C (Project and Planning Control) osobou přiřazenou na projekt a také na Level 1 Backlog Grooming a Level 1 Program Planning. Detailní ohodnocení pracovních úkolů probíhá na Level 3 Sprint Planning, kde vyvojový tým na základě svých vlastních zkušeností a metrik získaných z předchozích sprintů odhaduje časovou náročnost.

Konkrétní praktika

- Stanovení „High Level“ nákladových, časových plánů v IPDS 3a fázi
- Pravidelné přezkoumání a aktualizace těchto plánů (pravidelné meetingy PE-PP&C, Level 1 Backlog Grooming, Level 1 Program Planning)
- Level 3 Sprint Planning

Výstup

- Nákladový a časový plán (WBS v Microsoft Project)
- Detailní estimace User Stories ve Sprint Backlogu v systému JIRA
- Popis/odkaz procesu v AG 5791

SG 2 Develop a Project Plan / Vytvoř projektový plán

Projektový plán je stanoven a udržován jako základ pro řízení projektu (11).

SP 2.1 Establish the Budget and Schedule / Stanovte rozpočet a časový plán

Stanovte a udržujte rozpočet a časový plán projektu (11).

Rozpočet a časový plán je stanoven v IPDS fázi 3a (plánování), ten je v průběhu vývoje revidován a aktualizován na čtvrtletní bázi (Level 1 Backlog Grooming, Level 1 Sprint

Planning) nebo měsíční bázi (rozpočet, zodpovědnost PE a PP&C). Jako vstupy pro tyto činnosti slouží časové estimace User Stories v Product Backlogu. Závislosti jednotlivých úkolů a jejich komplexita je identifikována vývojovým týmem na Level 2 Backlog Groomingu nebo Level 2-3 Sprint Planningu.

Konkrétní praktika

- Stanovení „high level“ rozpočtu a časového plánu v IPDS 3a fázi
- Pravidelné přezkoumání a aktualizace těchto plánů (pravidelné meetingy PE-PP&C, Level 1 Backlog Grooming, Level 1 Program Planning)
- Level 2 Backlog Grooming, Level 2 Sprint Planning

Výstup

- Rozpočet a časový plán
- Definice závislostí a časové ohodnocení úkolů v systému JIRA
- Popis/odkaz procesu v AG 5791

SP 2.2 Identify Project Risks / Identifikujte projektové risky

Identifikujte a analyzujte projektové rizika (11).

Z pohledu SCRUMu jsou za rizika považovány překážky (impediments), které brání členům vývojového týmu pokračovat během sprintu ve své práci. Tyto překážky jsou identifikovány na Daily Scrum samotnými členy týmu. Překážky jsou zaznamenány a Scrum Master je zodpovědný za provedení korektivních akcí. Stejný princip je aplikován o úroveň výš, na Scrums of Scrum.

Z projektového pohledu jsou rizika řízeny dle procesu Risk Management AP 1485. Každý týden je svolán Risk Management meeting, kde jednotliví Product Ownéři, Technical Manager, Program Manager identifikují nebo aktualizují rizika projektu, jejich pravděpodobnost, dopad a mitigační akce. Rizika však mohou být identifikována a zanesena do Risk Logu na denní bázi kterýmkoliv členem týmu FSS.

Konkrétní praktika

- Identifikace a záznam překážek na Daily Scrum/Scrums of Scrum a jejich korektivní akce
- Proces Risk Management
- Týdenní meeting Risk Management

Výstup

- Seznam překážek identifikovaných na Daily Scrum/Scrums of Scrum a jejich stav
- Risk Log
- Popis/odkaz procesu v AG 5791

SP 2.3 Plan Data Management / Naplánujte řízení dat

Plán řízení projektových dat (11).

SCRUM jako takový nedefinuje jak zacházet a řídit data. Je potřeba vytvořit proces řízení dat. V systému JIRA jsou vedeny Epicy a User Stories, které obsahují popis požadavků a důležité informace. Na týmových Wiki stránkách jsou jednoduchou formou popsány důležité technické informace týkající se produktu. Projektový Sharepoint slouží k uchování důležitých projektových dokumentů, které například vycházející z požadavků procesu IPDS, nebo jsou získány od třetích stran participujících na projektu. Součástí je dokumentová struktura úložiště. Systém TestLink zase obsahuje testovací scénáře pro testování aplikace.

Konkrétní praktika

- Proces řízení dat

Výstup

- Systém JIRA, Sharepoint, Wiki, TestLink
- Proces popsán v AG 5791

SP 2.4 Plan the Project's Resources / Plánujte projektové zdroje

Plán zdrojů pro vykonávání projektu (11).

Prvotní plán je stanoven během 3a IPDS fáze. Tento plán se během agilního vývoje reviduje a aktualizuje na programové úrovni během Level 1 Backlog Grooming a Level 1 Sprint Planning. Další řízení zdrojů probíhá mezi SCRUM týmy během Program Synchronization. Při identifikaci chybějícího potřebného zdroje, inicializuje jeho získání Technical Manager spolu s Project Engineerem.

Konkrétní praktika

- 3a IPDS fáze (plánování)
- Projednání a aktualizace plánu zdrojů na Level 1 Backlog Grooming, Level 1 Sprint Planning, Program Synchronization

Výstup

- Plán zdrojů
- Popis/odkaz procesu v AG 5791

SP 2.5 Plan Needed Knowledge and Skills / Plánujte potřebné dovednosti a schopnosti

Plánujte dovednosti a schopnosti potřebné pro vykonávání projektu (11).

SCRUM se nezaobírá plánováním a rozvíjením potřebných dovedností a schopností. V týmu FSS je použit dokument „skill matrix“, který slouží pro identifikaci dovedností a jejich úrovně, které jsou obsaženy v jednotlivých týmech. Není zde však nastaven proces plánování dovedností podle požadavků projektu či plán rozvíjení těchto dovedností.

Řešením je stanovení procesu „řízení požadovaných dovedností“ („skill requirement management“).

Proces bude představovat následující kroky:

- Identifikace dovedností a schopností potřebných pro realizaci projektu
- Dokumentace těchto položek („skill requirement document“)
- Analýza stávajícího stavu (vstup „skill matrix“ dokument)
- Plán získání požadovaných dovedností (školení, trénink, nábor nového zaměstnance) „skill achievement plan“

Tato činnost bude probíhat pravidelně ve stanoveném intervalu za přítomnosti Product Ownerů, Project Engineera a Technical Managera.

Konkrétní praktika

- Proces „skill requirement management“

Výstup

- „skill requirement document, „skill matrix“, „skill achievement plan“
- Popis procesu v AG 5791

SP 2.6 Plan Stakeholder Involvement / Plánujte zapojení stakeholderů

Plánujte zapojení identifikovaných stakeholderů (11).

Agilní přístup je založen na filosofii aktivního zapojení zainteresovaných stran do vývoje produktu. Interní stakeholderi jsou pravidelně informováni o stavu projektu během Level 1 Backlog Grooming a Level 1 Sprint Planning. O výsledku jsou informováni během Program Sprint Demo. Externí zainteresované strany jsou informovány na bázi pravidelných informačních schůzek. Product Owner je zodpovědný za vedení zapojení zainteresovaných stran.

Konkrétní praktika

- Level 1 Backlog Grooming, Level 1 Sprint Planning
- Program Sprint Demo
- Pravidelné schůzky s externími stakeholdery

Výstup

- Popis procesu v AG 5791

SP 2.7 Establish the Project Plan / Stanovte projektový plán

Stanovte a udržujte ucelený projektový plán (11).

Prvotní plány jsou definovány v IPDS fázi 3a (plánování), tyto plány jsou pravidelně revidovány a aktualizovány. Plán práce představuje Product Backlog, který se projednává na Level 2 Backlog Grooming a Level 2 Sprint Planning.

Konkrétní praktika

- Stanovení „high level“ plánu v IPDS 3a fázi
- Pravidelné přezkoumání a aktualizace těchto plánů (Level 1 Backlog Grooming, Level 1 Program Planning)
- Level 2 Backlog Grooming, Level 2 Sprint Planning

Výstup

- Statement of Work
- Product Backlog
- Popis procesu v AG 5791

SG 3 Obtain Commitment to the Plan / Získejte závazek k plánu

Závazky k projektovému plánu jsou stanoveny a udržovány (11).

SP 3.1 Review Plans That Affect the Project / Zrevidujte plány týkající se projektu

Reviduj veškeré plány, které se týkají projektu pro pochopení závazků (11).

Na projektové úrovni jsou plány revidovány a aktualizovány na Level 1 Backlog Grooming a Level 1 Sprint Planning. Na týmové úrovni je plán projednáván na Level 2 Backlog Grooming a Level 2 Sprint Planning.

Konkrétní praktika

- Pravidelné přezkoumání a aktualizace plánů (Level 1 Backlog Grooming, Level 1 Program Planning)
- Aktualizace plánu na týmové úrovni (Level 2 Backlog Grooming, Level 2 Sprint Planning)

Výstup

- Statement of Work
- Product Backlog
- Popis procesu v AG 5791

SP 3.2 Reconcile Work and Resource Levels / Srovnejte práci a úroveň zdrojů

Upravte projektový plán podle dostupných a odhadnutých potřebných zdrojů (11).

SCRUM tuto praktiku pokrývá ceremonií Sprint Planning. Během Level 2-3 Sprint Planning tým srovnává náročnost požadavků se svojí kapacitou, tedy kolik práce dokážou dodat. Mezi jednotlivými SCRUM týmy dochází k synchronizaci zdrojů na Program Synchronization.

Konkrétní praktika

- Level 2-3 Sprint Planning
- Program Synchronization

Výstup

- Sprint Backlog
- Proces popsán v AG 5791

SP 3.3 Obtain Plan Commitment / Získejte závazek k plánu

Získejte závazek relevantních stakeholderů, kteří se podílejí nebo podporují exekuci projektu (11).

Opět SCRUM tuto praktiku částečně pokrývá během Level 3 Sprint Planning, kde se vývojový tým zavazuje ke splnění úkolů v následujícím sprintu. Výsledkem závazku je vytvořený Sprint Backlog.

Na programové úrovni představuje závazek stakeholderů akceptované ukončení fáze IPDS 3a (plánování), kde jsou všechny plánované dokumenty zrevidovány relevantními stakeholdery a plán je realizován fází agilního vývoje.

Konkrétní praktika

- Level 3 Sprint Planning
- Akceptované ukončení fáze IPDS 3a

Výstup

- Sprint Backlog
- Schválený PECA sheet
- Proces popsán v AG 5791

4.1.3 Project Monitoring and Control (PMC)

Záměrem této procesní oblasti je poskytnout porozumění o stavu projektu a identifikovat tak případné odchýlení od jeho plánu (11).

SG 1 Monitor the Project Against the Plan / Monitorujte stav projektu vůči plánu

Aktuální vývoj projektu je monitorován vůči stanovenému plánu (11).

SP 1.1 Monitor Project Planning Parameters / Monitorujte parametry projektového plánování

Monitorujte skutečné hodnoty parametrů projektu vůči projektovému plánu (11).

Ve SCRUMu je stav plánu reportován během sprintu na Daily Scrum. Graficky je stav reprezentován pomocí „Burndown chart“, tento graf ukazuje, kolik práce z naplánovaného sprintu bylo doručeno a kolik zbývá. Díky tomu můžeme hodnotit, zda bude cíle sprintu dosaženo či ne.

Na úrovni Scrums of Scrum (denní meeting za účasti Product Ownerů, Scrum Masterů, Technical Managera, PE, PM) se diskutuje aktuální stav týmů a aktuální události relevantní k projektu. Podle charakteru událostí se vykonávají patřičné akce.

Za sledování finanční stránky projektu (náklady, rozpočet, atd.) je zodpovědný Project Engineer. Ten za pomoci PP&C sleduje aktuální stav těchto parametrů vůči plánu a provádí korektivní akce, pakliže jsou zapotřebí.

Konkrétní praktika

- Daily Scrum
- Scrums of Scrum
- Pravidelná revize finančních parametrů

Výstup

- „Burndown chart“
- Aktualizace projektových plánů (rozpočet, nákladová plán, atd.)
- Popis procesu v AG 5791

SP 1.2 Monitor Commitments / Monitorujte závazky

Sledujte závazky k projektovému plánu (11).

Stanovení závazku vznikne při naplánování Sprint Backlogu během Level 3 Sprint Backlog.

Plnění tohoto závazku je sledováno během sprintu na Daily Scrum.

Konkrétní praktika

- Level 3 Sprint Planning
- Daily Scrum
- Sledování „Burndown chart“

Výstup

- Sprint Backlog
- „Burndown chart“
- Popis procesu v AG 5791

SP 1.3 Monitor Project Risks / Monitorujte rizika projektu

Monitoruj rizika uvedené v projektovém plánu (11).

Na SCRUM úrovni jsou rizika identifikována jako překážky bránící členům týmu vykonávat plánovanou práci. Tyto překážky jsou zaznamenány do systému JIRA („Impediment log“) a je zodpovědností Scrum Mastera tyto překážky odstranit a sledovat jejich stav. Stejný princip je aplikován o úroveň výš, na Scrums of Scrum.

Na úrovni programu je každý týden pořádán Risk Management meeting, kterého se účastní Product Owneři jednotlivých SCRUM týmů, PE, Technical Manager a PO. Záměrem tohoto meetingu je revize stávajících rizik, identifikace nových a jejich mitigace. Rizika mohou být však identifikována a vložena do registru rizik na denní bázi.

Konkrétní praktika

- Identifikace a záznam překážek na Daily Scrum/Scrums of Scrum a jejich korektivní akce
- Proces Risk Management
- Týdenní meeting Risk Management

Výstup

- Seznam překážek identifikovaných na Daily Scrum/Scrums of Scrum a jejich stav
- Risk Log
- Popis/odkaz procesu v AG 5791

SP 1.4 Monitor Data Management / Monitorujte řízení dat

Sledujte řízení projektových dat vůči projektovému plánu (11).

SCRUM ani IPDS tuto praktiku nepokrývá. Řešením je stanovení procesu Data Management, jehož součástí bude pravidelná schůzka PE, TM (Technical Manager) a PO, kde náplní této schůzky bude zajištění dostupnosti dat, aktuální stav potřebných dat vyžadovaných procesem, identifikace zlepšení procesu Data Management.

Konkrétní praktika

- Pravidelná schůzka na téma řízení dat

Výstup

- Popis/aktualizace procesu řízení dat v AG 5791
- Výstup schůzky v podobě dokumentu uloženém na Sharepointu

SP 1.5 Monitor Stakeholder Involvement / Monitorujte zapojení stakeholderů

Sledujte zapojení zainteresovaných stran vůči projektovému plánu (11).

Zodpovědnost za řízení, sledování zapojení interních či externích stakeholderů do projektu přebírá Product Owner. Pro jasnou definici rolí zainteresovaných stran a jejich pravomocí vytvoří a průběžně aktualizuje dokument matice zainteresovaných stran.

Konkrétní praktika

- Monitorování zapojení zainteresovaných stran Product Ownerem

Výstup

- Dokument matice zainteresovaných stran
- Popis procesu v AG 5791

SP 1.6 Conduct Progress Reviews / Provádějte posouzení vývoje projektu

Pravidelně provádějte revizi vývoje projektu (11).

Na úrovni SCRUM týmu se posouzení stavu projektu provádí během Daily Scrum, kde vývojový tým reportuje aktuální stav běžícího sprintu. Překážky v plnění plánu jsou identifikovány a zaznamenány na „Impediments board“. Aktuální stav sprintu reflektuje Burndown chart.

Na programové úrovni je stav projektu měsíčně reportován důležitým stakeholderům. Za tento reporting je zodpovědný Product Owner.

Konkrétní praktika

- Daily Scrum
- Burndown chart
- Měsíční reporting stakeholderům

Výstup

- Sprint Backlog v systému JIRA
- „Impediments board“ v systému JIRA
- Burndown chart v systému JIRA
- Popis procesu v AG 5791

SP 1.7 Conduct Milestone Reviews / Provádějte posouzení milníků

Zhodnot'te dosažené úspěchy a výsledky projektu ve stanoveném milníku (11).

Posouzení splnění požadovaných výsledků probíhá během Sprint Review. Vývojový tým představí výsledky své práce, které jsou následně akceptovány Product Ownerem po posouzení splnění požadavků.

Milníky v podobě data vydání aplikace (Application release) jsou posuzovány dle daného procesu. Splnění požadavků na aplikaci je posouzeno stakeholdery. Teprve po jejich akceptaci je aplikace připravena na vydání. Posouzení splnění milníku je dokumentováno písemně.

Konkrétní praktika

- Sprint Review
- Application release proces

Výstup

- Akceptované User Stories v systému JIRA
- Zdokumentované posouzení vydávané aplikace stakeholdery
- Popis procesu v AG 5791

SG 2 Manage Corrective Action to Closure / Spravujte nápravné akce k uzavření

Nápravné akce jsou řízeny, pokud dojde k ukončení projektu z důvodů výrazné odchylky od plánu (11).

SP 2.1 Analyze Issues / Analyzujte vzniklé problémy

Shromážděte a analyzujte vzniklé problémy a určete nápravné akce pro jejich řešení (11).

Problémy během sprintu jsou reportovány vývojovým týmem na Daily Scrum. Jsou stanoveny nápravné akce, za které je zodpovědný určený člen týmu nebo Scrum Master. Scrum Master je zodpovědný za dokumentaci tohoto problému do systému JIRA a současně za sledování stavu stanovené akce.

Stejný princip je aplikován o úroveň výš. Denně na Scrums of Scrum jsou reportovány konkrétní aktuální problémy napříč projekty, následně proběhne identifikace nápravného řešení a jeho řešení. Akce je zaznamenána v systému JIRA.

Dlouhodobější problémy jsou identifikovány během Sprint Retrospective. Tým identifikuje problémy, které naskytly během sprintu a následně navrhne jeho řešení a řešitele. Realizace řešení je přiřazena stanovenému řešiteli jako úkol v následujícím sprintu.

Konkrétní praktika

- Daily Scrum, Scrums of Scrum
- Sprint Retrospective

Výstup

- „Impediment board“ v systému JIRA
- Zápis Sprint Retrospective
- Úkol v systému JIRA plánován do následujícího sprintu
- Popis procesu v AG 5791

SP 2.2 Take Corrective Action / Učíte nápravnou akci

Realizujte nápravné akce (11).

Nápravné akce identifikovány během sprintu jsou realizovány okamžitě a sledovány zodpovědnou osobou pomocí systému JIRA. Nápravné akce identifikovány na Sprint Retrospective jsou realizovány v následujícím sprintu a sledovány během Daily Scrum.

Konkrétní praktika

- Realizace nápravné akce během sprintu
- Realizace nápravné akce jako pracovní úkol v následujícím sprintu

Výstup

- „Impediment board“
- Úkol v systému JIRA plánován do následujícího sprintu
- Popis procesu v AG 5791

SP 2.3 Manage Corrective Actions / Spravujte nápravné akce

Spravujte nápravné činnosti k ukončení (11).

Za správu nápravných akcí je zodpovědná určená osoba. Nápravná akce stanovená na Sprint Retrospective je spravována jako klasický úkol v Product Backlogu. Zda nápravná akce splnila svůj účel je hodnoceno na další Sprint Retrospective.

Konkrétní praktika

- Správa nápravných akcí během sprintu nebo mezi sprinty

Výstup

- Úkol v Product Backlogu v systému JIRA
- „Impediment board“
- Popis procesu v AG 5791

4.1.4 Integrated Project Management (IPM)

Záměrem této procesní oblasti je stanovení a řízení projektu dle integrovaného a definovaného procesu, který vychází z procesů organizace (11).

SG 1 Use the Project's Defined Process / Využijte definovaný proces projektu

Projekt je realizován v souladu s procesy, které vycházejí ze standardních podnikových procesů (11).

SP 1.1 Establish the Project's Defined Process / Stanovte definovaný proces projektu

Stanovte a udržujte definovaný proces projektu po celou dobu jeho života (11).

Pro popis životního cyklu projektu, jeho fází, nezbytných artefaktů a podpůrných procesů slouží dokument AG 5791. Realizací všech navrhovaných řešení v této práci se tento dokument stane uceleným popisem procesu vývoje mobilních aplikací nejen pro vývojový tým FSS, ale i týmy napříč celou organizací Honeywell, které následují agilní přístup k vývoji a jsou vyňaty z procesu IPDS.

Konkrétní praktika

- Proces vývoje mobilních aplikací

Výstup

- AG 5791

SP 1.2 Use Organizational Process Assets for Planning Project Activities / Využijte aktiva organizace pro plánování projektových aktivit

Využijte znalostní bázi metrik organizace pro estimaci a plánování projektových aktivit (11).

Tuto praktiku pokrývá proces plánování v IPDS fázi 3a (plánování). Výchozími dokumenty jsou SOW (Statement of Work), Project Charter.

Konkrétní praktika

- Proces plánování v 3a fázi IPDS

Výstup

- SOW, Project Charter
- Popis/odkaz procesu v AG 5791

SP 1.3 Establish the Project's Work Environment / Stanovte projektové pracovní prostředí

Stanovte a udržujte projektové pracovní prostředí na základě standardů organizace (11).

Za pořízení pracovních nástrojů nezbytných pro vykonávání práce popsané v pracovní pozici zaměstnance je zodpovědný liniový nadřízený pracovníka. Údržba je poskytována IT oddělením organizace.

Pracovní nástroje určené požadavky projektu jsou definovány při plánování projektu a během jeho vývoje. Za pořízení těchto nástrojů je zodpovědný Project Engineer. Je stanovena odpovědná osoba za správu a údržbu těchto nástrojů.

Konkrétní praktika

- Definice požadovaných nástrojů pro uspokojení požadavků projektu
- Pořízení těchto nástrojů

Výstup

- Popis procesu v AG 5791

SP 1.4 Integrate Plans / Integrujte jednotlivé plány

Integrujte projektové a ostatní plány, které jsou relevantní pro projekt (11).

Integrovaný plán projektu představuje dokument SOW (Statement of Work) finalizován ve fázi 3a IPDS. Plán je pravidelně revidován vůči aktuálnímu stavu

Konkrétní praktika

- IPDS 3a fáze
- Pravidelná revize plánu

Výstup

- Project Management Plan
- SOW (Statement of Work)
- Proces/odkaz v AG 5791

SP 1.5 Manage the Project Using Integrated Plans / Použijte integrované plány pro řízení projektu

Pro řízení projektu využijte definovaného procesu, projektového plánu a ostatních plánu, které tento projekt ovlivňují (11).

Proces popsán v dokumentu AG 5791 jasně definuje, které artefakty respektive plány jsou nezbytné pro řízení a realizaci projektu vývoje mobilní aplikace. Na programové úrovni tyto plány představuje Project Charter a SOW (Statement of Work) finalizovány v 3a fázi IPDS.

Na vývojové úrovni tento plán představuje Product Backlog, konkrétněji Sprint Backlog.

Konkrétní praktika

- 3a fáze IPDS (plánování)
- Vývoj dle metodiky SCRUM

Výstup

- Project charter, SOW
- Product Backlog, Sprint Backlog
- Popis procesu v AG 5791

SP 1.6 Establish Teams / Stanovte tým

Stanovte a udržujte tým (11).

Sestavení týmu vychází z požadavků projektu na schopnosti a dovednosti. Tým je sestaven z pracovníků dle plánu zdrojů. SCRUM jasně definuje role v projektu. Struktura týmů je zdokumentována.

Konkrétní praktika

- Sestavení týmu na základě požadavků projektu
- SCRUM role

Výstup

- Zdokumentována struktura týmu v Management operating sztemu
- Popis procesu v AG 5791

SP 1.7 Contribute to Organizational Process Assets / Přispívejte do procesních aktiv společnosti

Sdílejte procesní zkušenosti napříč celou organizací (11).

Popsaný proces vývoje mobilních aplikací představuje dokument AG 5791. Proces může být poupraven při nalezení míst pro zlepšení. Tento dokument je k dispozici napříč celou organizací a slouží ostatním týmům, které se vymezují v použití klasického procesu IPDS.

Všechny důležité skutečné metriky projektu jsou uloženy na Sharepointu projektu.

Konkrétní praktika

- Proces AG 5791
- Sběr projektových dat

Výstup

- Projektová data uložena na Sharepointu
- Popis procesu v AG 5791

SG 2 Coordinate and Collaborate with Relevant Stakeholders / Koordinujte a spolupracujte s relevantními stakeholdery

Provádějte koordinace a kolaboraci mezi projektem a jeho relevantními stakeholdery (11).

SP 2.1 Manage Stakeholder Involvement / Spravujte zapojení stakeholderu

Spravujte zapojení stakeholderů relevantních k projektu (11).

Za řízení relevantních stakeholderů je ve SCRUMu zodpovědný Product Owner. Ten reflektuje požadavky zainteresovaných stran a přenáší je do Product Backlogu, který následně prezentuje vývojovému týmu.

Konkrétní praktika

- Řízení stakeholderů

Výstup

- Podněty stakeholderů integrovány v Product Backlogu v systému JIRA
- Popis procesu v AG 5791

SP 2.2 Manage Dependencies / Spravujte závislosti

Spolupracujte se stakeholdery na identifikaci projektových závislostí (11).

Tuto praktiku lze pokrýt stejně jako předchozí. Závislosti mezi projekty a zainteresovanými stranami spravuje Product Owner.

Konkrétní praktika

- Řízení stakeholderů

Výstup

- Popis procesu v AG 5791

SP 2.3 Resolve Coordination Issues / Vyřešte koordinační problémy

Vyřešte problémy s relevantními stakeholdery (11).

Další z praktik, kterou pokrývá řízení stakeholderů, které má na starosti Product Owner.

Konkrétní praktika

- Řízení stakeholderů

Výstup

- Popis procesu v AG 5791

4.1.5 Risk Management (RSKM)

Účelem řízení rizik je identifikování problému před jeho vznikem a identifikování možností jak tomuto problému předejít (11).

SG 1 Prepare for Risk Management / Připravte proces řízení rizik

Proveďte přípravu procesu řízení rizik (11).

SP 1.1 Determine Risk Sources and Categories / Určete zdroje a kategorie rizik

Určete zdroje a kategorie rizik (11).

Rizika jsou rozdělena podle identifikace zdroje na interní a externí. Následně jsou rozdělena dle kategorií.

Konkrétní praktika

- Rozdělení rizik dle jejich zdroje a kategorií

Výstup

- Risklog
- Popis/odkaz procesu v AG 5791

SP 1.2 Define Risk Parameters / Stanovte parametry rizik

Stanovte parametry rizik, které slouží pro jejich analýzu (11).

Stanovené parametry pro hodnocení rizik jsou následující:

- Pravděpodobnost výskytu rizika (1-5, kde 1 představuje nejmenší pravděpodobnost)
- Dopad rizika na projekt (1-5, kde 1 představuje minimální dopad)
- Výsledná hodnota rizika (počítána jako pravděpodobnost * dopad)

Podle výsledné hodnoty jsou rizika prioritizována.

Konkrétní praktika

- Risk Management

Výstup

- Risk Log
- Odkaz na proces v AG 5791

SP 1.3 Establish a Risk Management Strategy / Stanovte strategii řízení rizik

Stanovte a udržujte strategii řízení rizik (11).

Strategie popsána v Risk Management dokumentu a odkaz na něj uveden v AG 5791.

Strategie obsahuje následující:

- Pravidelnost sledování rizik
- Nástroje použity pro identifikaci, analýzu, ohodnocení rizik
- Parametry rizik (pravděpodobnost, dopad, hodnota)
- Zdroje a kategorie rizik
- Způsoby redukce rizik

Konkrétní praktika

- Risk Management

Výstup

- Risk Management dokument
- Odkaz na Risk Management dokument v AG 5791

SG 2 Identify and Analyze Risks / Identifikujte a analyzujte rizika

Rizika projektu jsou identifikovány, analyzovány a je určena jejich důležitost (11).

SP 2.1 Identify Risks / Identifikujte rizika

Identifikujte rizika a zdokumentujte je (11).

Rizika na úrovni SCRUM týmu jsou identifikovány během Daily Scrum samotným týmem. Na úrovni managementu jsou rizika identifikována během Scrums of Scrum nebo na pravidelné schůzce Risk Managementu. Všechny rizika jsou zařazena do příslušné kategorie, ohodnocena a zdokumentována.

Rizika jsou popsána formou: „KDYŽ“ okolnost „PAK“ událost „VEDE K,, dopadu.

Konkrétní praktika

- Identifikace rizik na Daily Scrum, Scrums of Scrum, schůzce Risk Management
- Kategorizace, ohodnocení a dokumentace rizik
- Risk Management

Výstup

- Risklog
- Popis/odkaz procesu v AG 5791

SP 2.2 Evaluate, Categorize, and Prioritize Risks / Zhodnoťte, kategorizujte a prioritizujte rizika

Zhodnoťte a kategorizujte každé identifikované riziko a stanovte jeho prioritu (11).

Rizika jsou ohodnocena při jejich identifikaci. Riziko je kategorizováno dle kategorií stanovených v SP 1.1 Determine Risk Sources and Categories. Priorita rizik se stanovuje dle hodnoty vypočítané dle parametrů rizika popsanych v SP 1.2 Define Risk Parameters.

- Identifikace rizik na Daily Scrum, Scrums of Scrum, schůzce Risk Management
- Kategorizace, ohodnocení a dokumentace rizik
- Risk Management

Výstup

- Risklog
- Popis/odkaz procesu v AG 5791

SG 3 Mitigate Risks / Redukujte rizika

Dopad rizika je redukován (11).

SP 3.1 Develop Risk Mitigation Plans / Vytvořte plány redukce rizik

Vytvořte plán redukce rizik v souladu se strategií řízení rizik (11).

Plán pro redukci rizika je stanoven během Risk Management meetingu, nebo okamžitě při identifikaci kritického rizika. Je stanoven vlastník tohoto plánu, tato osoba je zodpovědná za realizaci plánu redukce rizika. Stav plánu je revidován pravidelně každý týden na Risk Management meetingu.

Konkrétní praktika

- Vytvoření plánu redukce rizika, určení zodpovědné osoby
- Risk Management

Výstup

- Risklog
- Popis/odkaz procesu v AG 5791

SP 3.2 Implement Risk Mitigation Plans / Realizujte plány redukce rizik

Pravidelně monitorujte stav identifikovaného rizika a realizujte plán jeho redukce, pokud je to vyžadováno (11).

Stav realizace plánu redukce rizika je revidován na pravidelných týdenních schůzkách Risk Management. Parametry rizika jsou aktualizovány podle stavu plánu jeho redukce. Pokud výsledky redukce rizika neodpovídají plánu, je tento plán modifikován.

Konkrétní praktika

- Revize stavu plánu redukce rizika
- Risk Management

Výstup

- Risklog
- Popis/odkaz procesu v AG 5791

4.2 Výsledný model

V této části se nachází tabulka reprezentující výsledný hybridní model. Obsahuje vybrané procesní oblasti, jejich cíle a specifické praktiky. U specifických praktik je uveden způsob navrženého pokrytí této praktiky metodou SCRUM, procesem IPDS nebo jejich společným působením. Navržené řešení může praktiku pokrývat zcela (P – pokrývá), částečně (ČP – částečně pokrývá) nebo vůbec (N – nepokrývá). Sloupec „Dokumentace v AG 5791“ indikuje stav pokrytí dokumentace navrženého řešení.

Procesní oblast	Cíl	Specifická praktika	SCRUM	IPDS	Dokumentace v AG 5791
REQM					
	SG 1				
		SP 1.1	P	-	P
		SP 1.2	P	-	P
		SP 1.3	ČP	P	ČP
		SP 1.4	P	-	ČP
		SP 1.5	P	-	ČP
PP					
	SG 1				
		SP 1.1	ČP	ČP	N
		SP 1.2	ČP	ČP	N
		SP 1.3	ČP	ČP	N
		SP 1.4	ČP	ČP	N
	SG 2				
		SP 2.1	ČP	ČP	N
		SP 2.2	ČP	P	N
		SP 2.3	N	N	N
		SP 2.4	N	p	N

Procesní oblast	Cíl	Specifická praktika	SCRUM	IPDS	Dokumentace v AG 5791
		SP 2.5	N	N	N
		SP 2.6	P	-	ČP
		SP 2.7	ČP	ČP	ČP
	SG 3				
		SP 3.1	ČP	ČP	ČP
		SP 3.2	P	-	N
		SP 3.3	ČP	ČP	N
PMC					
	SG1				
		SP 1.1	ČP	ČP	N
		SP 1.2	P	-	N
		SP 1.3	ČP	ČP	N
		SP 1.4	N	N	N
		SP 1.5	P	-	N
		SP 1.6	P	-	N
		SP 1.7	P	-	N
	SG 2				
		SP 2.1	P	-	N
		SP 2.2	P	-	N
		SP 2.3	P	-	N
IPM	SG 1				
		SP 1.1	N	N	N
		SP 1.2	N	P	N
		SP 1.3	N	N	N
		SP 1.4	N	P	N
		SP 1.5	N	P	N
		SP 1.6	N	N	N
		SP 1.7	N	N	N
	SG 2				
		SP 2.1	P	-	N
		SP 2.2	P	-	N
		SP 2.3	P	-	N
RSKM					
	SG 1				
		SP 1.1	N	N	N
		SP 1.2	N	P	N
		SP 1.3	N	P	N
	SG 2				

Procesní oblast	Cíl	Specifická praktika	SCRUM	IPDS	Dokumentace v AG 5791
		SP 2.1	N	P	N
		SP 2.2	N	P	N
	SG 3				
		SP 3.1	N	P	N
		SP 3.2	N	P	N

Tabulka č. 13: Hybridní model CMMI, Zdroj: Vlastní zpracování

Pro pochopení tabulky, která představuje výsledný hybridní model, budeme ilustrovat modelové kombinace s jejich základním vysvětlením v tabulce č. 14.

Varianta	SCRUM	IPDS	Dokumentace v AG 5791	Vysvětlení
1	P	-	P	Praktika je zcela pokryta agilní metodou SCRUM. Je zdokumentována.
2	P	-	ČP	Praktika je zcela pokryta agilní metodou SCRUM. Je částečně zdokumentována.
3	P	-	N	Praktika je zcela pokryta agilní metodou SCRUM. Není zdokumentována.
4	ČP	P	ČP	Praktika je částečně pokryta metodou SCRUM a plně procesem IPDS. Navržené řešení je kombinace těchto dvou přístupů. Je částečně zdokumentována.
5	ČP	P	N	Praktika je částečně pokryta metodou SCRUM a plně procesem IPDS. Navržené řešení je kombinace těchto dvou přístupů. Není zdokumentována.
6	ČP	ČP	ČP	Praktika je částečně pokryta metodou SCRUM a procesem IPDS. Vzájemným spojením těchto dvou přístupů je praktika pokryta. Je částečně zdokumentována.
7	ČP	ČP	N	Praktika je částečně pokryta metodou SCRUM a procesem IPDS. Vzájemným spojením těchto dvou přístupů je praktika pokryta. Není zdokumentována.
8	N	P	N	Praktika není pokryta metodou SCRUM. Je zcela pokryta procesem IPDS. Není zdokumentována.
9	N	N	N	Praktika není pokryta metodou SCRUM či procesem IPDS. Návrh procesu je popsán v dané kapitole.

Tabulka č. 14: Vysvětlení modelových kombinací, Zdroj: Vlastní zpracování

4.3 Způsob realizace řešení



Obrázek č. 19: Činnosti nutné k realizaci navrženého řešení, Zdroj: Vlastní zpracování

Pro realizaci navrhovaného řešení je potřeba provést následující činnosti:

Analýza aktuálního stavu

Vzhledem k časovému odstupu a velké variabilitě prostředí, kde se situace a procesy v týmu FSS stále mění, je potřeba provést jednoduchou analýzu aktuálního stavu, která má za cíl zjistit odchylky poznatků uvedených v této práci od skutečné situace.

Definice procesu do AG 5791

Tato činnost zahrnuje definici navrženého procesu do dokumentu, který slouží k popisu procesu vývoje mobilních aplikací s využitím agilního přístupu. Definice bude ve formě odkazu na již popsany proces, či zcela nový popis navrženého procesu.

Validace procesu

Definovaný proces je potřeba validovat s relevantními stakeholdery. Popsaná definice musí projít schválením.

Nasazení procesu

Činnost zahrnuje zavedení definovaného procesu do reálného užití.

Continuous improvement

Tato činnost je mimo rámec této diplomové práce, je to nikdy nekončící proces, který má za cíl neustálé zlepšování nastavených procesů.

Odhad náročnosti činností u jednotlivých variant

Tabulka č. 15 ukazuje odhad náročností v jednotkách mh - manhour (člověkohodina) činností u jednotlivých variant z tabulky č. 14. Každá varianta má v jednotlivých činnostech tři druhy odhadu náročnosti - optimistický (O), tedy činnost bude probíhat v souladu s plánem bez problémů, pesimistický (P), kdy budou odhaleny určité potíže a počítá se s vyšší náročností a průměr (M), počítán z dvou předchozích odhadů. Poslední sloupec představuje součet průměrů odhadů u jednotlivých činnostech.

Varianta	Analýza			Definice			Validace			Nasazení			Celkem (mh)
	O (mh)	P (mh)	M (mh)	O (mh)	P (mh)	M (mh)	O (mh)	P (mh)	M (mh)	O (mh)	P (mh)	M (mh)	
1	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	4	2.5	1	3	2	1	1	1	0	1	0.5	6
3	1	4	2.5	1	5	3	1	1	1	0	1	0.5	7
4	1	6	3.5	1	3	2	1	3	2	0	4	2	9.5
5	1	4	2.5	1	6	3.5	1	3	2	0	4	2	10
6	1	6	3.5	1	5	3	1	3	2	0	6	3	11.5
7	1	6	3.5	1	8	4.5	1	5	3	0	6	3	14
8	1	2	1.5	1	2	1.5	1	1	1	0	4	2	6
9	1	12	6.5	1	12	6.5	1	10	5.5	1	16	8.5	27

Tabulka č. 15: Odhad náročnosti činností u jednotlivých variant, Zdroj: Vlastní zpracování

4.4 Ekonomické zhodnocení

Ekonomické zhodnocení navrhovaného řešení představuje kalkulaci nákladů, které je potřeba vynaložit na jeho implementaci. Následující tabulka č. 16 představuje kalkulaci odhadu náročnosti realizace celého navrhovaného řešení. Každá varianta je vyčíslena počtem jejího výskytu v hybridním modelu (četnost) a násobena odhadem náročnosti z tabulky č. 15 pro danou variantu. Výsledkem je odhadovaná náročnost realizace celého navrhovaného řešení v mh - manhours (člověkohodiny) a md - mandays (člověkodny).

Varianta	Četnost	Náročnost (mh)	Celkem (mh)
1	2	0	0
2	3	6	18
3	11	7	77
4	1	9.5	9.5
5	1	10	10
6	2	11.5	23
7	8	14	112
8	10	6	60
9	8	27	216
		Celkem (mh)	526
		Celkem (md)	66

Tabulka č. 16: Odhad náročnosti realizace vlastního řešení, Zdroj: Vlastní zpracování

Cena za člověkohodinu je stanovena na 1 000 Kč (18).

Celkové náklady realizace navrhovaného řešení jsou tedy odhadem stanoveny na **526 000 Kč**.

4.5 Přínos vlastního řešení

Realizace navrhovaného řešení přináší přínos zejména ve třech oblastech. První z nich je interní audit splnění požadavků CMMI. Po realizaci navrženého řešení je schopen tým FSS zajistit splnění požadavků CMMI v procesních oblastech Requirements Management, Project Planning, Project Monitor and Control, Integrated Project Management a Risk Management. Díky této implementaci bude tým FSS splňovat požadavky, které jsou striktně vyžadovány specifickými a významnými zákazníky pro získání zakázek. Tento přínos je v případě získání těchto zakázek více než zásadní.

Druhou oblastí je samotné nasazení a úspěšná aplikace této výšeče CMMI. CMMI při svém užití zvyšuje kvalitu a produktivitu týmu. Díky aplikací metod z tohoto procesního rámce se bude zvyšovat produktivita týmu a zároveň zvyšovat kvalita výsledného produktu.

Třetí oblast je využití navrženého řešení napříč celou organizací. Pro nově vznikající týmy ve společnosti Honeywell, které se vyjímají z klasického procesu IPDS a budou pro vývoj

softwaru následovat agilní přístup metodou SCRUM, může výsledný model této práce sloužit jako podklad pro stanovení činností, procesů a praktik, které zaručí splnění jednotlivých cílů CMMI v procesních oblastech Requirements Management, Project Planning, Project Monitor and Control, Integrated Project Management a Risk Management.

5 ZÁVĚR

V současné době se organizace snaží neustále zvyšovat kvalitu svých produktů či služeb. Existuje mnoho rámců a metodik, které slouží k dosažení tohoto cíle. Cílem agilní metody SCRUM, je zapojení zákazníka do samotného vývoje produktu, který probíhá v krátkých iteracích a tím tak dodat kvalitní produkt, který bude splňovat požadavky zákazníka. Model CMMI slouží ke zvýšení kvality produktu, zdokonalením procesu jeho vývoje. Tyto dva rámce spolu mohou vzájemně spolupracovat. Spolupráce spočívá v tom, že model CMMI poskytuje návod organizaci, co je potřeba praktikovat a agilní metoda říká, jakým způsobem. Samozřejmě metoda SCRUM není schopna pokrýt všechny praktiky modelu CMMI, a proto je nutno ji doplnit klasickými metodami projektového managementu.

Cílem práce byla analýza současného stavu použití agilní metodologie SCRUM a projektového řízení v rámci modelu kvality CMMI a na základě této analýzy navržení implementačního modelu hybridního fungování v softwarovém týmu. Práce se zaměřuje na vybrané procesní oblasti modelu CMMI z kategorie Project Management. Konkrétně na Requirements Management, Project Planning, Project Monitor and Control, Integrated Project Management a Risk Management

Práce naplnila svůj cíl a zanalyzovala současný stav použití agilní metodologie SCRUM a projektového řízení v rámci modelu kvality CMMI. Zkoumaný subjekt představoval vývojový tým ve společnosti Honeywell, který se zaměřuje na vývoj mobilních aplikací v leteckém odvětví. Tento tým se vymezuje z klasického procesu vývoje produktu, který je nastaven napříč celou organizací Honeywell. Pro svou činnost následuje modifikaci tohoto procesu, která zahrnuje samotný vývoj pomocí agilní metody SCRUM a využití metod projektového managementu.

Na základě vypracované analýzy byl navržen model hybridního fungování v softwarovém týmu spolu s činnostmi nutnými k jeho aplikaci a jeho ekonomickým zhodnocením. Odhad náročnosti realizace navrženého řešení je stanoven na 66 člověkodnů. Náklady jsou stanoveny na 526 000 Kč. Tento model je možné dále rozvíjet a přizpůsobovat měnícím se potřebám týmu.

Práce není striktně užitečná pouze pro zkoumaný subjekt, naopak výsledek práce je využitelný v obdobně založených týmech, které nenásledují klasický proces vývoje produktů. Nadále je práce díky své struktuře a výzkumu lehce znovupoužitelná pro další oblasti CMMI.

6 SEZNAM POUŽITÉ LITERATURY

- (1) RICHARDSON, G. L. *Project management theory and practice*. 2. vyd. Boca Raton: Auerbach Publications, 2014. 654 s. ISBN 9781482254952.
- (2) LARSON, E. W. a F. G. CLIFFORD. *Project management: the managerial process*. 5 vyd. New York: McGraw-Hill Irwin, 2011. 608 s. ISBN 0071289291.
- (3) DOLEŽAL, J., P. MÁČHAL a B. LACKO. *Projektový management podle IPMA*. 2. aktualiz. a dopl. vyd. Praha: Grada, 2012. 528 s. ISBN 978-80-247-4275-5.
- (4) ŠOCHOVÁ, Z. *CO JSOU AGILNÍ METODY?*. Sochova.cz [online]. 2016 [cit. 2016-05-18]. Dostupné z: <http://sochova.cz/co-jsou-agilni-metody.htm>
- (5) THE AGILE ALLIANCE. *Manifest Agilního vývoje software*. Agilemanifesto.org [online]. 2001 [cit. 2016-05-18]. Dostupné z: <http://www.agilemanifesto.org/iso/cs/>
- (6) ŠOCHOVÁ, Z. a E. KUNCE. *Agilní metody řízení projektů*. Brno: Computer Press, 2014. 176 s. ISBN 978-80-251-4194-6.
- (7) MYSLÍN, J. *Scrum: průvodce agilním vývojem softwaru*. Brno: Computer Press, 2016. 167 s. ISBN 978-80-251-4650-7.
- (8) SCRUM ALLIANCE. *The Scrum Guide*. ScrumAlliance.com [online]. 2014 [cit. 2016-05-18]. Dostupné z: <https://www.scrumalliance.org/why-scrum/scrum-guide>
- (9) COHN, M. *Succeeding with agile: software development using Scrum*. Upper Saddle River, NJ: Addison-Wesley, 2010. 504 s. ISBN 0321579364.
- (10) CMMI Product Team. *CMMI for Development, Version 1.3*. In: Software Engineering Institute, Carnegie Mellon University, 2010 [online]. [cit. 2016-05-19]. Dostupný z: <http://resources.sei.cmu.edu/library/asset-view.cfm?AssetID=9661>
- (11) CHRISSIS, M. B., M. KONRAD a S. SHRUM. *CMMI for development: guidelines for process integration and product improvement*. 3. vyd. Upper Saddle River, NJ: Addison-Wesley, 2011. ISBN 9780321711502.
- (12) HONEYWELL INTERNATIONAL INC. *2015 Annual Report*. In: Honeywell [online]. 2015 [cit. 2016-05-18]. Dostupné z: http://www.honeywell.com/-/media/Honeywell_com/Files/PDF/2015%20Annual%20Report%20pdf.pdf
- (13) HONEYWELL INTERNATIONAL INC. *Praha*. Honeywell.cz [online]. 2015 [cit. 2016-05-18]. Dostupné z: <http://honeywell.jobs.cz/prace-v->

honeywellu/brno/cz/?brand=g2&exportRCM=82037103&trackingBrand=honeywell.jobs.cz&rps=186&ep=

- (14) HONEYWELL INTERNATIONAL INC. *Olomouc*. Honeywell.cz [online]. 2015 [cit. 2016-05-18]. Dostupné z: <http://honeywell.jobs.cz/prace-v-honeywellu/olomouc/cz/?brand=g2&exportRCM=82037103&trackingBrand=honeywell.jobs.cz&rps=186&ep=>
- (15) HONEYWELL INTERNATIONAL INC. *Olomouc*. Honeywell.cz [online]. 2015 [cit. 2016-05-18]. Dostupné z: <http://honeywell.jobs.cz/prace-v-honeywellu/brno/cz/?brand=g2&exportRCM=82037103&trackingBrand=honeywell.jobs.cz&rps=186&ep=>
- (16) Honeywell, spol. s r.o. - HTS CZ o.z. *Projektové dokumenty týmu FSS*. Brno: Honeywell, spol. s r.o. - HTS CZ o.z. 2016.
- (17) Zaměstnanci týmu FSS. *Interview*. Honeywell, spol. s r.o. - HTS CZ o.z. Tuřanka 100/1387, Brno 627 00.
- (18) SUTHERLAND J., C. R. JAKOBSEN a K. JOHNSON. *Scrum and CMMI Level 5: The Magic Potion for Code Warriors*. In: IEEE: Agile Conference (Agile), Washington, DC 13.-17. srpna 2007 [online]. [cit. 2016-05-19]. Dostupný z: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4293608&isnumber=4293563>
- (19) MARCAL A. S. C. *Mapping CMMI Project Management Process Areas to SCRUM Practices*. In: IEEE: Software Engineering Workshop, Columbia, MD 8. února 2007 [online]. [cit. 2016-05-19]. Dostupný z: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4402760&isnumber=4402752>
- (20) GLAZER H., J. DALTON, D. ANDERSON. *CMMI or Agile: Why Not Embrace Both!* In: Software Engineering Institute, Carnegie Mellon University, 2008 [online]. [cit. 2016-05-19]. Dostupný z: <http://resources.sei.cmu.edu/library/asset-view.cfm?AssetID=8533>

SEZNAM OBRÁZKŮ

Obrázek č. 1: Trojimperativ projektu, Zdroj: Vlastní zpracování	16
Obrázek č. 2: Životní cyklus projektu, Zdroj: Upraveno dle (2, str. 7)	18
Obrázek č. 3: Proces řízení zainteresovaných stran, Zdroj: Upraveno dle (4, str. 51)	19
Obrázek č. 4: SCRUM, Zdroj: Upraveno dle (9).....	31
Obrázek č. 5: Komponenty procesních oblastí, Zdroj: Upraveno dle (11, str. 10).....	33
Obrázek č. 6: Oficiální logo společnosti Honeywell, Zdroj:Převzato z (12).....	37
Obrázek č. 7: Pobočka Honeywell v Brně, Zdroj: Převzato z (15)	38
Obrázek č. 8 Logo FSS, Zdroj: Upraveno dle (16).....	39
Obrázek č. 9: Organizační struktura FSS týmu, Zdroj: Upraveno dle (16).....	40
Obrázek č. 10: Weather Information Service, Zdroj: Upraveno dle (16).....	41
Obrázek č. 11: Metodiky a rámce týmu FSS, Zdroj: Vlastní zpracování	46
Obrázek č. 12: Fáze životního cyklu procesu IPDS, Zdroj: Upraveno dle (17)	48
Obrázek č. 13: Životní cyklus procesu vývoje mob. aplikací, Zdroj: Upraveno dle (16)	49
Obrázek č. 14: Struktura FSS týmu z pohledu SCRUMu, Zdroj: Upraveno dle (16)	52
Obrázek č. 15: Epic v systému JIRA, Zdroj: Vlastní zpracování	56
Obrázek č. 16: User Story v systému JIRA, Zdroj: Vlastní zpracování	57
Obrázek č. 17: Product Backlog v systému JIRA, Zdroj: Vlastní zpracování.....	57
Obrázek č. 18: Sprint Backlog v systému JIRA, Zdroj: Vlastní zpracování	58

SEZNAM TABULEK

Tabulka č. 1: SWOT analýza týmu FSS, Zdroj: Vlastní zpracování.....	45
Tabulka č. 2: Level 1 Backlog Grooming, Zdroj: Upraveno dle (16)	52
Tabulka č. 3: Level 1 Program Planning, Zdroj: Upraveno dle (16)	53
Tabulka č. 4: Level 2 Backlog Grooming, Zdroj: Upraveno dle (16)	53
Tabulka č. 5: Program Synchronization, Zdroj: Upraveno dle (16)	53
Tabulka č. 6: Level 2 Sprint Planning, Zdroj: Upraveno dle (16)	53
Tabulka č. 7: Level 3 Sprint Planning, Zdroj: Upraveno dle (16)	54
Tabulka č. 8: Sprint Demo, Zdroj: Upraveno dle (16).....	54
Tabulka č. 9: Program Sprint Demo, Upraveno dle (16).....	54
Tabulka č. 10: Sprint Retrospective, Zdroj: Upraveno dle (16)	54
Tabulka č. 11: Daily Scrum, Zdroj: Upraveno dle (16).....	54
Tabulka č. 12: Daily Scrums of Scrum, Zdroj: Upraveno dle (16)	55
Tabulka č. 13: Hybridní model CMMI, Zdroj: Vlastní zpracování.....	98
Tabulka č. 14: Vysvětlení modelových kombinací, Zdroj: Vlastní zpracování	98
Tabulka č. 15: Odhad náročnosti činností u jednotlivých variant, Zdroj: Vl. zpracování..	100
Tabulka č. 16: Odhad náročnosti realizace vlastního řešení, Zdroj: Vlastní zpracování....	101