

Česká zemědělská univerzita v Praze

Provozně ekonomická fakulta

Katedra systémového inženýrství



Diplomová práce

**Rozbor agilních přístupů v řízení IT projektů
ve vybrané firmě**

Bc. Magdalena Kočová

© 2016 ČZU v Praze

ZADÁNÍ DIPLOMOVÉ PRÁCE

Magdalena Kočová

Projektové řízení

Název práce

Rozbor agilních přístupů v řízení IT projektů ve vybrané firmě

Název anglicky

Analysis of the Agile Project Management in selected IT Company

Cíle práce

Cílem práce je popsání a rozbor současného stavu agilních přístupů ve vybrané firmě a vypracování vlastních návrhů na zlepšení stavu řízení IT projektů.

Metodika

1. Metodika:

Po nastudování vybrané odborné literatury a navázání spolupráce s danou firmou, bude zpracována literární rešerše o firmou používaných agilních přístupech a metodikách a o principech a nástrojích projektového řízení. Následně bude provedeno šetření za pomoci osobních rozhovorů s majiteli vybrané firmy za účelem vyhodnocení současné situace v oblasti agilních přístupů využívaných ve firmě. Na základě vyhodnocení budou vypracovány i vlastní návrhy na zlepšení stavu řízení IT projektů v dané firmě.

2. Harmonogram:

Studium vybrané odborné literatury: leden – srpen 2015

Navázání spolupráce s vybranou firmou: únor – červen 2015

Sběr dat a rozhovory s pracovníky firmy: červenec – listopad 2015

Rozbor agilního přístupu v řízení projektů: září 2015 – listopad 2016

Návrh vlastních doporučení: listopad – prosinec 2015

Sepsání literární rešerše: prosinec 2015 – únor 2016

Revize a kompletace textů: únor 2016

Doporučený rozsah práce

50 – 70 stran

Klíčová slova

Agilní přístup, metoda SCRUM, projektové řízení, projektový tým

Doporučené zdroje informací

A guide to the project management body of knowledge: (PMBOK guide). 4th ed. Newton Square: Project Management Institute, 2008, xxvi, 467 s. ISBN 978-1-933890-51-7.

DOLEŽAL, Jan, MÁCHAL, Pavel a LACKO, Branislav. Projektový management podle IPMA. 2., aktualiz. a dopl. vyd. Praha: Grada, 2012, 526 s. Expert (Grada). ISBN 978-80-247-4275-5.

KERZNER, Harold. Project management: a systems approach to planning, scheduling, and controlling. 10th ed. Hoboken, New Jersey: John Wiley & Sons, c2009, xxiv, 1094 s. ISBN 978-0-470-27870-3.

SCHWABER, Ken. Agile project management with Scrum. Redmond, Wash.: Microsoft Press, c2004, xix, 163 p. ISBN 073561993x.

SVOZILOVÁ, Alena. Projektový management. 2., aktualiz. a dopl. vyd. Praha: Grada, 2011, 380 s. Expert (Grada). ISBN 978-80-247-3611-2.

Předběžný termín obhajoby

2015/16 LS – PEF

Vedoucí práce

Ing. Jan Bartoška, Ph.D.

Garantující pracoviště

Katedra systémového inženýrství

Elektronicky schváleno dne 18. 11. 2015

doc. Ing. Tomáš Šubrt, Ph.D.

Vedoucí katedry

Elektronicky schváleno dne 18. 11. 2015

Ing. Martin Pelikán, Ph.D.

Děkan

V Praze dne 18. 03. 2016

Čestné prohlášení

Prohlašuji, že svou diplomovou práci "Rozbor agilních přístupů v řízení IT projektů ve vybrané firmě" jsem vypracovala samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou citovány v práci a uvedeny v seznamu použitých zdrojů na konci práce. Jako autorka uvedené diplomové práce dále prohlašuji, že jsem v souvislosti s jejím vytvořením neporušila autorská práva třetích osob.

V Praze dne 29.3.2016

Poděkování

Ráda bych touto cestou poděkovala Ing. Janu Bartoškovi, Ph.D. za odborné vedení a pomoc při zpracování diplomové práce. Dále chci poděkovat společnosti JuicyMo s.r.o. za možnost s ní spolupracovat a poskytnutí jejích interních informací. V neposlední řadě bych chtěla poděkovat své rodině a přátelům za podporu a pomoc při psaní této práce.

Rozbor agilních přístupů v řízení IT projektů ve vybrané firmě

Souhrn

Diplomová práce se zaměřuje na problematiku agilního přístupu Scrum v řízení IT projektů, konkrétně na popis a rozbor stávajícího stavu agilních přístupů ve vybrané firmě a následné vypracování vlastních návrhů na zlepšení stavu řízení IT projektů. Literární rešerše se zabývá základními charakteristikami a pravidly agilního přístupu Scrum. Ve vlastní části práce je představena firma JuicyMo s.r.o., po jejíž stručné charakteristice je popsán a rozebrán její současný stav agilního řízení IT projektů. V práci jsou především detailně popsány případy, ve kterých se řízení firmy odlišuje od definicí a pravidel Scrum. Součástí práce jsou zároveň vypracované konkrétní návrhy, které mají vést ke zlepšení stavu řízení vybrané společnosti.

Klíčová slova: Agilní přístup, metoda SCRUM, projektové řízení, projektový tým.

Analysis of the Agile Project Management in Selected IT Company

Summary

The thesis focuses on the issue of Agile approach Scrum in IT project management, namely the description and analysis of the current state of Agile approaches in selected company and the subsequent drafting own proposals for improvement of the management of IT projects. A literature background review deals with the basic characteristics and rules of agile approach Scrum. In the practical part of the Thesis is introduced the company JuicyMo s.r.o. After a brief description of the company, the current state of agile approach to IT project management is described and analyzed. The thesis primarily described in detail cases in which the management of the company differs from the definitions and rules of Scrum. The thesis also drawn up concrete proposals on how to improve the current state of management of the selected company.

Keywords: Agile Project Management, Scrum, Project Management, Project Team.

OBSAH

1	Úvod.....	12
2	Cíl práce a metodika	13
2.1	Cíl práce	13
2.2	Metodika	14
2.3	Harmonogram zpracování.....	15
3	Teoretická východiska	16
3.1	Scrum	16
3.1.1	Historie Scrum	17
3.1.2	Pilíře Scrum	19
3.1.2.1	Pilíř I – Transparentnost	19
3.1.2.2	Pilíř II – Kontrola	19
3.1.2.3	Pilíř III – Přizpůsobení	20
3.1.3	Scrum Tým	21
3.1.3.1	Product Owner.....	23
3.1.3.2	Scrum Master.....	24
3.1.3.3	Vývojářský tým	25
3.1.4	Události Scrum	27
3.1.4.1	Sprint	27
3.1.4.2	Sprint Planning	29
3.1.4.3	Daily Scrum.....	31
3.1.4.4	Sprint Review	32
3.1.4.5	Sprint Retrospective	34
3.1.4.6	Backlog Grooming	35
3.1.5	Artefakty Scrum.....	37
3.1.5.1	Product Backlog	37

3.1.5.2	Sprint Backlog	39
3.1.5.3	Inkrement.....	41
3.1.6	Úkoly, User Stories a Epics	42
3.1.6.1	Úkol	42
3.1.6.2	User Story.....	42
3.1.6.3	Epics	43
4	Vlastní práce	44
4.1	Popis vybrané firmy	44
4.1.1	Základní údaje.....	44
4.1.2	Historie a současnost firmy.....	45
4.1.3	Organizační a řídicí struktura	46
4.2	Současný stav řízení IT zakázek v Juicyo s.r.o.	49
4.2.1	Agilní přístup řízení IT zakázek v Juicyo s.r.o.	49
4.2.2	Životní cyklus IT zakázek v Juicyo s.r.o.	50
4.2.3	Užívané SW nástroje na podporu řízení IT zakázek.....	52
4.2.3.1	Přehled používaných SW nástrojů na řízení IT zakázek.....	52
4.2.3.2	Scrum řízený pomocí OpenProject.....	58
4.3	Rozbor nasazených modifikací metody Scrum v Juicyo s.r.o.	61
4.3.1	Přizpůsobení rolí projektového týmu Juicyo s.r.o. pro řízení projektu	62
4.3.1.1	Náplň práce jednotlivých členů projektového týmu a jejich zodpovědnosti	63
4.3.2	Stavy úkolů při vývoji Inkrementu a jejich rozšíření v Juicyo s.r.o.	66
4.3.2.1	Zodpovědnosti členů týmu k vykonání úkolu v určitém stavu.....	66
4.3.2.2	Stavy v procesu práce na úkolech	68
4.3.2.3	Stavy úkolů v uživatelském rozhraní OpenProject	69
4.3.3	Změna ve vedení Daily Scrums	71

4.4	Vlastní návrhy na zlepšení řízení IT zakázek ve firmě Juicyo s.r.o.....	72
4.4.1	Zřízení komunikačního kanálu pro Daily Scrums	72
4.4.2	Zlepšení odhadů časové náročnosti úkolů při Sprint Planning.....	74
4.4.3	Zlepšení ve sledování vytížení lidských zdrojů.....	77
4.4.3.1	Zřízení dokumentace sledující přiřazení lidských zdrojů k jednotlivým projektům 78	
4.4.3.2	Zavedení pracovního kalendáře pro lidské zdroje.....	79
5	Výsledky a diskuse	82
6	Závěr	85
7	Zdroje.....	87

SEZNAM OBRÁZKŮ

Obrázek 1 - Sprint.....	28
Obrázek 2 – Sprint planning	30
Obrázek 3 - Daily Scrum	31
Obrázek 4 – Sprint Review	33
Obrázek 5 - Sprint Retrospective.....	34
Obrázek 6 – Činnosti aktivity Grooming.....	36
Obrázek 7 – Product Backlog	37
Obrázek 8 – Proces sestavení a seřazení Product Backlog.....	38
Obrázek 9 – Sprint Backlog s konkrétními úkoly.....	39
Obrázek 11 – Scrum Board na začátku, v průběhu a na konci Sprintu	40
Obrázek 10 – Ukázka Task Board	40
Obrázek 12 – Inkrement Produktu.....	41
Obrázek 13 – Znázornění úkolů.....	42
Obrázek 14 – Ukázka Product Backlog a Sprint Backlog firmy Juicyo s.r.o.....	59
Obrázek 15 – Ukázka Burndown grafu z probíhajícího Sprintu firmy Juicyo s.r.o.	60
Obrázek 16 – User Story v průběhu Sprintu.....	70
Obrázek 17 – Navržená podoba Daily Scrum zpráv ve Slack	73

SEZNAM DIAGRAMŮ

Diagram 1 – Organizační struktura Juicymo s.r.o.	47
Diagram 2 – Řídící struktura Juicymo s.r.o.	47
Diagram 3 – Obsazení manažerských rolí Juicymo s.r.o.	48
Diagram 4 – Životní cyklus IT zakázek v Juicymo s.r.o.	50
* Diagram 5 – Iterace.....	51
Diagram 6 – Příklady projektových týmů v Juicymo s.r.o.	62
Diagram 7 – Stavový diagram, Stavů úkolů.....	68

SEZNAM TABULEK

Tabulka 1 – Raci matice odpovědnosti týmových rolí vůči danému softwaru.....	52
Tabulka 2 – Funkce sw nástrojů ve firmě Juicymo s.r.o.	53
Tabulka 3 - RACI matice odpovědnosti příslušných oddělení / osob vůči sw FatFreeCRM..	56
Tabulka 4 – RACI matice – Zodpovědnosti jednotlivých členů projektového týmu	63
Tabulka 5 – Srovnání fází Scrum vs. Juicymo s.r.o	66
Tabulka 6 – RACI matice, Zodpovědnosti členů týmu k vykonání úkolu v určitém stavu.....	67
Tabulka 7 – Pracovní kalendář vývojáře	80
Tabulka 8 – Pracovní kalendář souhrnný	80

1 Úvod

V dnešní době, kdy moderní IT technologie nás provází na každém kroku, je jejich vývoj, rychlé a cílené uvedení do praxe a jejich stálá aktualizace jedním ze základních stavebních kamenů úspěšného podnikání v IT oboru. Pro udržení konkurenceschopnosti musí firmy být schopné pružně řešit vývoj trhu jak současných softwarových aplikací, tak i nových technologií, které na trh teprve pronikají.

Rychlý technologický pokrok v oblasti informačních technologií a informačních systémů se musí odrážet i ve způsobu řízení IT projektů. Z důvodů častých požadavků na změnu či modifikování výstupů projektu, na které tradiční projektové řízení není schopno efektivně reagovat, projektoví manažeři byli donuceni nalézt nové přístupy k vedení IT projektů. Jedním z těchto přístupů je agilní přístup Scrum, kterým se zabývá tato diplomová práce.

Scrum je v současné době jedním z nejoblíbenějších agilních přístupů díky své jednoduchosti a flexibilitě. Vzhledem k jeho popularitě, mnoho organizací tvrdí, že „provozují Scrum“, ale ve skutečnosti se ani zdaleka neblíží jeho skutečné definici – ač je Scrum jednoduchý na pochopení, je zároveň extrémně obtížný pro dokonalé zvládnutí. Scrum totiž neudává, co člověk „musí udělat“, aby dosáhl vytyčeného cíle. Namísto toho definuje specifické pojmy a postupy, které organizacím umožňují se hladce přizpůsobit rychle se měnícím požadavkům na vytvářený produkt. Vhodné zvolení způsobu používání Scrum je čistě na organizacích samotných, a právě v tom se skrývá ona záludnost dokonalého zvládnutí této agilní metodiky.

Teoretická část práce má za účel čtenáře seznámit s agilním přístupem Scrum, jeho historií, základními pilíři, událostmi, artefakty a rolemi lidí ve Scrum týmu. Ve vlastní části práce je představena firma JuicyMo s.r.o., která k řízení IT projektů Scrum používá. Součástí této části je popis a rozbor řízení IT projektů ve firmě. Výstupem práce je vypracování vlastních návrhů na zlepšení současného řízení firmy JuicyMo s.r.o.

2 Cíl práce a metodika

2.1 Cíl práce

Cílem je vypracování vlastních návrhů na zlepšení stavu řízení IT projektů na základě popsání a rozboru současného stavu agilních přístupů ve vybrané firmě.

Výstupem práce jsou tedy nejen zpracované výstupy rozboru agilního řízení IT projektů v dané firmě, ale i konkrétní návrhy, které mají vést ke zlepšení stavu řízení vybrané společnosti.

2.2 Metodika

První fází ve vypracování práce bude studium vybrané odborné literatury a internetových zdrojů. V průběhu studia literatury bude rovněž navázána spolupráce s vybranou firmou, která ke svému řízení využívá agilní přístup Scrum.

Po studiu literatury bude zpracována literární rešerše, pro jejíž sepsání bude využita metoda syntézy dat z odborné literatury a internetových zdrojů.

Po získání přehledu o agilní metodě Scrum, proběhne sběr podkladových dat z prostředí vybrané společnosti. Data budou získána z interních zdrojů společnosti, dále pomocí pozorování a rozhovorů se zaměstnanci firmy. Na základě sebraných dat bude proveden rozbor agilního přístupu v řízení IT projektů vybrané firmy. Rozbor bude zaměřen na to, jakým způsobem společnost agilní metodiku Scrum používá. Cílem bude zjistit, zda společnost dodržuje pravidla metodiky Scrum, či si metodiku nějakým způsobem přizpůsobuje a modifikuje, respektive jakým.

V poslední fázi budou pro společnost vypracovány vlastní návrhy na zlepšení stavu řízení IT projektů. K tomu budou použity jak výstupy z předchozích fází, tak i znalosti nabyté při studiu oboru Projektové řízení.

2.3 Harmonogram zpracování

leden - srpen 2015

Studium vybrané odborné literatury.

únor - červen 2015

Navázání spolupráce s vybranou firmou.

červenec - listopad 2015

Pozorování, sběr dat a rozhovory s pracovníky firmy.

září 2015 - únor 2016

Rozbor agilního přístupu v řízení projektů.

prosinec 2015 - únor 2016

Návrh vlastních doporučení.

prosinec 2015 - únor 2016

Sepsání literární rešerše.

březen 2016

Revize a kompletace textů.

3 Teoretická východiska

3.1 Scrum

Scrum je procesní rámec pro vyvíjení a udržování komplexních produktů. Lidé v něm mohou řešit složité adaptivní problémy a současně produktivně a tvůrčím způsobem dodávat výrobky s co nejvyšší přidanou hodnotou.[18]

Kromě toho, že Scrum je velmi jednoduchý a srozumitelný, je zároveň extrémně obtížný pro dokonalé zvládnutí. [18]

Procesní rámec Scrum se skládá ze Scrum týmů a z rolí v nich obsažených, z událostí, artefaktů a pravidel. Pravidla jsou určena vztahy a interakcemi mezi činnostmi, rolemi a artefakty. Každá součást procesního rámce Scrum slouží ke konkrétnímu účelu a je zásadní a nezbytná pro jeho úspěšné nasazení. [18]

V dnešní době se Scrum využívá především v oboru informačních technologií, a to ve vyvíjení a udržování softwarových produktů. Mezi další využívané obory patří marketing, inženýrství, vzdělávání či operace. Konkrétní strategie pro použití metodiky Scrum se v každém zmíněném oboru může odlišovat. [18]

3.1.1 Historie Scrum

Pojem „Scrum“, v překladu „Skrumáž“, byl poprvé použit roku 1986 manažery Takeuchi a Nonaka v publikaci *“The New New Product Development Game”*. V této knize odkazují na hru rugby a zdůrazňují význam týmů a některé analogie mezi týmovým sportem a úspěšností při vyvíjení nových produktů. Zároveň zde uveřejňují výzkum, na jehož výsledcích jsou postaveny základní předpoklady pro fungování Scrum v organizaci. Nejdůležitější výsledky výzkumu jsou následující [14]:

- Vynikajícího výkonu při vývoji nových složitých produktů je dosaženo, když malým a sebeorganizujícím se týmům jsou předkládány cíle, nikoli úkoly.
- Nejlepší týmy jsou ty, kterým je udán směr, ve kterém mají prostor, aby vypracovaly svoji vlastní taktiku, jak nejlépe směřovat k jejich společnému cíli.
- K dosažení vynikajících výsledků týmy potřebují samostatnost.

Samotná metodika Scrum byla koncipována na začátku 90. let. Jejími zakladateli byli Jeff Sutherland a Ken Schwaber. Roku 1995 byl Scrum kodifikován na konferenci OOPSLA (Object-Oriented Programming, Systems, Languages & Applications) v Austin, Texas. Pro tuto konferenci byla vydána publikace *„SCRUM Software Development Process“*. [14]

Metodika Scrum pro vývoj softwaru implementuje zásady popsané ve výše zmíněném dokumentu v rozvoji a udržování komplexních softwarových produktů. [14]

V počátečním stádiu vývoje a používání metodiky Scrum byl osloven profesor Babatunde A. Ogunnaike Tunde, uznávaný odborník řízení procesu výzkumu. Byl požádán, zda by se mohl odborně vyjádřit k současným procesům využívaným pro vývoj softwaru. Prof. Tunde prostudoval několik komerčních metodik používaných pro vývoj softwaru, jako jsou například vodopádové či prediktivní modelování. Z jeho studií vyplynulo, že empirický přístup metodiky Scrum je pro softwarový vývoj nejvíce vhodný. [14]

Scrum byl poprvé vyzkoušen a zdokonalen ve společnostech Individual, Inc., Fidelity Investments, and IDX (nyní GE Medical). [14]

Rok 2001 byl klíčový pro rozvoj Scrum. V únoru tohoto roku se 17 softwarových vývojářů, mezi nimiž byli i Jeff Sutherland a Ken Schwaber, podílelo na sestavení „Agilního manifestu softwarového vývoje“. Ve stejném roce rovněž vznikla Agilní aliance, které jako první předsedal Ken Schwaber. [14]

V následujícím roce 2002 Ken Schwaber společně s Mikem Cohnem a Esther Derby založili Scrum alianci, které opět předsedal Ken Schwaber. Následující roky se vyznačovaly úspěchy v zavedení certifikačních programů pro Scrum Masters a jejich obdobných derivací. [14]

V roce 2006 Jeff Sutherland založil svou vlastní společnost, Scrum.inc, přičemž nadále nabízel a učil certifikační kurzy Scrum. [14]

Ken Schwaber opustil Scrum alianci na podzim roku 2009, a založil Scrum.org, aby mohl nadále zlepšovat kvalitu a účinnost Scrum, a to hlavně prostřednictvím Professional Scrum kurzů. [14]

Jeff Sutherland a Ken Schwaber v roce 2010 prvně zveřejnili Příručku Scrum, tzv. „*Scrum Guide*“, jež byla postupně aktualizována v letech 2011 a 2013. Tato příručka posloužila k založení celosvětově uznávané znalostní báze metodiky Scrum. [14]

3.1.2 Pilíře Scrum

Metodika Scrum je založena na empirickém řízení procesů, tzv. empirismu. Empirické řízení procesů se vyznačuje tím, že poznání přichází ze zkušenosti a rozhodování je založené na tom, co je známo. [18]

Všechny implementace empirismu, a tím pádem i metodika Scrum, jsou postaveny na třech pilířích. Těmito pilíři jsou [18]:

1. transparentnost,
2. kontrola,
3. přizpůsobení.

3.1.2.1 Pilíř I – Transparentnost

Základním rysem tohoto pilíře je, že podstatné aspekty procesu musí být zřetelné pro osoby odpovědné za výsledek. Tyto aspekty musí být definovány společnou normou tak, aby pozorovatelé sdíleli stejné porozumění toho, co vidí. [18]

Transparentnost musí být rovněž zachovávána i v komunikaci. V procesu používaná řeč musí být stejně chápána všemi účastníky. Například definice stavu „Hotovo“, tj. „Done“, musí být stejná jak pro pracovníky, který daný produkt vyrábějí, tak i pro ty, co daný produkt akceptují. [18]

3.1.2.2 Pilíř II – Kontrola

Uživatelé metodiky Scrum musí často kontrolovat Scrum artefakty a progres v naplňování cílů Sprintu, aby se odhalily nežádoucí odchylky v procesu vývoje. Četnost kontrol by měla být optimalizována tak, aby v co nejmenší míře narušovala pracovní činnosti. [18]

Aby byly kontroly co nejefektivnější, doporučuje se, aby je prováděl kvalifikovaný pracovník přímo na pracovišti v okamžiku provádění práce. [18]

3.1.2.3 Pilíř III – Přizpůsobení

Třetí pilíř, *Přizpůsobení*, úzce navazuje na pilíř Kontroly. Pokud pověřený pracovník při kontrole objevil, že jeden nebo více aspektů procesu se odchýlil mimo přijatelné meze, a tudíž výsledný produkt by se stal nepřijatelný, proces či materiál, který je zpracováván, musí být upraven. Úprava procesu, resp. materiálu, musí být provedena co nejdříve, aby se minimalizovaly další odchylky. [18]

Scrum předepisuje čtyři formální události pro kontrolu a úpravu [18]:

- 1) Sprint Planning
- 2) Daily Scrum
- 3) Sprint Review
- 4) Sprint Retrospective

Více o těchto událostech pojednává kapitola 3.1.4.

3.1.3 Scrum Tým

Model Scrum Týmu je navržen tak, aby optimalizoval flexibilitu, kreativitu a produktivitu. Scrum definuje tři role v Scrum týmu – Product Owner, Scrum Master a člen Vývojářského týmu. Členu Scrum týmu musí být přiřazena vždy jedna ze tří rolí definovaných metodikou Scrum, ne více. Pokud by člen Scrum týmu zastupoval více rolí, nastal by u něj střet zájmů. Veškerá odpovědnost za vedení projektu je rozdělena mezi zmíněné tři role. [5]

V metodice Scrum neexistuje role projektového manažera [6]. Odpovědnosti “tradičního” projektového manažera jsou rozděleny do scrum rolí, kdy Product Owner spravuje produkt a návaznost investic, Scrum Master řídí proces vývoje a vývojářský tým ovládá sám sebe.[10]

O blíže specifikovaných povinnostech a odpovědnostech každé ze Scrum rolí pojednávají podkapitoly níže.

Hlavními dvěma charakteristikami Scrum týmu jsou sebe-organizace a multifunkčnost. Sebe-organizující se týmy si sami určují, jak nejlépe zvládnout přidělenou práci. Z definice Scrum nesmí do jejich práce zasahovat nikdo vně týmu. [10]

Sebe-organizace v případě Scrum neznamena přístup laissez-faire, naopak, sebe-organizované týmy jsou vysoce disciplinované. Sebe-organizace týmu nastává až po dohodnutí určitých omezení. V mezích těchto omezení je týmu dána plná autonomie a dobrovolnost v přijímání závazků k sobě samým. Členové Scrum týmu přijímají zodpovědnost a odpovědnost za dostání jejich závazků v mezích jejich možností. Jsou podporováni v tom, aby na sebe brali přiměřená rizika a aby se učili prostřednictvím selhání a sebereflexe. Sebe-organizující se týmy vykazují vysokou důvěru a mají vysokou vnitřní motivaci. [10]

Správné týmové sebe-organizaci musí předcházet sebe-organizace každého z členů týmu. Každý den všichni členové musí koordinovat svou práci se zbytkem týmu. K synchronizaci členů týmu se následně využívají pravidelná setkání, jako jsou například Stand-Up meetings, Sprint Review nebo Sprint Retrospectives. [11]

Multifunkční týmy mají mezi jejich členy všechny nutné kompetence k vykonání přidělené práce, a tudíž nejsou závislé na někom, kdo není součástí jejich týmu. [10]

Scrum týmy dodávají vytvářený produkt iteračně a postupně. Takto je maximalizována možnost zpětné vazby pro kontrolu, adaptaci a další vývoj a plánování. Další výhodou je, že postupné dodávání „Hotového“ („Done“) produktu zaručuje, že potenciálně použitelná verze vyvíjeného produktu je vždy dostupná. [10]

Hodnoty Scrum týmu

Všechny činnosti vykonávané pomocí metodiky Scrum potřebují sadu hodnot, které jsou schopny tvořit základ týmových procesů a interakcí a které vedou ke zdravému rozvoji týmu a k úspěchu. Do této sady patří hodnoty [17]:

1. Soustředěnost
 - Tým se soustředí pouze na pár věcí najednou, členové spolu spolupracují a předkládají tak výborné pracovní výsledky.
2. Odvaha
 - Pracovníci pracují jako tým, cítí se podporovaní a mají k dispozici více zdrojů. To jim dodává odvahu přijímat větší výzvy.
3. Otevřenost
 - Členové týmu si navzájem sdělují, jak se jim daří, co jim stojí v cestě vývoje a další pracovní záležitosti, aby případné problémy mohly ihned řešit.
4. Zavázanost
 - Tým je sebe-organizovaný, třímá kontrolu nad svým „vlastním osudem“, a proto je více zavázán k jeho úspěchu, resp. neúspěchu.
5. Respekt
 - Členové týmu sdílejí úspěchy i neúspěchy. To vede k vzájemnému respektu a pomoc druhému se stane hodna uznání.

3.1.3.1 *Product Owner*

V každém Scrum týmu zastává vždy jedna osoba roli Product Owner. Product Owner bývá obvykle vedoucí uživatel informačního systému nebo někdo z marketingového oddělení či produktového managementu. Také jím může být někdo s plným pochopením uživatelů, trhu, konkurence anebo budoucích trendů pro doménu nebo typ systému, který je vyvíjen. [15]

Odpovědnosti Product Owner jsou následující [10]:

- zajištění existence společné vize produktu,
- správa a prioritizace artefaktu Product Backlog,
- pomáhat členům týmu pochopit, co se bude vytvářet a proč,
- akceptování dodaného inkrementu produktu na konci každé iterace,
- správa plánu pro spuštění systému,
- informování zúčastněných stran o pokroku a řízení jejich očekávání,
- maximalizace ekonomické návratnosti produktu.

Primární odpovědnost Product Owenera je optimalizace návratnosti investice, tzv. ROI. Optimální návratnost investice je dosahována tím, že členové Scrum týmu se podílejí na dodávání aktuálně nejhodnotnější vlastnosti vytvářeného produktu [10]. Způsoby, jakými je toho dosaženo, se mohou lišit napříč organizacemi, Scrum týmy i jednotlivci [18].

Product Owner je zpravidla klíčový stakeholder projektu. Jednou z jeho povinností je mít vizi toho, co chce vytvořit. Vize se hodně liší podle toho, zda Scrum tým vyvíjí komerční software, software pro vnitřní použití, hardware nebo nějaký jiný typ produktu. Klíčem k úspěchu je, že osoba v roli Product Owner musí mít představu toho, co má být vytvořeno a sdělit tuto představu Scrum týmu. To činí skrze správu artefaktu Product Backlog, což je seznam vlastností produktu, které jsou seřazené podle jejich priorit. [15]

Správa Product Backlog zahrnuje tyto činnosti [18]:

- jasná formulace Product Backlog položek,

- seřazení položek v Product Backlog tak, aby Product Backlog co nejlépe odrážel cíle a poslání vytvářeného produktu,
- optimalizace hodnoty vykonávané práce vývojářského týmu,
- zajištění viditelnosti, transparentnosti a jasnosti obsahu Product Backlog,
- ukázání toho, na čem bude Scrum tým v nejbližší době pracovat,
- ujištění se, že vývojářské tým rozumí všem položkám Product Backlog.

Správu Product Backlog může vykonávat Product Owner sám anebo tým může pověřit vývojářský tým. V každém případě Product Owner za Product Backlog a jeho správu nese zodpovědnost. [18]

Jak již bylo zmíněno, v jednom Scrum týmu může figurovat pouze jeden Product Owner, Product Owner tudíž nemůže být komise. Komisi může pouze zastupovat a reprezentovat její zájmy. [18]

Aby Product Owner dosáhl úspěchu, celá organizace musí respektovat jeho rozhodnutí, která jsou viditelná v obsahu a uspořádání Product Backlog. Pokud by někdo chtěl změnit pořadí položek v Product Backlog, musí to učinit přes Product Owner. Nikdo jiný není oprávněn pověřit vývojářský tým, aby pracoval podle jiných požadavků, a vývojářský tým nemá dovoleno pracovat podle toho, co řekne někdo jiný. [18]

3.1.3.2 Scrum Master

Roli Scrum Master zastává v Scrum týmu jedna osoba. Ta je zodpovědná za to, že Scrum je všemi chápán a dodržován. Toho Scrum Masters dosahují tím, že zajišťují, aby Scrum týmy znaly a ovládaly teorii, praktiky a pravidla metodiky Scrum. [18]

Roli Scrum Master obvykle zastává bývalý projektový manažer nebo vedoucí technického týmu, může jím být ale kdokoli. [23]

Scrum Master řídí všechny aspekty týkající se procesů ve Scrum týmu. Vzhledem k tomu, že Scrum tým je sebe-organizační, Scrum Master k řízení využívá spíše vlivu než autority. A právě řízení pomocí vlivu je pro vedení většiny organizací úplná změna paradigmatu. Je to jedna z nejtěžších věcí k uchopení v metodice Scrum a zároveň nejlepší místo umožňující zlepšení lidské výkonnosti [10]. Scrum Master totiž dělá vše pro to, aby

pomohl týmu fungovat na nejvyšší možné úrovni. To zahrnuje například odstranění veškerých zábran v pokroku a zprostředkovávání setkání [23].

Scrum Master pomáhá osobě v roli Product Owner v několika různých směrech. Mezi ty nejdůležitější patří hledání vhodných technik pro efektivní Product Backlog management; pomoc při objasňování Scrum týmu, proč položky v Product Backlog musí být jasné a výstižné; pochopení plánování produktů v empirickém prostředí; kontrolování, že Product Owner je schopen uspořádat Product Backlog tak, aby dosahoval maximální možné hodnoty; porozumění a praktikování agilních přístupů a v neposlední řadě podle potřeby a požadavků moderování Scrum událostí. [18]

Vývojářský tým Scrum Master trénuje v sebe-organizaci a v multi-funkcionalitě. Pomáhá mu vytvářet vysoce hodnotné produkty, odstraňuje překážky bránící v jeho rozvoji a pomáhá vývojářskému týmu v organizačním prostředí, ve kterém Scrum není ještě plně přijat a pochopen. [18]

Scrum Master nepomáhá pouze Scrum týmu, ale zároveň slouží i organizaci, a to hned několika způsoby: plánuje implementaci metodiky Scrum v organizaci; vede a školí pracovníky organizace v osvojování si Scrum; pomáhá pracovníkům a stakeholderům porozumět a osvojit si Scrum a empirický vývoj produktu; iniciuje změny, které vedou k vyšší produktivitě Scrum týmu a v neposlední řadě spolupracuje s ostatními Scrum Masters na zvýšení efektivnosti metodiky Scrum v organizaci. [18]

Scrum Master je také často vnímán jako ochránce týmu. Nejběžnějším příkladem je skutečnost, že Scrum Master chrání tým tím, že zajistí, aby se tým kvůli tlaku ze strany příliš agresivního Product Owner nezavázal k něčemu, co by nebyl schopen během stanoveného Sprintu zvládnout. Na druhou stranu, dobrý Scrum Master také chrání tým od samolibosti. [23]

3.1.3.3 Vývojářský tým

Vývojářský tým je tvořen skupinou osob, které jsou zodpovědné za dodání „hotového“ Inkrementu produktu na konci každého Sprintu. Na inkrementu produktu pracují vždy a pouze členové vývojářského týmu. [10]

Organizace vytváří vývojářské týmy a následně je zmocní k organizování a řízení jejich vlastní práce. Výsledná synergie výrazně zlepšuje celkovou účinnost a efektivitu vývojářského týmu. [18]

Stejně jako Product Owner a Scrum Master, tak i vývojářský tým je vyznačován specifickými charakteristikami. Jednou z nich je rovnoprávné postavení všech členů uvnitř vývojářského týmu. Ve vývojářském týmu neexistují jiné „tituly“ než *vývojář*, bez ohledu na práci, kterou daná osoba zrovna vykonává. Toto pravidlo platí ve Scrum bez výjimek. [18]

Další charakteristikou vývojářského týmu je jeho multi-funkcionalita. Ta neznámá, že by každý člen týmu měl umět provádět všechny úkoly, ale že tým zahrnuje všechny kompetence nutné ke zvládnutí zadané práce. [10]

Vývojářský tým je také sebe-organizující se. Nikdo, a to ani Scrum Master, nemá právo vývojářskému týmu říkat, jak zacházet s Product Backlog a jakým způsobem jeho položky uspořádat do uveřejnitelných inkrementů. [18]

Vývojářský tým se již dále nedělí na další, menší týmy. Toto Scrum pravidlo platí bez výjimky, a to i v případě zahrnutí do projektu specifických činností, jako je například testování produktu nebo business analýzy. [18]

Poslední, ale ne méně důležitou charakteristikou je, že jednotliví členové vývojářského týmu mohou mít odborné schopnosti a oblastí zaměření, ale odpovědnost patří vývojářskému týmu jako celku. [18]

Scrum neudává, kolik členů by měl vývojářský tým přesně obsahovat. Říká však, že tým by měl být dostatečně malý na to, aby zůstat pohotový, a dostatečně velký, aby dokončit významnou část práce v rámci Sprintu. Příručka Scrum upozorňuje, že v týmech o méně než třech členech klesá interakce, což vede k menšímu růstu produktivity. V menších týmech také může nastat situace, že tým nebude mít během Sprintu dostatečné kompetence k vytvoření a dodání Inkrementu. Na druhou stranu, tým o více jak devíti členech vyžaduje nadměrnou koordinaci. Velké vývojářské týmy jsou příliš komplexní pro řízení empirického procesu. [18]

3.1.4 Události Scrum

Procesní rámec Scrum si lze představit prostřednictvím a odpovídajícím artefaktům. Mezi události předepsané metodikou Scrum patří Sprint, Sprint Planning, Daily Scrum, Sprint Review a Sprint Retrospective. Všechny Scrum události jsou časově ohraničené, tj. v projektu mají předem stanovenou maximální dobu trvání. [12]

Centrálním bodem pro všechny události je Sprint. Všechny ostatní události jsou formální příležitostí pro kontrolu a případné přizpůsobení. Události Scrum jsou speciálně navrženy tak, aby maximálně umožnily transparentnost a kontrolu procesu. Vynechání kterékoliv ze zmíněných událostí vede ke snížení transparentnosti a ztrátě příležitostí pro kontrolu a případnou adaptaci. [18]

3.1.4.1 *Sprint*

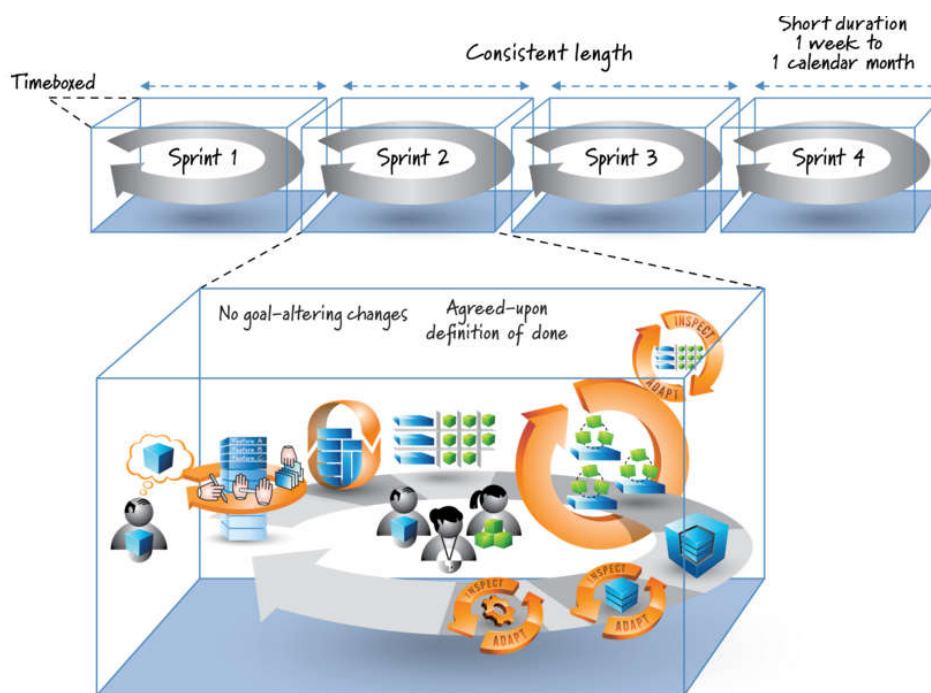
Jádrem celého Scrum je Sprint. Sprint je časově ohraničená událost s maximální dobou trvání jeden měsíc. Jednoměsíční délka Sprintu byla určena z toho důvodu, protože v případě delšího časového intervalu by se mohla změnit definice koncového produktu, složitost nebo výše podstupovaného rizika. Sprints zajišťují nejméně jednou za kalendářní měsíc kontrolu a případné přizpůsobení progresu vůči cíli Sprintu. Svou krátkou dobou trvání také omezují nebezpečí pouze na jeden měsíc ušlých nákladů. [18]

Jakmile Sprint započne, jeho délku už není možné zkrátit ani prodloužit [18]. V projektu je nejlepší, když Sprints mají konzistentní doby trvání a mohou tudíž vytvořit kadenci. [5]

Na konci Sprintu by vývojářský tým měl být schopen předat „Hotový“, použitelný a potencionálně vydatelný Inkrement vytvářeného produktu. Z toho vyplývá, že počet Sprintů v projektu není omezen – Sprints se opakují tolikrát, dokud není konečný produkt hotov. Jakmile jeden Sprint skončí, ihned po něm začne Sprint další. [18]

Sprints se skládají a zahrnují Sprint Planning (Plánování Sprintu), Daily Scrums (Denní schůze), vývojářskou práci, Sprint Review (Vyhodnocení Sprintu) a Sprint Retrospective (Retrospektivu Sprintu) [18]. Průběh Sprintu znázorňuje Obrázek 1.

Obrázek 1 - Sprint



Copyright © 2012, Kenneth S. Rubin and Innolution, LLC. All Rights Reserved.

Zdroj: [26]

Během Sprintu musí být dodržována tři základní pravidla [18]:

1. Nesmí se provádět změny, které by ohrozily stanovený cíl Sprintu.
2. Kvalita vytvářeného Inkrementu se nesmí snížit.
3. V závislosti na nových poznacích, vývojářský tým může Product Owner žádat o upřesnění rozsahu či o jeho znovu projednání.

Ve velmi ojedinělých případech Sprint může být během jeho trvání zrušen. Tato situace může nastat v případě, že cíl Sprintu se stane zastaralým, resp. za daných podmínek přestane dávat smysl – což ale díky krátkým časovým trváním Sprintů nastává velmi výjimečně. Oprávnění zrušit Sprint během jeho konání má pouze Product Owner, i když tak může učinit na popud stakeholderů, vývojářského týmu nebo Scrum Master. [18]

V případě zrušení Sprintu všechny „Hotové“ položky Product Backlog přezkoumány. Jestliže je část produktu použitelná pro případné vydání, Product Owner ji obvykle akceptuje. Ostatní nedodělané položky jsou opětovně odhadnuty na náročnost vytvoření (protože práce na nich vykonaná se rychle znehodnocuje) a umístěny nazpět do Product Backlog. Následně se musí Scrum tým rychle přeskupit a naplánovat nový Sprint. [18]

3.1.4.2 *Sprint Planning*

Sprint Planning je schůze celého Scrum týmu, na které se členové týmu domlouvají, na jaké práci budou následující Sprint pracovat. [18]

Maximální doba trvání Sprint Planning je v případě jednoměsíčního Sprintu 8 hodin. V případě kratších Sprintů tato schůze obvykle trvá kratší dobu. Scrum Master zajišťuje, aby se schůze uskutečnila a nepřesáhla vymezený čas. Zároveň je na Scrum Master, aby všichni účastníci rozuměli jejímu účelu. [18]

Sprint Planning se obecně dělí do dvou tematických částí. V první části, často v praxi také nazývané SP1 [10], se řeší odpověď na otázku „Co může být vytvořeno během nadcházejícího Sprintu?“ [18] a ve druhé části, také nazývané SP2 [10], se řeší otázka „Jak bude naplánovaná práce provedena?“ [18]

SP1 – „Co může být vytvořeno během nadcházejícího Sprintu?“

SP1 se dá ve skutečnosti považovat za detailní workshop na požadavky. Product Owner představí sadu funkcí, které jsou umístěny na vrchu Product Backlog, tzn., že jsou prioritní. Členové vývojářského týmu pokládají vyčerpávající dotazy, aby předloženým požadavkům dostatečně důkladně porozuměli, a byli schopni předpovědět, co budou schopni během následujícího Sprintu dodat. Vývojářský tým si sám určí, co je dosažitelné vzhledem k délce trvání Sprintu, velikosti a kompetentnosti týmu, definici „Hotovo“, státním svátkům, dovoleným a podobným přerušením prací, a k tomu nejdůležitějšímu – ke všem zlepšením, kterým se tým zavázal na poslední schůzi Sprint Retrospective. [10]

Potom, co vývojářský tým předurčí, které položky z Product Backlog bude schopen doručit, Scrum tým definuje Cíl Sprintu, tzv. „Sprint Goal“. Cíl Sprintu je záměr, který je naplňován v průběhu Sprintu prostřednictvím realizace položek z Product Backlog. Zároveň slouží vývojářskému týmu jako vodítko k tomu, proč pracuje na daném Inkrementu. [18]

SP2 – „Jak bude naplánovaná práce provedena?“

Potom, co je zvolen Cíl Sprintu a vybrány položky z Product Backlog, vývojářský tým se rozhodne, jak ze zvolené práce vytvoří během Sprintu „Hotový“ Inkrement. Vybrané položky pro daný Sprint z Product Backlog a plán pro jejich dodání jsou nazývány Sprint Backlog. [18]

Vývojářský tým obvykle začne návrhem systému a pracemi potřebnými k přeměně položek z Product Backlog do fungujícího Inkrementu produktu. Charakter prací se může během Sprintu změnit, stejně tak jako i odhad pracnosti. Nicméně, během Sprint Planning je naplánováno dostatek práce, aby vývojářský tým mohl poměrně přesně odhadnout, kolik práce bude moci během Sprintu splnit. Práce naplánované na první dny Sprintu jsou na konci schůze rozloženy do pracovních jednotek, často o délce jednoho dne a méně. [18]

Obrázek 2 – Sprint planning



Copyright © 2012, Kenneth S. Rubin and Innovation, LLC. All Rights Reserved.

Zdroj: [26]

Konkrétními výstupy Sprint Planning jsou [1]:

- Cíl Sprintu,
- seznam členů týmu a jejich úroveň závazku (pokud není 100 %),
- Sprint Backlog,
- délka trvání Sprintu, resp. datum jeho ukončení,
- stanovení času a místa pro porady Daily Scrum.

3.1.4.3 Daily Scrum

Daily Scrum, rovněž nazývaný Stand-Up meeting, je každodenní porada vývojářského týmu, která má maximální dobu trvání 15 minut [18]. Je jedním z primárních bodů pro kontrolu a přizpůsobení ve Scrum [10]. Slouží k synchronizování aktivit v týmu a vytvoření pracovního plánu na dalších 24 hodin. Stand-Up meetingy se pořádají každý den ve stejný čas na stejném místě, aby se snížila komplexita jejich pořádání. [18] Ideálním případem je porada Daily Scrum pořádána v ranních hodinách, resp. na začátku pracovní doby. [22]

Na poradě se probírá, co bylo od minulého Stand-Up meetingu uděláno a co se udělá v daný pracovní den – každý vývojář ve stručnosti sdělí týmu, na čem pracoval minulý den a na čem bude pracovat nyní, aby pomohl vývojářskému týmu splnit Cíl Sprintu, a případně i skutečnost, která by mu bránila v provedení naplánované práce. [18]

Daily Scrum není událost určená pro podávání reportingu Scrum Master ani nikomu ostatnímu. Ve skutečnosti příručka Scrum říká, že Scrum Master by měl zajistit, aby se Stand-Up meetingů nezúčastňoval nikdo jiný kromě členů vývojářského týmu. [10]

Obrázek 3 - Daily Scrum



Zdroj: [26]

Porady Daily Scrum zlepšují komunikaci v týmu, eliminují potřebu dalších schůzí, identifikují překážky ve vývoji produktu nutné k odstranění, vyzdvihují a podporují rychlé rozhodování a zlepšují úroveň znalostí všech vývojářů o projektu. [18]

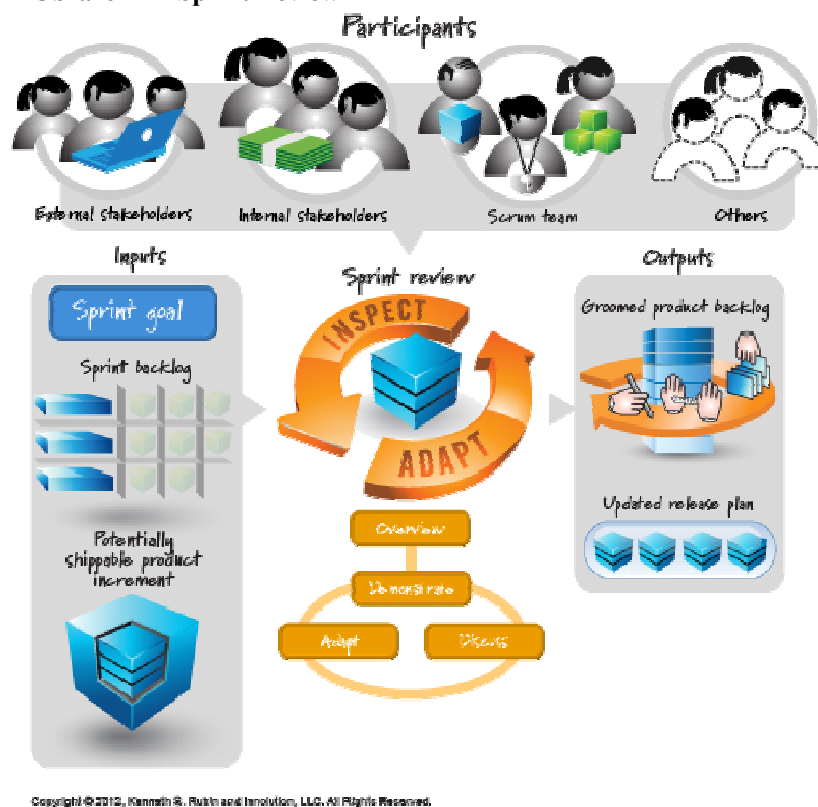
3.1.4.4 *Sprint Review*

Sprint Review je neformální schůze pořádaná po ukončení Sprintu, na které se prezentuje dodaný Inkrement. Zaměřuje se na produkt, který vývojářský tým vytváří [10]. Kontroluje se dodaný Inkrement a případné přizpůsobení Product Backlog, které nastalo během Sprintu. V průběhu Sprint Review Scrum tým a zúčastněné strany projednávají odvedenou práci během posledního Sprintu. V závislosti na výsledcích Sprintu a případných změnách v Product Backlog během něj, účastníci schůze probírají budoucí vývoj produktu a jeho přidané hodnoty, jinými slovy Sprint Review má vyvolat zpětnou vazbu k produktu a podpořit spolupráci. [18]

Maximální doba trvání Sprint Review pro jednoměsíční Sprinty jsou čtyři hodiny, v případě kratších Sprintů je schůze kratší. Je na Scrum Master, aby zajistil, aby se událost uskutečnila a aby účastníci rozuměli jejímu účelu. Scrum Master také hlídá dodržení maximální délky trvání schůze. [18]

Sprint Review se zúčastňuje celý Scrum tým a zúčastněné strany, které na událost pozval Product Owner. Product Owner shrne, které položky z Product Backlog dosáhly statusu „Hotovo“ a které položky nestihly být dokončeny. Je projednáno, jak si Product Backlog celkově stojí a na tomto základě Product Owner odvodí pravděpodobné data ukončení vývoje produktu k danému dni. [18]

Obrázek 4 – Sprint Review



Zdroj: [26]

V dalším bodě schůze vývojářský tým prodiskutuje, co se během Sprintu dařilo, jaké se vyskytly problémy a jak tyto problémy byly vyřešeny. Následně demonstruje „Hotovou“ práci a zodpoví otázky k dodanému Inkrementu [18]. Je důležité, aby položky, které nestihly být během Sprintu dokončeny, nebyly představeny, jinak by se mohlo stát, že jsou také „Hotové“. [19]

Celá skupina následně společně navrhne, co bude řešeno dále. Sprint Review tedy poskytuje cenný vstup do následného Sprint Planning. Skupina také zhodnotí aktuální situaci trhu vytvářeného výrobku a jak se potenciální užití produktu mohlo během vývoje změnit. Dále určí, která z nejdůležitějších vlastností produktu by se měla vyvíjet jako další. [18]

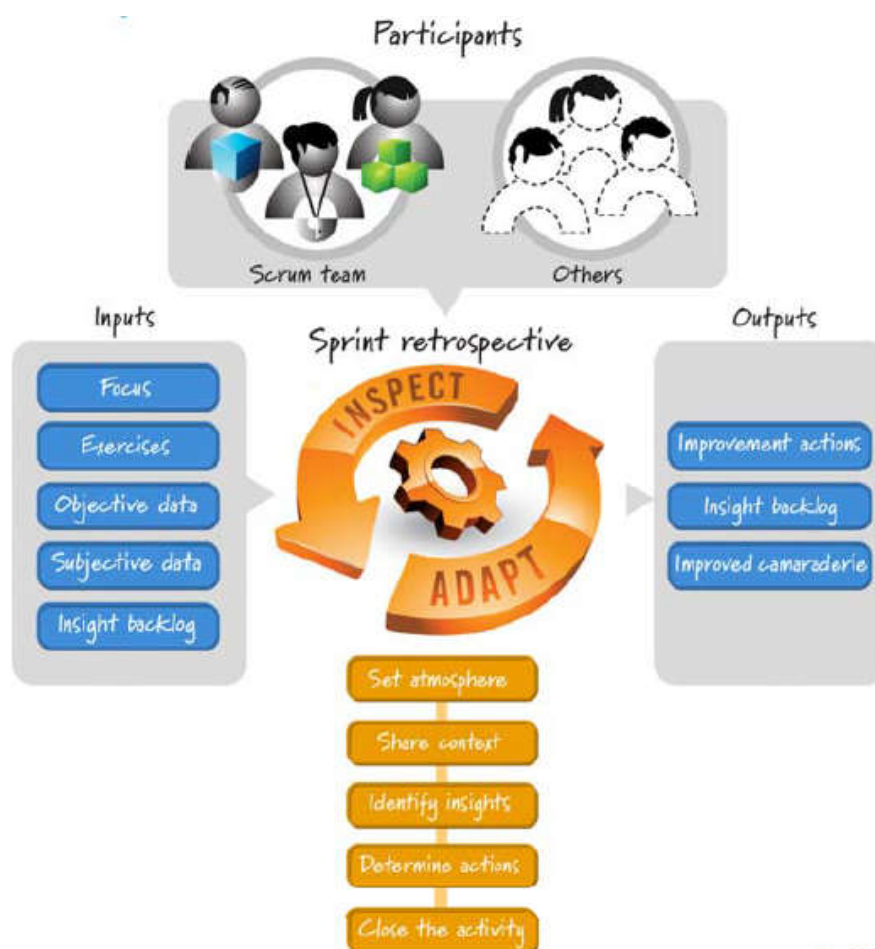
Posledním bodem schůze Sprint Review je zhodnocení časového průběhu, rozpočtu, potenciální funkce produktu a předpokládaná situace na trhu při dodání dalšího Inkrementu. [18]

Výstupem ze Sprint Review je revidovaný Product Backlog s vytipovanými položkami pro eventuální zařazení do následujícího Sprintu. [18]

3.1.4.5 *Sprint Retrospective*

Sprint Retrospective je poslední událostí Sprintu, následuje ihned po skončení Sprint Review. Zatímco Sprint Review je zaměřen na produkt, Sprint Retrospective je zaměřen na proces – na způsob, jakým Scrum tým spolupracuje, včetně jejich technických dovedností, praktik jejich softwarového vývoje a nástrojů, kterých využívají [10]. Je to příležitost pro Scrum tým ke kontrole sebe sama a pro vytvoření plánu na zlepšení, který se použije při průběhu dalšího Sprintu. [18]

Obrázek 5 - Sprint Retrospective



Zdroj: [26]

Dalším rozdílem mezi Sprint Review a Sprint Retrospective je uzavřenost schůze pro všechny až na Scrum tým – schůze se výhradně účastní jen Product Owner, Scrum Master a vývojářský tým. [10]

Sprint Retrospective má maximální dobu trvání tři hodiny pro jednoměsíční Sprints. V případě kratšího Sprintu je schůze obvykle kratší. Scrum Master zajišťuje konání schůze, je zodpovědný, že účastníci znají její účel, a že je dodržena maximální doba konání. Na rozdíl od všech předchozích událostí se Scrum Master schůze účastní jako řadový člen týmu, který má pouze na odpovědnost proces Scrum. [18]

Scrum Master se snaží, v rámci procesu Scrum, podpořit ostatní členy Scrum týmu ve zlepšování vývojového procesu a postupů za účelem efektivnějšího a příjemnějšího průběhu následujícího Sprintu. [18]

Další součástí Sprint Retrospective je debata nad tím, jak vhodnou adaptací definice „Hotovo“ zvýšit výslednou kvalitu produktu. [18]

Výstupem Sprint Retrospective by měla být identifikovaná zlepšení, která budou zavedena v následném Sprintu. Implementace těchto zlepšení v dalším Sprintu je v podstatě adaptace plynoucí z kontroly, kterou Scrum tým provedl sám na sobě. I když zlepšení můžou být zaváděna kdykoliv v průběhu procesu, Sprint Retrospective poskytuje formální příležitost k zaměření se na kontrolu a adaptaci. [10]

3.1.4.6 Backlog Grooming

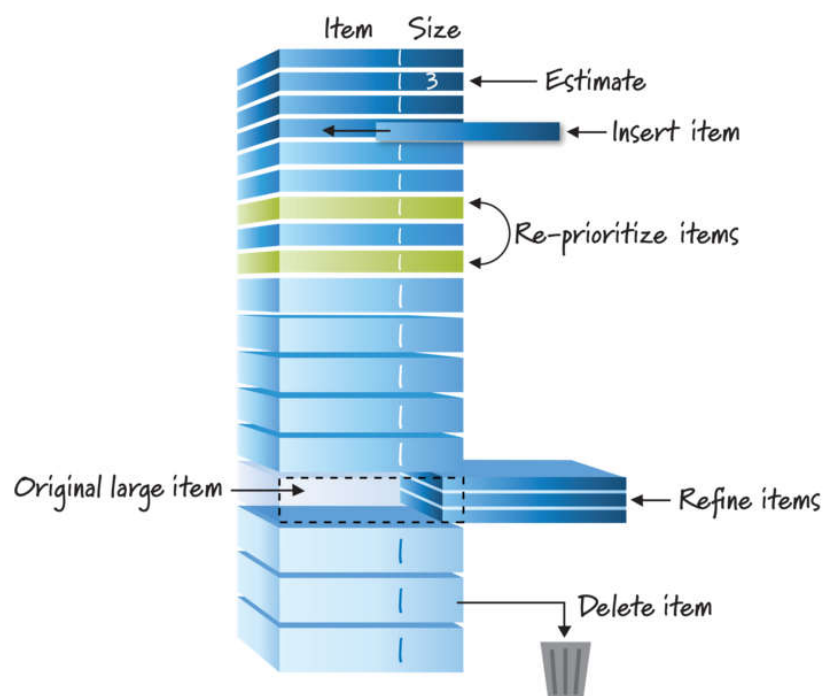
Backlog Grooming je neformální procedura v procesu Scrum, která není přímo definována Scrum. Jedná se o přípravu Product Backlog na schůzi Sprint Planning, které se Product Owner a vývojářský tým věnuje z 5 – 10 % své kapacity. Je doporučováno, aby se jí tým během Sprintu věnoval v jednom z dnů v posledním týdnu před skončením Sprintu. [8]

Při Backlog Grooming se z Product Backlogu odstraňují User Stories, které se již nezdají relevantní a naopak, v závislosti na nové potřeby, se vytvářejí User Stories nové. Pro nově vzniklé User Stories se dělají odhady, znovu se posuzují priority jednotlivých User Stories a také ve světle nových informací se opravují odhady starých User Stories.

Backlog Grooming rovněž slouží k rozdělení Epics s vysokou prioritou, které by nebylo možné zpracovat v jedné iteraci, na menší celky. [9]

Upravený Product Backlog činí schůze Sprint Planning mnohem efektivnější, v opačném případě se tyto porady mohou protáhnout na několik hodin. Pokud položky Product Backlog mají jasně stanovená akceptační kritéria a jsou odhadnuty příslušnými členy Scrum týmu, proces plánování dalšího Sprintu nemusí být napjatý a zbytečně dlouhý. [8]

Obrázek 6 – Činnosti aktivity Grooming



Copyright © 2012, Kenneth S. Rubin and Innolution, LLC. All Rights Reserved.

Zdroj: [26]

3.1.5 Artefakty Scrum

Scrum definuje tři artefakty [18]:

1. Product Backlog,
2. Sprint Backlog,
3. Inkrement.

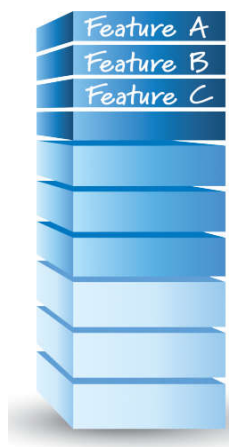
Artefakty reprezentují práci nebo její hodnoty. Slouží k zajištění transparentnosti a jsou v procesním rámci Scrum místem pro kontrolu a přizpůsobení. Artefakty definované Scrum jsou speciálně navrženy tak, aby se maximalizovala transparentnost klíčových informací pro jednoznačné porozumění daných Artefaktů. [18]

3.1.5.1 *Product Backlog*

Product Backlog je seznam pracovních položek, popsanych na funkční úrovni, které jsou zapotřebí v průběhu času udělat [10]. Je jediným zdrojem požadavků na jakékoliv požadované změny produktu [18].

Product Backlog je dynamický a nikdy není kompletní. Neustále se mění tak, aby obsahoval vše, co produkt potřebuje – aby byl konkurenceschopný, užitečný a patřičný ke svému zadání. Je to seznam všech vlastností, funkcí, požadavků, rozšíření a oprav chyb, které představují všechny změny, které budou provedeny v produktu v budoucích iteracích. Položky Product Backlog mají popis, prioritu a odhad. [18]

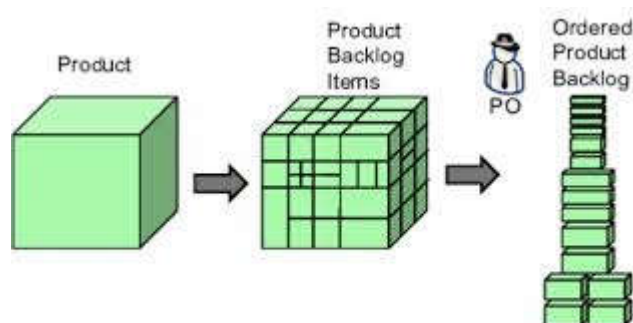
Obrázek 7 – Product Backlog



Zdroj: [26]

Za Product Backlog je zodpovědný Product Owner, včetně jeho obsahu, platnosti a seřazení [18]. Položky do Product Backlog může přidat kdokoli z týmu, je však na Product Owner, aby určil pořadí, v jakém budou provedeny [10]. Pořadí položek v seznamu je určeno podle priority, kterou ji přiřadí Product Owner. Na vrchu seznamu jsou položky s nejvyšší prioritou, dole naopak s nejnižší [20]. Podobně tomu je i s detailností zadání položek – jasnost zadání a míra detailu postupně klesá od vrchu seznamu [18].

Obrázek 8 – Proces sestavení a seřazení Product Backlog



Zdroj: [13]

Správa Product Backlog zahrnuje zpřesňování detailů, odhadů pracnosti a uspořádání jeho jednotlivých položek. Je to neustálá aktivita, která probíhá paralelně se Sprintem. Spousty položek jsou přidávány postupem času; existující položky jsou rozčleněny do několika dalších, menších položek; některé položky jsou odebrány, protože jejich funkce již nejsou potřeba. Navíc některé položky musí být předimenzovány tak, aby bylo možné určit pravděpodobný vztah mezi hodnotou, časem a náklady. [10]

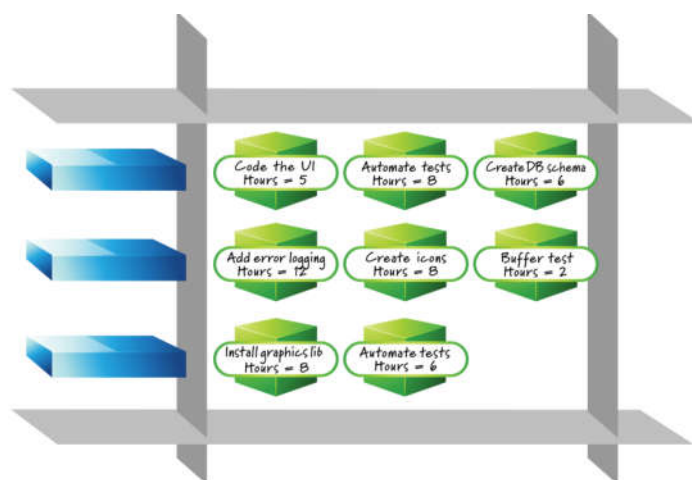
Na upřesnění jednotlivých položek spolupracuje Product Owner společně s vývojářským týmem, kterému to obvykle nezabere více jak 10 % jeho kapacity. Zatímco za obsah, platnost a seřazení Product Backlog je odpovědný Product Owner, viz výše v textu, za všechny finální odhady položek je odpovědný vývojářský tým. [18]

Často na vývoji jednoho produktu pracuje více Scrum týmů. V tomto případě se používá pouze jeden Product Backlog, jehož položky jsou seskupeny podle určitých atributů. [18]

3.1.5.2 Sprint Backlog

Sprint Backlog je složen ze sady položek vybraných pro daný Sprint z Product Backlog a plánech pro dodání Inkrementu produktu a realizování Cíle Sprintu. Ze Sprint Backlog se dá vyčíst všechna práce, kterou si vývojářský tým vytýčil jako nezbytnou pro splnění Cíle Sprintu (viz Obrázek 9), jaká funkcionality bude dodána v následujícím Sprintu a kolik práce bude zapotřebí k dodání této funkcionality. [18]

Obrázek 9 – Sprint Backlog s konkrétními úkoly



Zdroj: [26]

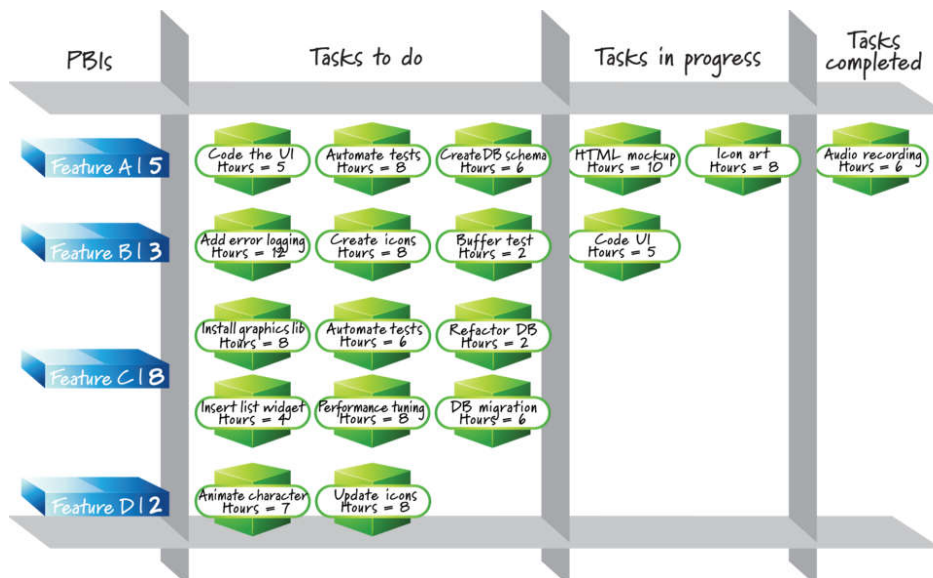
Vývojářský tým upravuje Sprint Backlog během Sprintu. Úpravy Sprint Backlog jsou nutné z důvodu neustálého získávání nových poznatků během Sprintu vývojářským týmem. Sprint Backlog je natolik detailní, že případné jeho změny mohou být projednávány na poradách Daily Scrum. [18]

Pouze vývojářský tým může měnit Sprint Backlog v průběhu Sprintu. V případě, že je požadována nová práce, tým ji přidá. Pokud se některé úkoly ukáží v průběhu Sprintu jako zbytečné, jsou ze Sprint Backlog odstraněny. [18]

Většina vývojářských týmů zobrazuje svůj Sprint Backlog na Task Board (tabuli s úkoly), která je fyzickým zobrazením seznamu práce, kterou se vývojářský tým sjednal udělat během daného Sprintu. [10]

Task Board byl pro tento účel přejat z agilního přístupu Kanban [10]. Je na něm transparentně vidět, v jaké fázi se položky ze Sprint Backlog nacházejí, viz Obrázek 10.

Obrázek 10 – Ukázka Task Board

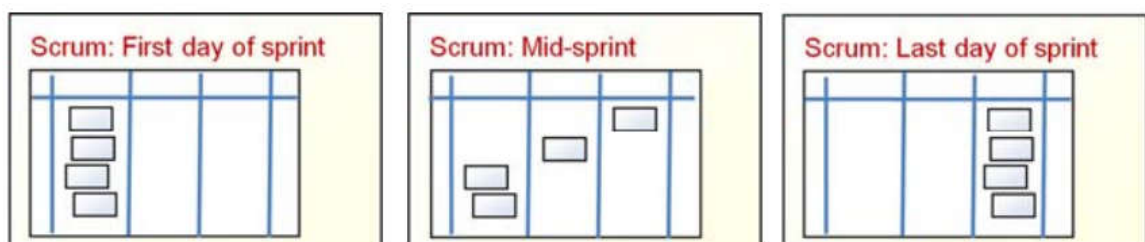


Zdroj: [26]

Na Task Board je vývoj Inkrementu rozdělen do tří fází – „To Do“ („Co je potřeba udělat“), „Ongoing“ („Na čem se pracuje“) a „Done“ („Hotovo“). [2]

Na začátku Sprintu jsou všechny naplánované úkoly zařazeny do sloupce fáze „To Do“, viz Obrázek 11, vlevo. Během Sprintu se stavy úkolů posunují po tabuli podle toho, v jaké fázi se úkoly zrovna nacházejí. Pokud se na úkolu zatím nepracuje, zůstává ve sloupci „To Do“. Pokud se na úkolu začalo pracovat, je přesunut do fáze „Ongoing“. Když úkol splní všechny náležitosti stanovené definice „Hotovo“, je přesunut do sloupce „Done“; viz Obrázek 11, uprostřed. Na konci Sprintu by všechny úkoly měly být ve fázi „Done“; viz Obrázek 11, vpravo. V případě, kdyby vývojářský tým nestihl dokončit naplánovaný úkol, tento úkol se vrátí nazpět do Product Backlog a bude se na něm pracovat v následujících Sprintech. [2]

Obrázek 11 – Scrum Board na začátku, v průběhu a na konci Sprintu

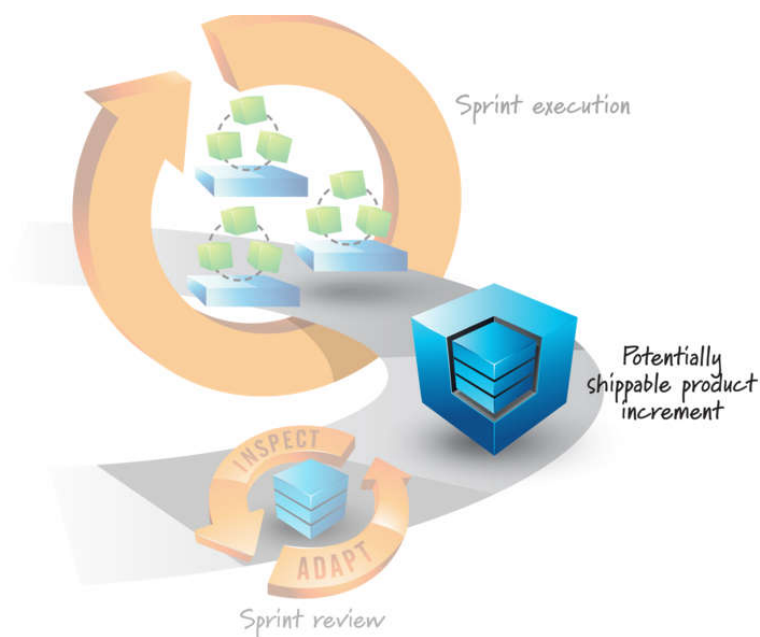


Zdroj: [2]

3.1.5.3 Inkrement

Inkrement (Obrázek 12) je suma všech Product Backlog položek dokončených během posledního Sprintu a hodnota všech Inkrementů z předchozích Sprintů. Na konci Sprintu vytvořený Inkrement musí být použitelný a splňovat týmovou definici „Hotovo“, a to bez ohledu na to, zda se ho Product Owner rozhodne vydat nebo ne. [18]

Obrázek 12 – Inkrement Produktu



Copyright © 2012, Kenneth S. Rubin and Innolution, LLC. All Rights Reserved.

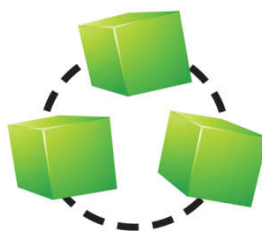
Zdroj: [26]

3.1.6 Úkoly, User Stories a Epics

3.1.6.1 Úkol

Úkol je technická práce, kterou provádí vývojářský tým s cílem vyhotovit položku Product Backlog. Většina úkolů je definována v malém rozsahu, tak aby představovaly ne více jak několik hodin práce denně. [21]

Obrázek 13 – Znázornění úkolů



Zdroj: [26]

3.1.6.2 User Story

User Story je nástroj využívaný v agilním vývoji softwaru, který zachycuje popis funkcí softwaru z pohledu koncového uživatele. Slouží k vytvoření zjednodušeného popisu požadavku. User Story popisuje formou krátkého příběhu druh uživatele, co chce a proč. Délka agilní User Story by měla být krátká, měla by se vejít na kartičku. User Stories by měly být napsány jazykem zákazníka, aby bylo jasné jak zákazníkovi, tak i vývojářskému týmu, co zákazník požaduje a proč to požaduje. Úkolem vývojářského týmu je postarat se o to, jak vytvořit kód, který bude splňovat požadavky User Story. [27]

Formou User Stories jsou většinou psány záznamy v Product Backlog. Jejich výhodou je, že se zaměří přesně na to, co uživatel potřebuje / chce, aniž by zacházely do detailů o tom, jak toho dosáhnout. [16]

Příklad User Story - obecně:

Jako [osoba], [chci|musím] [akce] takže dosáhnu [dosažení].

[Osoba]: „Vlastník“ User Story. Bývá jí často uživatel, ale doporučuje se být v popisu více specifický. Při použití konkrétní osoby (například „Správce“, „Přihlášený zákazník“, „Neregistrovaný zákazník“) je snazší pochopit a stanovit kontext User Story. [16]

[Akce]: Co Osoba chce udělat. Pokud se jedná o povinnou Akci, může se použít termín „musím“. V opačném případě je používán termín "chci". [16]

[Dosažení]: Čeho chce Osoba dosáhnout provedením Akce. Z pohledu Osoby je to výsledek provedené Akce. [16]

Příklad User Story - konkrétně:

Jako [Uživatel], chci mít možnost [přihlásit se] takže [budu mít přístup do chráněné části systému].

3.1.6.3 Epics

Epics jsou User Stories, které jsou natolik rozsáhlé, že aby se daly realizovat v jedné iteraci. K jejich realizaci se tudíž musí nejdříve rozdělit do menších User Stories. [24]

Epics mívají v Product Backlog většinou menší prioritu a jsou tedy umístěny níže. Jakmile se dostanou k vrcholu Product Backlog, jsou reorganizovány do menších User Stories a začne se na nich pracovat. [24]

4 Vlastní práce

Vlastní práce se zaměřuje na současný stav řízení IT zakázek firmy Juicyo s.r.o. pomocí agilního přístupu Scrum.

4.1 Popis vybrané firmy

4.1.1 Základní údaje

Název firmy:	Juicyo s.r.o.
Datum založení:	30. ledna 2013
Sídlo firmy:	Jugoslávských partyzánů 1580/3, Praha 6 - 16000
Majitelé a jednatele:	Ing. Tomáš Jukin - podíl: 50 % Ing. Jana Moudrá - podíl: 50 % [25]

Hlavní činnost firmy:

- tvorba mobilních aplikací pro platformy iOS a android,
- tvorba webových aplikací,
- tvorba webových backendů.

4.1.2 Historie a současnost firmy

Juicyo, s.r.o. je rozvíjející se firma, která na trhu mobilních aplikací figuruje od roku 2013.

Hlavním důvodem k založení Juicyo s.r.o. byla nespokojenost jeho zakladatelů, Ing. Jukina a Ing. Moudré, s dosavadním vývojem mobilních aplikací – vývojem, kdy mobilní aplikace nutí lidi, aby se jim přizpůsobovali. Juicyo s.r.o. by chtělo tento světový trend změnit. Jeho hlavní mottem je vytvářet takové mobilní aplikace, které se budou přizpůsobovat lidem a ne naopak. Takovým aplikacím firma přezdívá „šťavnaté“, v překladu „juicy“. Odtud je odvozen název firmy *Juicyo*.

Firma si je dobře vědoma toho, že pouze jedna malá společnost není schopna změnit směr, kterým se dosavadní trh mobilních aplikací ubírá, proto v rámci své divize Juicyo Academy se snaží pomocí konzultantské činnosti předávat tuto myšlenku širší veřejnosti. Kromě konzultantské činnosti majitelé společnosti rovněž přednáší na vysokých školách, kde své vědomosti a zkušenosti předávají novým generacím IT techniků.

Juicyo s.r.o. bylo založeno 30. ledna 2013 Ing. Tomášem Jukinem a Ing. Janou Moudrou, přičemž každý z nich vlastní 50 % podíl [25]. Na plánování strategie firmy se podílí ve vzájemném konsenzu. První rok byla firma vedena pouze jejími zakladateli, v polovině roku 2014 se firma rozrostla na 8 lidí, 2 majitele a 6 zaměstnanců.

Juicyo s.r.o. po vzoru firmy *Base Camp* (dříve *37 Signals*) zastává myšlenku „*Stay small, do big!*“, v překladu: „*Zůstaň malý, dělej velké věci!*“. Principem tohoto myšlení je zaměřování se na produkt, na jeho kvalitu, ne na vyrobenou kvantitu. Toto se odráží i na současné velikosti firmy - firma preferuje malý tým s kvalitními a zároveň těmi nejlépe kvalifikovanými pracovníky. Juicyo s.r.o. nemá vizi, že se z ní stane společnost, která bude mít 100 zaměstnanců.

Juicyo s.r.o. v současné době je zaměřené na vývoj mobilních aplikací na zakázku. To by ale chtělo v budoucnosti změnit. Chtělo by vytvářet své vlastní produkty, které budou rozvíjet, vyvíjet a udržovat. Zaměřit se na tzv. horizontální škálování, kdy firma udržuje stejný počet zaměstnanců a přitom vytváří zisk na jimi vytvořených aplikacích, o které se stará. Tento transfer není jednoduchý. Záleží na tom, jak se aplikace ujme veřejností a provádí se tím lépe, čím je firma menší.

4.1.3 Organizační a řídicí struktura

Organizační struktura

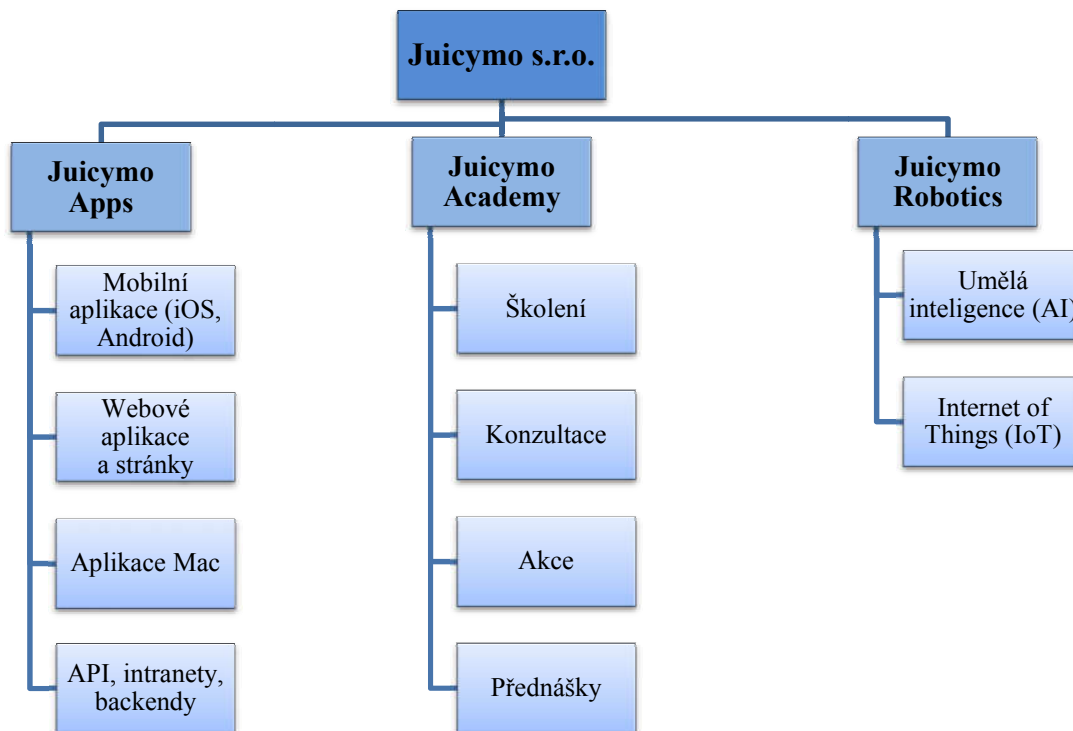
Juicymo s.r.o. se dělí do tří divizí. Největší divizí firmy jsou tzv. *Juicymo Apps*. Tuto divizi vede Ing. Tomáš Jukin. *JuicymoApps* tvoří největší část obrátu, a to až 90 %, z něhož 80 % tvoří zahraniční klientela. Hlavní náplní této divize je tvoření aplikací na zakázku. Vedlejší náplní je tvorba tří vlastních produktů, aplikací, které jsou určené pro koncové uživatele – pro B2C a B2B.

Druhou divizí je oddělení výzkumu a vývoje, *Juicymo Robotic R&D*. *Juicymo Robotic R&D* vede rovněž Ing. Tomáš Jukin. Divize se zabývá zkoumáním nových technologií se zaměřením na umělou inteligenci (AI) a jejího nasazení na řízení robotů a expertních systémů, případně evolučního programování, a Internet of Things (IoT), v překladu „internet věcí“, a s tím společný Open Hardware.

Třetí a nejmladší divizí je *Juicymo Academy*. *Juicymo Academy* vznikla začátkem roku 2015. Divize je vedena Oldřichem Novákem a zabývá se školením, konzultacemi, pořádáním akcí, přednášek atd.

Znázornění organizační struktury Juicymo s.r.o. je znázorněné v diagramu 1.

Diagram 1 – Organizační struktura Juicyo s.r.o.

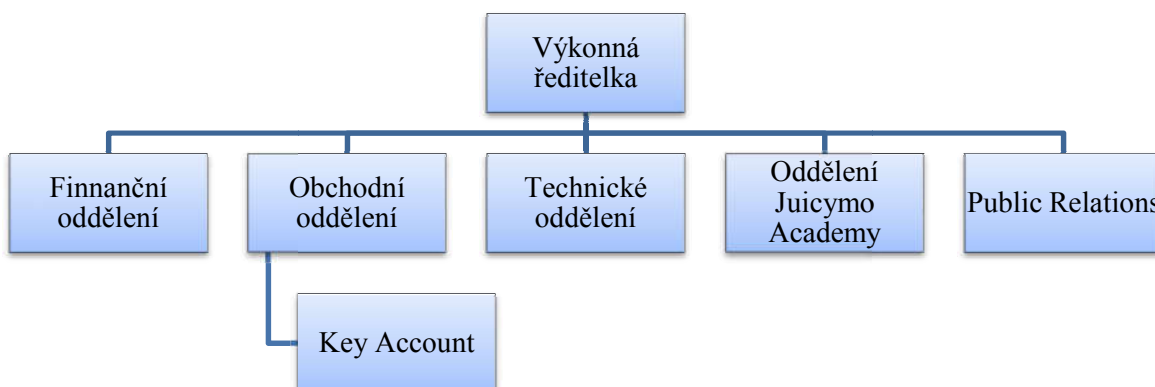


Zdroj: Vlastní zpracování

Řídící struktura

Na vrcholu společnosti Juicyo s.r.o. stojí výkonný ředitel (CEO), pod kterého spadá šest oddělení: finanční (CFO), obchodní (Sales), marketingové, technické (CTO), oddělení řídicí Juicyo Academy a public relations. Viz Diagram 2.

Diagram 2 – Řídící struktura Juicyo s.r.o.

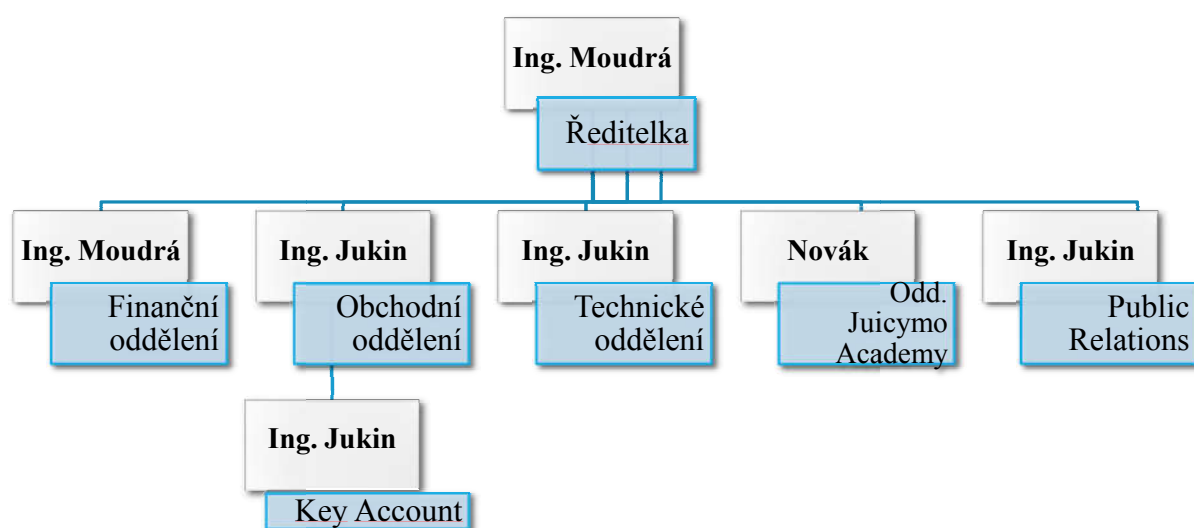


Zdroj: Vlastní zpracování

V následujícím

Diagram 3 je možno vidět, kdo vykonává vedoucí pozici každého oddělení. Výkonnou ředitelkou firmy Juicyo s.r.o. je Ing. Jana Moudrá. Ing. Moudrá ještě zastává pozici ředitelky finančního oddělení. O obchodní, marketingové, technické a PR oddělení se stará Ing. Tomáš Jukin. Vedoucí pozici oddělení Juicyo Academy zastává Oldřich Novák.

Diagram 3 – Obsazení manažerských rolí Juicyo s.r.o.



Zdroj: Vlastní zpracování

4.2 Současný stav řízení IT zakázek v Juicyo s.r.o.

4.2.1 Agilní přístup řízení IT zakázek v Juicyo s.r.o.

Když majitelé Ing. Jukin a Ing. Moudrá v roce 2013 zakládali firmu Juicyo s.r.o., veškerý vývoj aplikací dělali svépomocí. S uřízením zakázek měli minimum problémů. Jakmile však v druhé polovině roku 2014 přijali do firmy šest dalších vývojářů a rozšířili své portfolio o další produkty a služby, začalo být zřejmé, že ve svém dosavadním stylu řízení zakázek nemohou pokračovat. Začali se zajímat, jaký typ řízení by se nejvíce hodil pro vývojářské prostředí. Řešení jejich dilema bylo poměrně jednoduché – *Scrum*.

Agilní přístup Scrum firmě vyhovuje hned z několika důvodů. Klíčovým faktorem pro jeho zvolení byl samozřejmě obor informačních technologií zaměřený na vývoj a udržování softwarových produktů, ve kterém firma podniká. Majitelům se líbilo, že ač je Scrum jednoduchý a srozumitelný, dají se pomocí něj řešit složité adaptivní problémy, zatímco produktivně a tvůrčím způsobem dodávat produkty s vysokou přidanou hodnotou.

Dalším příznivým faktorem pro implementaci Scrum byl nízký počet zaměstnanců firmy Juicyo s.r.o. Společnost má devět pracovníků, což je pro vytváření malých Scrum týmů ideální počet. V projektovém týmu Juicyo s.r.o. rozeznává pět rolí: Product Owner, Scrum Master, Team Member, Watcher a Stakeholder. Více o týmových rolích pojednává kapitola 4.3.1.

Majitelé Juicyo s.r.o. si již od náborů zaměstnanců v roce 2014 velmi zakládají na dobrých vztazích mezi zaměstnanci a přívětivé firemní kultuře. Té dosahují osobním přístupem, dobrými a rovnocennými vztahy a hlavně důvěrou ve své zaměstnance, což zároveň vytváří dobrý základ pro efektivní fungování Scrum. Z tohoto pohledu se dá říci, že výběr agilního přístupu Scrum byl pro firmu a její firemní kulturu velmi výhodný, protože díky Scrum Juicyo s.r.o. nemusí řešit zaměstnaneckou hierarchii, při tvorbě projektů mají všichni firemní pracovníci stejnou výchozí pozici, což se kladně podepisuje na přetrvávajících dobrých zaměstnaneckých vztazích.

Jak již bylo zmíněno, Scrum byl ve společnosti Juicyo s.r.o. implementován v roce 2014. Od té doby je ve firmě úspěšně provozován. Majitelé berou metodiku Scrum jako „návod“ k řízení jejich IT projektů. Nebojí se jej rozvíjet a rozšiřovat tak, aby se co nejlépe přizpůsobil jejich potřebám.

4.2.2 Životní cyklus IT zakázek v Juicyo s.r.o.

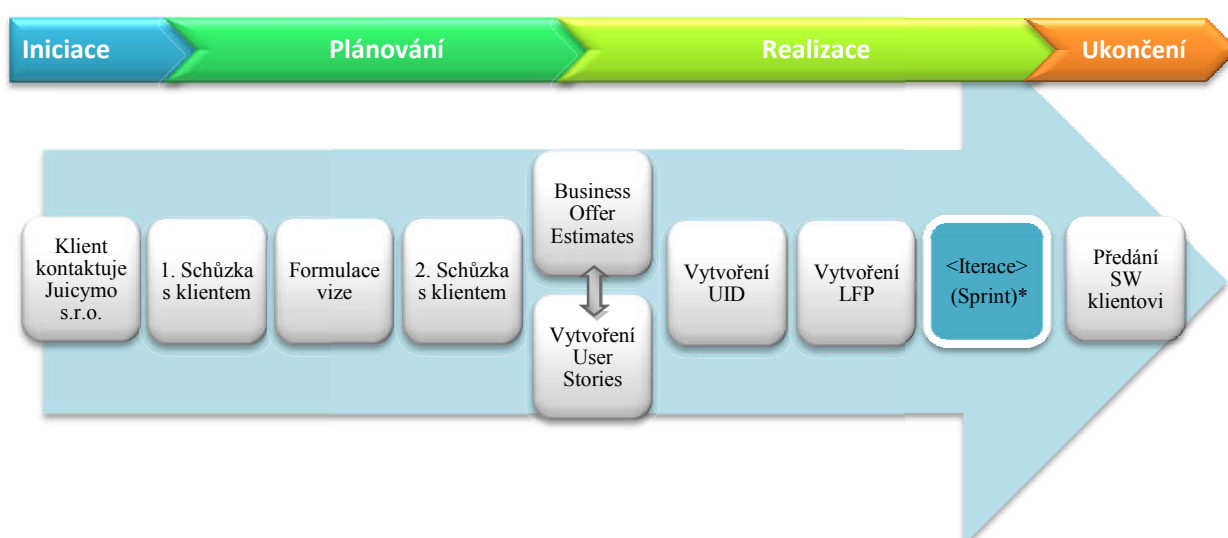
Životní cyklus řízení IT zakázek ve firmě Juicyo s.r.o., viz Diagram 4, je započat v okamžiku, kdy klient kontaktuje Juicyo s.r.o. a sdělí firmě svůj „nápad“.

Následuje schůzka zástupce firmy Juicyo s.r.o. s klientem. Na této schůzce je klientovi představena společnost. Klient sdělí svůj nápad a požadavky jednatelemi firmy. Domluví se spolu, co by chtěl klient dodat, jakým způsobem a proč.

V další fázi životního cyklu IT zakázky je zformulována vize navrhnutého produktu. Jak klient, tak i zástupce Juicyo s.r.o. sepíší nezávisle na sobě tzv. Lean Canvas, tj. „Business plán“ pro jeden produkt, jehož obsah je rozsahem na cca stranu A4. Důvodem tohoto kroku je, že na další schůzce klienta s jednatelem Juicyo s.r.o. se oba Lean Canvas navzájem porovnají. Pokud komunikace na první schůzce proběhla v pořádku a bez šumů, Lean Canvas by se měly shodovat. Pokud se Lean Canvas od sebe významně liší, klient a jednatel Juicyo s.r.o. si neporozuměli, musí si vše objasnit a tuto fázi zopakovat.

Po druhé schůzce s klientem nastává fáze v Juicyo s.r.o. nazývaná „Business Offer Estimates“. Jednatel Juicyo s.r.o. na základě informací získaných od klienta vytvoří User Stories, které nechá týmem ocenit Story body. Na základě Story bodů následně odhadne cenu zakázky, kterou předloží klientovi. Ten danou cenu buďto akceptuje či neakceptuje. Až do tohoto okamžiku jsou služby Juicyo s.r.o. neplacené.

Diagram 4 – Životní cyklus IT zakázek v Juicyo s.r.o.



Zdroj: Vlastní zpracování

Pokud klient cenu akceptoval, je vytvořen Scrum tým a nastává fáze realizace. Product Owner si připraví Product Backlog, který dá ocenit vývojářskému týmu.

V dalším kroku je vytvořen User Interaction Diagram (UID), který znázorňuje druhy interakcí v dané aplikaci. UID slouží k tomu, aby si klient lépe představil funkce aplikace.

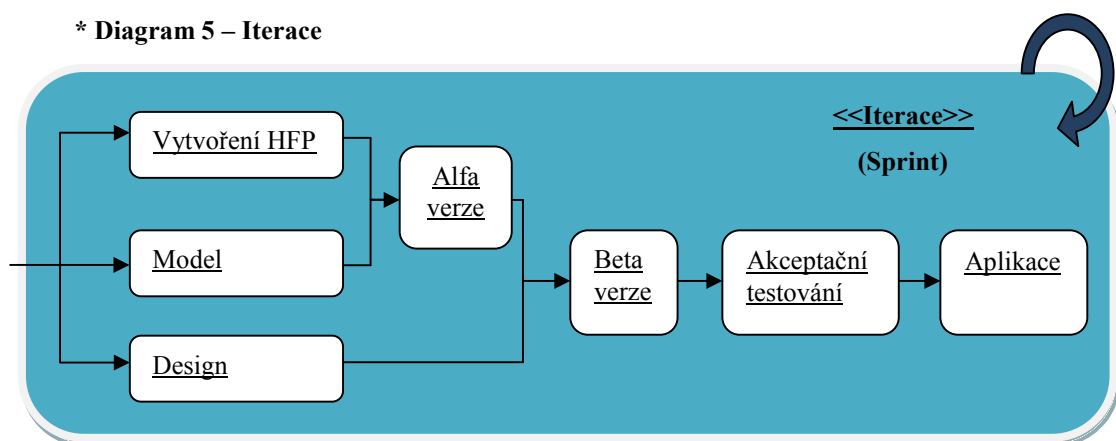
Po vytvoření UID je vytvořen Low Fidelity Prototype (LFP), aneb Prototyp nízké věrnosti. LFP je uložen v intranetu a dá se v něm v menší míře otestovat, zda navržené rozhraní pro danou aplikaci dává smysl.

Jakmile je vytvořen Low Fidelity Prototype, Scrum tým začne vyvíjet produkt v tzv. iteracích neboli Sprintech, viz * Diagram 5.

Vyvíjení produktu zahrnuje vytvoření High Fidelity Prototype (HFP), tj. Prototyp vysoké věrnosti, který slouží k testování složitějšího UI, jako jsou například animace, přímo v cílovém zařízení, a Modelu, tzv. AppLogic, tj. vzorce aplikačního chování. Z High Fidelity Prototype a Modelu vznikne Alfa verze produktu, která, když se propojí s Designem, vytvoří Beta verzi produktu. Po vytvoření Beta verze produktu, následuje akceptační testování. Pokud aplikace projde akceptačním testováním, je prohlášena za hotovou.

Poslední fází životního cyklu IT zakázky, fází ukončení projektu, je předání SW aplikace zákazníkovi.

* Diagram 5 – Iterace



Zdroj: Vlastní zpracování

4.2.3 Užívané SW nástroje na podporu řízení IT zakázek

Juicyo s.r.o. nepoužívá konkrétní nástroje v podobě metod a přístupů. Firma si vybrala a spojila různé softwarové aplikace pro zvládnutí veškeré administrace plynoucí z provozované činnosti.

Všechny nástroje, které firma využívá jako podporu řízení, jsou webové, tzn., že běží na internetu v podobě webové aplikace, tj. jsou online. Všichni členové firmy mají k těmto nástrojům přístup odkudkoliv a kdykoliv.

Jednou ze společných vlastností využívaných softwarových aplikací je, že jsou všechny Open Source. Majitelé preferují tyto Open Source aplikace z důvodu ušetření nákladů na jejich provoz, které jsou až v řádech tisíců dolarů.

Firma Juicyo s.r.o. využívá pro své fungování tyto nástroje: OpenProject, Git a GitLab Community Edition, Drone.io, Fat Free CRM, Slack, Google Drive a Google Kalendář.

4.2.3.1 Přehled používaných SW nástrojů na řízení IT zakázek

RACI matice, viz Tabulka 1, zobrazuje odpovědnosti týmových rolí vůči danému softwaru. Přehled funkcí jednotlivých softwarových nástrojů zaznamenává Tabulka 2.

Tabulka 1 – Raci matice odpovědnosti týmových rolí vůči danému softwaru

	Založení projektu v OpenProject	Založení projektu v Git	Slack	Drone.io	Fabric.io	Google Drive	Google kalendář	Fat Free CRM
Product Owner	A R	A R	A R	A	C	I	A R	R ¹
Scrum Master	C		R	C			I	
Team Members	I	I	R	R	A R	AR	C	
Watcher			C		I		I	
Stakeholder	I				I		I	

Zdroj: Vlastní zpracování

¹ RACI matice pro SW FatFreeCRM viz Tabulka 3.

Tabulka 2 – Funkce sw nástrojů ve firmě Juicymo s.r.o.

Aplikace SW Aktivity	OpenProject	Git & GitLab CE	Dronc.io	Fat Free CRM	Slack	Fabric.io	Google Drive	Google Kalendář
Projektová dokumentace	✓	✗	✗	✗	✗	✗	✗	✗
Správa Product Backlog	✓	✗	✗	✗	✗	✗	✗	✗
Správa Sprint Backlog	✓	✗	✗	✗	✗	✗	✗	✗
Zobrazení stavů User Stories v právě provozovaném Sprintu	✓	✗	✗	✗	✗	✗	✗	✗
Klientova reakce na změnu stavu projektu	✓	✗	✗	✗	✗	✗	✗	✗
Historie projektové dokumentace	✓	✗	✗	✗	✗	✗	✗	✗
Seznam a kontakty pracovníků	✓	✗	✗	✗	✗	✗	✗	✗
Seznam a kontakty pracovníků projektu	✓	✗	✗	✗	✗	✗	✗	✗
Vývoj zdrojového kódu	✗	✓	✗	✗	✗	✗	✗	✗
Automatizované testování aplikací	✗	✗	✓	✗	✗	✗	✗	✗
Marketing	✗	✗	✗	✓	✗	✗	✗	✗
Sales	✗	✗	✗	✓	✗	✗	✗	✗
Akvizice nového klienta	✗	✗	✗	✓	✗	✗	✗	✗
Zaznamenávání Leadů	✗	✗	✗	✓	✗	✗	✗	✗
Sledování vývoje oportunit	✗	✗	✗	✓	✗	✗	✗	✗
Firemní komunikace	✗	✗	✗	✗	✓	✗	✗	✗
Komunikace Scrum týmů	✗	✗	✗	✗	✓	✗	✗	✗
Distribuce betaverzí směrem ke klientovi	✗	✗	✗	✗	✗	✓	✗	✗
Správa chyb mobilních aplikací	✗	✗	✗	✗	✗	✓	✗	✗
Sdílení dokumentů	✗	✗	✗	✗	✗	✗	✓	✗
Harmonogram schůzí	✗	✗	✗	✗	✗	✗	✗	✓

Zdroj: Vlastní zpracování

OpenProject – nástroj pro management projektové dokumentace

Tabulka 2, uvedená výše, vypovídá, že OpenProject je firmě Juicymo s.r.o. primární nástrojem pro správu jednotlivých projektů, hlavně co se týče projektové dokumentace. V aplikaci je veden seznam pracovníků a jejich kontaktní informace a je v ní spravována zadávací projektová dokumentace.

Dále je v OpenProject vytvářen a spravován Product Backlog a Sprint Backlog jednotlivých zakázek. Pracovníci firmy si mohou zobrazit stavů User Stories v právě provozovaném Sprintu a zobrazit si historii projektu, či klientovu reakci na změnu stavu projektu.

Přístup do OpenProject mají všichni pracovníci Juicymo s.r.o., přičemž projekt v OpenProject zakládá a je za tento úkol zodpovědný Product Owner daného projektu. Založení projektu Product Owner může konzultovat se Scrum Master. Členové vývojářského týmu a Stakeholder jsou o založení nového projektu informováni. Watcher se na založení nového projektu do OpenProject nepodílí, ale má přístupové právo se na daný projekt kdykoli v jeho průběhu podívat.

Git & GitLab Community Edition – nástroj pro management zdrojového kódu

Git & GitLab Community Edition je čistě technická aplikace. Git & GitLab Community Edition slouží jako software pro ukládání zdrojového kódu na jednom místě. Každá změna ve vývoji aplikace, kterou programátor učiní, je v tomto SW možná vidět v reálném čase.

Za založení nového projektu v softwaru Git je zodpovědný Product Owner, který následně o této skutečnosti uvědomí členy vývojářského týmu.

Z funkce sw Git vyplývá, že tato softwarová aplikace je v průběhu projektu využívána pouze firemními vývojáři.

Drone.io

Software Drone.io je využíván jako integrační server. Slouží pro automatizované testování aplikací.

Drone.io je propojen se softwarem GitLab. Když nastane změna ve zdrojovém kódu (v softwaru GitLab), Drone.io spustí test, který danou změnu otestuje. Výsledkem testování je výstup, ze kterého je možné vyčíst stav a následky daného zásahu do zdrojového kódu.

O rozhodování ohledně výsledků testů je zodpovědný Product Owner.

CRM (Customer Relationship Management) – Fat Free CRM

Fat Free CRM je Open Source software, který firma využívá pro management vztahů se zákazníky. Software využívá zástupce Marketingového a Sales oddělení, Ing. Jukin.

Fat Free CRM je nástroj pro plánování nových zakázek. Spravuje proces od Marketingu, přes Sales, po akvizici klienta. V softwaru jsou rovněž vytvářeny a sledovány marketingové kampaně. Jednatel se do systému zaznamenává Leads a sleduje oportunitu, tj. vývoj spolupráce, resp. nespolečné spolupráce se zákazníkem; příklad uveden níže.

Příklad Leads a oportunitu:

Na firemním večírku majitel Juicyma s.r.o. dostane vizitku od potenciálního zákazníka. Tento kontakt si zaznamená do aplikace Fat Free CRM, aby ho mohl následně využít v marketingové kampani. Pokud zákazník naváže s firmou další kontakt a tento kontakt dospěje až do stádia pořádání obchodních schůzek, stává se z Lead tzv. oportunita.

Vedení a zodpovědnost za Fat Free CRM spadá pod manažery z oddělení Sales a Key Account. Na vedení dokumentace ve Fat Free CRM se rovněž podílí i Product Owners. CEO (výkonná ředitelka společnosti) může být konzultována.

Za správu klientů, osob a Leeds zodpovídá manažer Sales oddělení, který o správě informuje CEO. Správu oportunit rovněž provádí a je za ní zodpovědný manažer Sales oddělení. Manažer Sales správu oportunit konzultuje s CEO a informuje o ní oddělení Key Account.

Správu projektů vede oddělení Sales, ale odpovědnost za správu projektů spadá do kompetencí výkonné ředitelky společnosti Juicyo s.r.o. (CEO) Ing. Moudré. Oddělení Key Account a Product Owners jsou o rozhodnutích ohledně projektů informováni.

Přehled funkcí sw a oddělení / osob za ně zodpovědných je znázorněn pomocí RACI matice, viz Tabulka 3.

Tabulka 3 - RACI matice odpovědnosti příslušných oddělení / osob vůči sw FatFreeCRM

	FatFreeCRM	Správa Klientů	Správa osob	Správa Leads	Správa oportunit	Správa projektů
Sales	A R	A R	A R	A R	A R	R C
Key Account	A R				I	I
CEO	C	I	I	I	C	A
Product Owner	R					I

Zdroj: Vlastní zpracování

Slack

Slack je primární nástroj, který firmy Juicyo s.r.o. používá pro komunikaci. Je to jediná online firemní komunikace. Stejně jako všechny ostatní sw aplikace, tak i tato je Open Source.

Slack slouží k týmové pracovní komunikaci. Juicyo s.r.o. má rozlišeno několik komunikačních kanálů:

1. komunikační kanály mezi pracovníky firmy bez závislosti na přidělených projektech,
 - komunikaci vidí všichni, anebo pouze ten komu je adresována.
2. komunikační kanály vytvořené právě pro daný projekt,
 - komunikaci vidí pouze členové, kteří pracují na daném projektu.

Založení komunikačních kanálů má na zodpovědnost Product Owner. Ve Slack následně komunikují všichni z projektového týmu, až na Stakeholdera, přičemž Watcher nefiguruje v konkrétních komunikačních skupinách, ale členové týmu s ním mohou kdykoli konzultovat průběh projektů.

Fabric.io

Fabric.io je monitorovací SW pro mobilní aplikace. Dosud není ve firmě zaveden (ke konci roku 2015), ale plánuje se s ním v blízké budoucnosti.

Fabric.io má 2 funkce:

- Distribuce beta verzí směrem ke klientovi.
- Správa chyb mobilních aplikací,
 - když nějaká aplikace přestane fungovat, „spadne“, tato aplikace nahlásí firmě Juicyo s.r.o. chybu.

Pozn.: Fabric.io nefunguje jako správa chyb webových aplikací, ta je prováděna emailem.

Zodpovědnost za chod Fabric.io mají členové vývojářského týmu, přičemž ho konzultují s Product Owner. Watcher a Stakeholder jsou o hlášeních z Fabric.io informováni.

Google Drive

Google Drive slouží ve firmě Juicyo s.r.o. ke sdílení dokumentů. Dokumenty na Google Drive vkládají členové vývojářského týmu a informují o této skutečnosti Product Owner.

Google Kalendář

Google Kalendář slouží k plánování harmonogramu schůzí. V aplikaci je zaznamenáno datum schůze, k jakému projektu schůze je naplánována a pro koho je určena.

O zadávání dat do Google Kalendáře se stará Product Owner, který je konzultuje s členy vývojářského týmu. Scrum Master, Watcher a Stakeholder jsou o naplánované schůzi informováni.

4.2.3.2 Scrum řízený pomocí OpenProject

OpenProject

Jak již už bylo zmíněno, aplikace OpenProject je ve firmě Juicyo s.r.o. využívána jako primární nástroj pro správu jednotlivých projektů.

Majitelé Juicyo s.r.o. se rozhodli používat aplikaci OpenProject z několika důvodů. Jedním z nich je, že aplikace OpenProject je Open Source, tudíž náklady za její využívání jsou minimální. Firma je navíc schopna upravit si zmíněnou aplikaci podle svých potřeb.

Dalšími důvody jsou prvky, které aplikace OpenProject využívá, a to především v oblasti agilního přístupu Scrum.

OpenProject umožňuje vytvářet a spravovat úkoly, User Stories a Epics. Tato data dokáže strukturovat právě podle metodiky Scrum, snadno je aktualizovat a zároveň ukázat, kdy a kdo daná data naposledy aktualizoval. Spolu s tím OpenProject umožňuje firmě

vytvářet a spravovat Product Backlog a Sprint Backlog². Product Owner je schopen vytvořit, udržovat a prioritizovat požadavky v Product Backlog. Při Sprint Planning je tým schopen snadno přiřadit požadavky do Sprint Backlog a následně je prioritizovat.

Ukázku vedení firemního Product a Sprint Backlog znázorňuje Obrázek 14.

Obrázek 14 – Ukázka Product Backlog a Sprint Backlog firmy Juicyo s.r.o.

Product Backlog		Sprint 2 Backlog	
ID	Description	Status	Points
2155	User Story: Uživatel se může registrovat...	In Progr...	8
2156	User Story: Uživatel se může přihlásit po...	In Progr...	5
2145	User Story: Uživatel může vytvořit WC n...	In Progr...	8
2134	User Story: Uživatel vidí WC na mapě ve ...	In Progr...	8
2135	User Story: Uživatel vidí WC v seznamu ...	In Progr...	13
2172	User Story: Administrátor může spravov...	Ready fo...	8
2171	User Story: Administrátor může spravov...	Ready fo...	8
2194	User Story: Administrátor může spravov...	Ready fo...	3
2184	User Story: Administrátor může "dočas...	Ready fo...	2
2142	User Story: Uživatel může WC u kterého ...	New	8
2149	User Story: Uživatel může přidávat obsa...	New	13
2174	User Story: Administrátor může spravov...	New	5
2176	User Story: Administrátor vidí seznam W...	New	3
2137	User Story: Uživatel si může zobrazit de...	New	8
2138	User Story: Uživatel si může zobrazit de...	New	2
2148	User Story: Uživatel může číst obsah WC...	New	13
2150	User Story: Uživatel vidí (jen dle barvy?) ...	New	13
2136	User Story: Uživatel vidí seznam "svých ...	New	13
2160	User Story: Uživatel si může prohlédnou...	New	13
2146	User Story: Uživatel může nasdílet WC z ...	New	1
2140	User Story: Uživatel se může stát sponz...	New	8
2154	User Story: Uživatel může nareportovat ...	New	3
2153	User Story: Uživatel může nareportovat ...	New	3
2152	User Story: Uživatel může nareportovat ...	New	3
2151	User Story: Uživatel může nareportovat ...	New	3
2141	User Story: Uživatel může WC u kterého ...	New	2
2147	User Story: Uživatel může WC, které má ...	New	2
2139	User Story: Uživatel v seznamu nebo det...	New	2
2159	User Story: Uživatel může v aplikaci zasl...	New	5
2162	User Story: Uživatel může zareagovat na...	New	5
2158	User Story: Uživatel může aplikaci dopor...	New	1
2157	User Story: Uživatel si může zobrazit inf...	New	2
2161	User Story: Aplikace uživatele pomocí n...	New	40

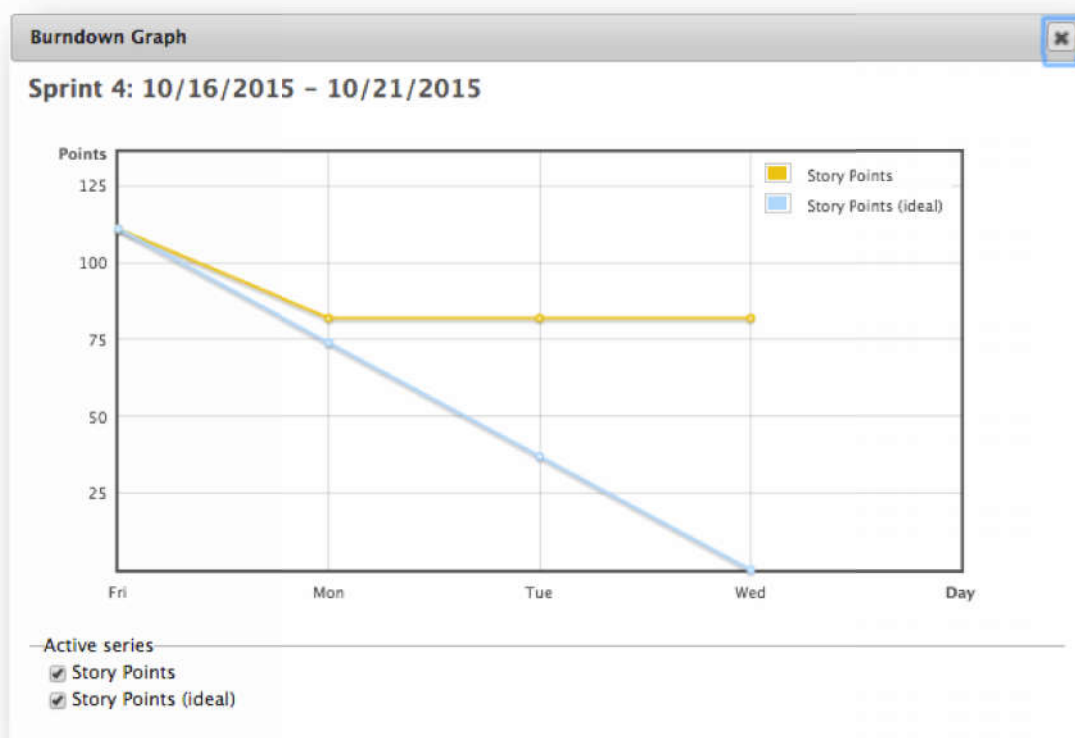
Zdroj: Juicyo s.r.o., SW OpenProject, 2015

Společně se správou Backlogs, vývojářskému týmu je v OpenProject umožněno ohodnotit User Stories pomocí Story Points, tj. Story bodů. Aplikace následně vypočítá sumu Story bodů všech User Stories zařazených v probíhajícím Sprintu. Story body umožňují vývojářskému týmu pomocí Burndown diagramu monitorovat pracovní úsilí.

² Ve firmě Juicyo s.r.o. je v softwaru OpenProject Product Backlog označován jako „Backlog“ a Sprint Backlog jako „Sprint“.

Obrázek 15 znázorňuje Burndown graf z probíhajícího Sprintu firmy Juicyo s.r.o. Na grafu jsou vidět dvě křivky, modrá křivka znázorňuje ideální „spalování“ práce (Story bodů) a žlutá křivka znázorňuje reálně odvedenou práci.

Obrázek 15 – Ukázka Burndown grafu z probíhajícího Sprintu firmy Juicyo s.r.o.



Zdroj: Juicyo s.r.o., *SW OpenProject*, 2015

Jednou z dalších funkcí OpenProject, které Juicyo s.r.o. usnadňují řízení projektů, je schopnost tvorby dokumentace výsledků schůzí Sprint Planning a Sprint Retrospective vycházejících z dat z Product Backlog a Sprint Backlog.

4.3 Rozbor nasazených modifikací metody Scrum v Juicyo s.r.o.

Scrum byl ve společnosti Juicyo s.r.o. implementován v roce 2014. Od té doby je ve firmě úspěšně provozován. Majitelé berou metodiku Scrum jako „návod“ k řízení jejich IT projektů. Nebojí se jej rozvíjet a rozšiřovat tak, aby se co nejlépe přizpůsobil jejich potřebám.

Jednou z těchto potřeb bylo k typickému Scrum týmu nutnost zavedení dvou dalších projektových rolí, Watcher a Stakeholder. Druhá modifikace, týkající se metody Scrum, je rozšíření fází vývoje Inkrementu. Třetí a poslední modifikací byla změna v Daily Scrum, způsobená vývojáři prováděním prací z domova.

Bližší o všech pozměněních pojednávají následující kapitoly.

4.3.1 Přizpůsobení rolí projektového týmu Juicyo s.r.o. pro řízení projektu

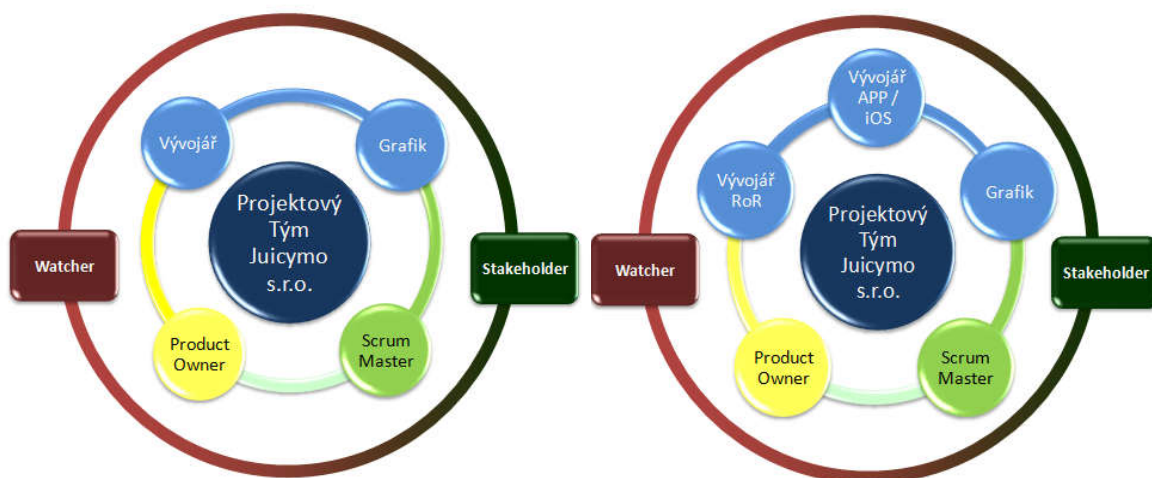
Agilní přístup Scrum se v procesu řízení zakázek v Juicyo s.r.o. objevuje již ve fázi Business Offer Estimates a s tím spojené fázi Vytvoření User Stories.

Během zmíněných fází se začne vytvářet tým, který, pokud klient nabídku odsouhlasí, na zakázce bude pracovat. Do programu OpenProject se zadá, kdo se projektu bude účastnit a v jaké roli.

Firma pro řízení projektů využívá pět rolí – Product Owner, Scrum Master, Team Members, Watcher a Stakeholder. Přičemž role Product Owner, Scrum Master a Team Members (v metodice Scrum nazývané jako členové vývojářského týmu) jsou definované metodikou Scrum jako členové Scrum týmu. Role Watcher a Stakeholder byly primárně zavedeny z důvodu zajištění přístupových oprávnění k dokumentaci projektu.

Personální sestavení týmu závisí na konkrétním projektu, kterým se firma zrovna zabývá. Tým může být sestaven podle povahy projektu pouze z grafika a vývojáře, nebo například z vývojáře RoR, vývojáře APP či iOS a z grafika, viz Diagram 6.

Diagram 6 – Příklady projektových týmů v Juicyo s.r.o.



Zdroj: *Vlastní zpracování*

V projektových týmech nemá nikdo pozici manažera. Firma se v tomto ohledu snaží dodržovat metodiku Scrum a ponechává volný přístup. Snaží se omezit vztah zaměstnavatel – zaměstnanec na úplné minimum, v práci jsou si všichni rovnocennými kolegy.

V roli Product Owner vystupuje v některých případech sám klient, pokud jsou jeho odborné znalosti dostačující, nebo Ing. Jukin či Ing. Moudrá. V roli Scrum Master vystupuje Ing. Jukin nebo Ing. Moudrá. Ostatní členové týmu, Team Members, jsou do projektu zařazeni podle jeho povahy.

4.3.1.1 Náplň práce jednotlivých členů projektového týmu a jejich zodpovědnosti

V řízení projektů pomocí metodiky Scrum existuje několik činností, za které mají jednotliví členové Scrum týmu určitou zodpovědnost. Tabulka 4 znázorňuje RACI matici odpovědností jednotlivých členů projektového týmu za stanovené činnosti.

Tabulka 4 – RACI matice – Zodpovědnosti jednotlivých členů projektového týmu

	Sestavení Product Backlog	Ocenění Product Backlog	Správa Product Backlog	Sestavení Sprint Backlog	Návrh Sprintu	Průběh Sprintu	Sestavení Epics	Vytvoření User Stories	Vytvoření Úoklů	Páce na User Stories	Práce na úkolech
Product Owner	A R	C	A R	C	C	I	A R	A R	C	I	I
Scrum Master	I	I	C	C	C	C	C	C	C	C	C
Team Members	C	A R	C	A R	A R	A R	C	R	A R	A R	A R
Watcher											
Stakeholder	C	I	I	I	I	I	I	I		I	

Zdroj: Vlastní zpracování

Product Owner

Product Owner může být klient sám, pokud k tomu má dostatečnou kvalifikaci, anebo jím bývá zástupce firmy Juicyo s.r.o., obvykle Ing. Jukin nebo Ing. Moudrá.

Product Owner má zodpovědnost za sestavení a správu Product Backlog, za sestavení Epics a společně s Team Members vytvoření User Stories.

Team Members s ním konzultují ocenění Product Backlog. Dále Product Owner společně s Team Members navrhuje Sprint a s ním spojený Sprint Backlog. Team Members s ním také mohou konzultovat tvorbu úkolů.

Product Owner je pouze informován o průběhu Sprintu a o práci na User Stories a úkolech.

Scrum Master

Scrum Master je člověk, který dohlíží na to, zda jsou metodiky Scrum správně dodržovány. V roli Scrum Master vystupuje Ing. Moudrá nebo Ing. Jukin, v závislosti na tom, kdo je právě Product Owner daného projektu.

Scrum Master je informován o sestavení a ocenění Product Backlog. V ostatních činnostech zaujímá roli konzultanta.

Team Members

Team Members jsou klíčovými členy projektového týmu pro dodání nového Inkrementu produktu v rámci každého Sprintu.

Členem vývojářského týmu v Juicyo s.r.o. je například vývojář, designér, specialista na UX (User Experience), architekt SW aplikace či konzultant.³

Team Members mají odpovědnost za ocenění Product Backlog, za návrh Sprintu a sestavení Sprint Backlog, vývojářskou práci v průběhu Sprintu, vytvoření úkolů a práci

³ Diagram 6 znázorňuje role Team Members v modré barvě.

na User Stories a úkolech. Také se společně s Product Owner dělí o zodpovědnost na vytvoření User Stories.

Product Owner s Team Members konzultuje sestavení a správu Product Backlog a sestavení Epics.

Watcher

Watcher je role, kterou metodika Scrum nedefinuje. Juicyo s.r.o. rozhodlo zavést tuto roli z důvodu zajištění přístupových oprávnění výkonné ředitelky k dokumentaci projektu. Z toho vyplývá, že funkci Watcher téměř vždy zastává ředitelka Juicyo s.r.o. Ing Jana Moudrá.

Watcher má přístup pouze *Lead Only* do všech projektů, tzn., kontroluje stav projektů, ale nemůže v dokumentaci projektu sám nic měnit. Pokud je z nějakého důvodu s vývojem projektu nespokojen, spojí se s Product Owner daného projektu a problém řeší skrz něj.

Stakeholder

Stakeholder je funkce, kterou zastává klient, resp. pověřený zástupce zadavatelské firmy, který není v pozici Product Owner. Pověřeným zástupcem zadavatelské firmy je míněna jedna, maximálně dvě osoby, které s firmou Juicyo s.r.o. komunikují (zadávají, navrhuji, kontrolují a konzultují projekt) za zadavatelskou firmu. Firma Juicyo s.r.o. nepřichází do kontaktu s jinými osobami ze zadavatelské firmy než se Stakeholder.

Stakeholder má, stejně jako Watcher, přístup *Lead Only* do svého projektu v softwaru OpenProject. S projektovým týmem však konzultuje podobu sestaveného Product Backlog. O všech ostatních probíhajících činnostech, až na vytvoření úkolů a práci na nich, je vždy informován.

4.3.2 Stavby úkolů při vývoji Inkrementu a jejich rozšíření v Juicyo s.r.o.

Juicyo s.r.o. jako většina firem používajících metodu Scrum zobrazuje svůj Sprint Backlog na Task Board. Od ostatních se však liší tím, že místo toho, aby ve fázi vývoje rozdělovala úkoly do tří stavů - „To Do“ („Co je potřeba udělat“), „Ongoing“ („Na čem se pracuje“) a „Done“ („Hotovo“), rozděluje fázi vývoje do stavů pěti – „New“ („Nový“), „In Progress“ („Pracuje se na tom“), „Ready for Test“ („Připraven na testování“), „Needs Info“ („Potřebuji informaci“) a „Done“ („Hotovo“). Pro srovnání viz Tabulka 5.

Tabulka 5 – Srovnání fází Scrum vs. Juicyo s.r.o

	Scrum			Juicyo s.r.o.				
	To Do	Ongoing	Done	New	In Progress	Ready for Test	Needs Info	Done
Úkol 1	✓			✓				
Úkol 2		✓				✓		
Úkol 3		✓					✓	
Úkol 4		✓			✓			
Úkol 5			✓					✓

Zdroj: Vlastní zpracování

4.3.2.1 Zodpovědnosti členů týmu k vykonání úkolu v určitém stavu

Juicyo s.r.o. se rozhodlo pro zavedení výše zmíněných pěti stavů z důvodu přehlednějšího řízení vývoje Inkrementu a pro jasnější kontrolu toho, jak si tým v daném Sprintu počíná.

Každému členovi týmu připadá v každém stavu určitá zodpovědnost k vytvářenému úkolu, viz Tabulka 6.

Product Owner je plně odpovědný za stav *Ready for Test*, kdy provádí testování dodané části inkrementu. Ve stavu *Needs info* komunikuje a podává informace Team

Members. A v neposlední řadě, pokud dodané části inkrementu projdou testem, zařadí je do stavu *Done*.

Scrum Master nepřebírá za stavy, v jakých se nacházejí úkoly, žádnou odpovědnost, pouze je informován o tom, když se úkol přesune do stavu *Done*.

Team Members jsou zodpovědní za to, že pro User Stories vytvoří úkoly, které vloží do stavu *New*. Vývojáři na daném úkolu pracují, když se nachází ve stavu *In Progress*. Ve stavu *Needs info* jsou zodpovědní za to, že Product Owner dostane požadované informace k testované části dodávaného inkrementu. Na závěr se podílejí na tom, že dodaná část inkrementu je hotová.

Watcher ani Stakeholder nemají ke stavům, v jakých se úkoly nacházejí, žádné odpovědnosti. Stakeholder je pouze informován v případě, že úkol byl přesunut do stavu *Done*, a část vyvíjeného inkrementu je tedy hotová.

Tabulka 6 – RACI matice, Zodpovědnosti členů týmu k vykonání úkolu v určitém stavu

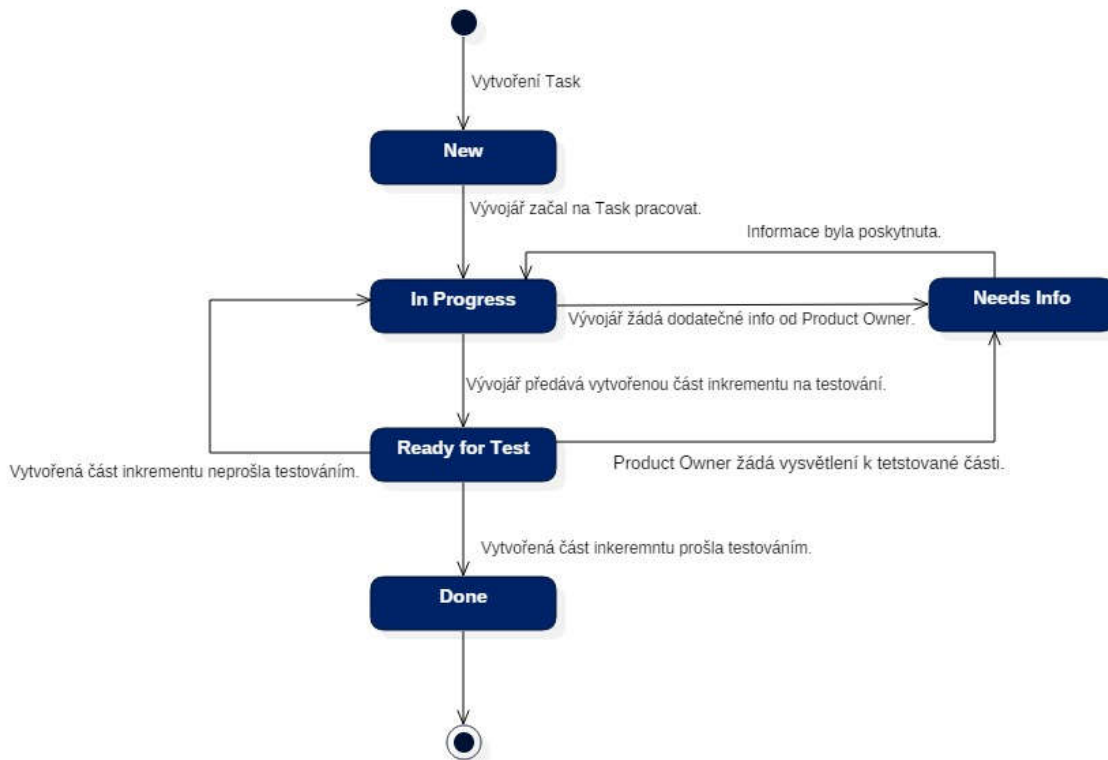
	New	In Progress	Ready for Test	Needs Info	Done
Product Owner			A R	A R C	A R
Scrum Master					
Team Members	A R	A R	I	A R	R
Watcher					
Stakeholder					I

Zdroj: *Vlastní zpracování*

4.3.2.2 Stavby v procesu práce na úkolech

Proces, kterým jednotlivé User Stories, popř. úkoly (Tasks), prochází je znázorněn na stavovém diagramu níže, viz Diagram 7.

Diagram 7 – Stavový diagram, Stavby úkolů



Zdroj: StarUML, Vlastní zpracování

Ve stavu *New* se nachází nově vytvořené úkoly, které se zadaly do informačního systému firmy. Na úkolu se však zatím nepracuje.

Jakmile vývojář začne na zadaném úkolu pracovat, přesune si ho do stavu *In Progress*. V tomto stavu úkol zůstává tak dlouho, dokud není splněn a přesunut do stavu *Ready for Test*, anebo dokud vývojář ke splnění úkolu nepotřebuje další informace. V tom případě přesune úkol do stavu *Needs Info*.

Ve stavu *Ready for Test* je úkol vývojářem označen za hotový a je poslán na otestování. Testováním je ověřeno, že daná vyvíjená část softwaru obsahuje a funguje tak, jak je to

v User Story požadováno. Pozn.: Testování je prováděno až v okamžiku, kdy jsou ve stavu *Ready for Test* přítomny všechny úkoly spadající pod danou User Story.

Testování provádí vždy Product Owner. V případě pozitivního výsledku testování je User Story přeřazena do stavu *Done*. V případě negativního výsledku testování je User Story vrácena do stavu *In Progress* k předělání. Pokud Product Owner žádá vysvětlení k testované části, přesune úkol do stavu *Needs Info*.

Needs Info je stav, ve kterém se nachází úkol, když vývojář potřebuje / žádá doplňující informace k dalšímu vývoji, či Product Owner vysvětlení k testované části. Do systému se zadá, ke komu je dotaz směřován. Potom, co je dotaz daným člověkem zodpovězen, je Úkol vrácen do stavu *In Progress*.

Do stavu *Done* je User Story přesunuta, pokud je celá vytvořena, otestována s pozitivním výsledkem a považována za definitivně hotovou, tj. vytvořená User Story splnila týmovou definici „Hotovo“. Do stavu *Done* má právo přesouvat hotové User Stories pouze Product Owner.

4.3.2.3 Stavby úkolů v uživatelském rozhraní OpenProject

User Story, pokud je složitějšího charakteru, představuje soubor úkolů, tzv. Tasks. Jednotlivé úkoly User Story jsou přiřazeny konkrétnímu člověku, který za ně zodpovídá.

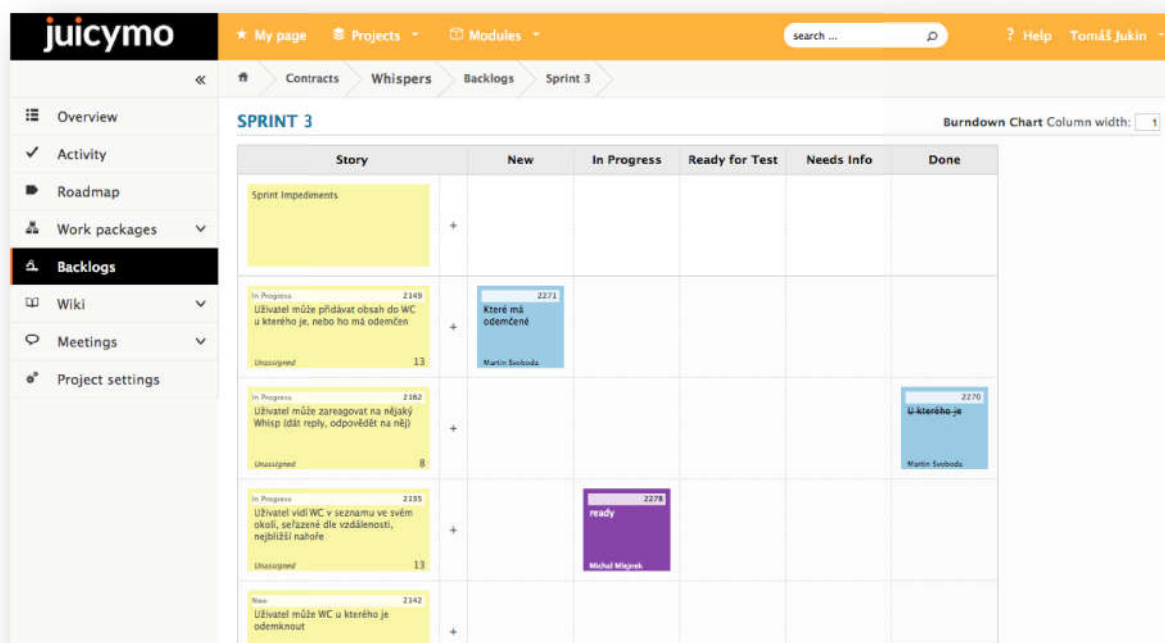
Každý vývojář, který se podílí na dané zakázce, má ve firemním IS vygenerovaný svůj Seznam úkolů. Pokud v jeho seznamu nastane změna, např. byl mu vytvořen nový úkol, nebo byl na něj vznesen dotaz (stav úkolu v *Needs Info*), přijde mu na jeho pracovní email zpráva a daný úkol se objeví ve výše zmiňovaném Seznamu.

Každý úkol, na kterém vývojář pracuje, obsahuje jeho historii. Tato historie slouží například k lepší a konkrétnější komunikaci, když se úkol nachází ve stavu *Needs Info*.

Příklad plnění Sprintu, rozpracování User Stories a úkolů, znázorňuje Obrázek 16. V levém sloupci pojmenovaném „Story“, ve žlutých obdélnících, jsou seřazeny jednotlivá User Stories tak, jak si je tým přetáhl na Sprint Planning poradě z Product Backlog. Dalších pět sloupců znázorňuje aktuální stavy úkolů – *New*, *In Progress*, *Ready for*

Testing, Needs Info a *Done*. Každý úkol je označen svým identifikačním číslem, názvem a jménem vývojáře, který za něj zodpovídá. Úkoly jsou rovněž pro větší přehlednost barevně odlišeny právě podle zodpovídajícího vývojáře.

Obrázek 16 – User Story v průběhu Sprintu.



Zdroj: JuicyMo s.r.o., SW OpenProject, 2015

4.3.3 Změna ve vedení Daily Scrums

Juicyo s.r.o. poctivě dodržuje všechny schůze, které definuje metodika Scrum, až na Daily Scrums, tzv. Stand-Up meetings. V současnosti se tyto porady konají velmi sporadicky. Na vině je nastavený systém pracovní doby, kdy firma nechává svým zaměstnancům možnost práce z domova, v jim vyhovujících pracovních dobách.

V případě, že vývojáři pracující na stejném projektu se sejdou v kancelářních místnostech v sídle firmy, Stand-Up meeting proběhne. V opačném případě, kdy vývojáři odvádí svou práci z domova, nenastává mezi nimi žádná komunikace.

Vedení firmy Juicyo s.r.o. si je vědomo toho, že situace ohledně konání Daily Scrums není ideální, ale zatím se mu nepodařilo najít efektivní systém komunikace mezi vývojáři, který by daný problém vyřešil.

4.4 Vlastní návrhy na zlepšení řízení IT zakázek ve firmě Juicyo s.r.o.

4.4.1 Zřízení komunikačního kanálu pro Daily Scrums

Problém

V Juicyo s.r.o. se Daily Scrums konají velmi sporadicky. Pracovníci nejsou schopni se sejít v konkrétním čase, aby se synchronizovali a předali si informace k nadcházejícímu pracovnímu dni.

Příčina problému:

1. práce pracovníků z domova,
2. každý pracovník si sám určuje svou pracovní dobu.

Návrh řešení

Juicyo s.r.o. ke vnitropodnikové komunikaci ohledně projektů využívají primárně SW nástroj Slack. V něm má Juicyo s.r.o. zřízeny komunikační kanály jak mezi pracovníky firmy bez závislosti na přidělených projektech, tak i komunikační kanály vytvořené právě pro daný projekt, ve kterých komunikaci vidí pouze členové, kteří pracují na daném projektu.

Jako řešení pro špatně fungující Daily Scrums je navrženo zřízení dalšího komunikačního kanálu v aplikaci Slack, který bude výhradně používán pro Daily Scrums - v aplikaci bude pro každý probíhající projekt vytvořen další komunikační kanál, který bude sloužit výhradně pro členy týmu pracujícím na daném projektu a ve kterém si budou vývojáři sdělovat informace, které by se za normálních okolností sdělili na Stand-Up meetingu.

Zprávy pro své týmové kolegy by vývojář zanechával vždy po skončení jeho pracovní doby. Zpráva bude obsahovat: na čem daný den pracoval, na čem bude pracovat další den a zda mu něco brání v práci. Návrh Daily Scrum zprávy viz Obrázek 17.

Na začátku své pracovní doby si vždy vývojář přečte Daily Scrum zprávu od svých spolupracovníků. Tímto způsobem získá včas potřebné informace, které mu pomohou se

synchronizovat se zbytkem vývojářského týmu, bez ohledu na to kdy kdo začal v daný den pracovat.

Obrázek 17 – Navržená podoba Daily Scrum zpráv ve Slack

Projekt XY – Daily Scrum Messages

Message from: Jan Date: 29.2.2016 10:00	Message from: Tomáš Date: 29.2.2016 16:00	Message from: Martin Date: 29.2.2016 12:00
Work completed today:	Work completed today:	Work completed today:
Work planned for tomorrow:	Work planned for tomorrow:	Work planned for tomorrow:
Facing any problems or issues:	Facing any problems or issues:	Facing any problems or issues:

Zdroj: Vlastní zpracování

Očekávaná zlepšení:

- zlepšení komunikace mezi vývojáři,
- vývojáři budou mít lepší přehled, kdo na čem pracoval,
- zlepšení synchronizace vývojářského týmu.

4.4.2 Zlepšení odhadů časové náročnosti úkolů při Sprint Planning

Problém

Na poradě Sprint Planning si tým z Product Backlog vybere User Stories, na kterých se bude pracovat v následujícím Sprintu. Vývojáři následně k naplnění User Stories vytvoří úkoly. Těmto úkolům namísto Story bodů již přiřazují časové jednotky, které určují pravděpodobnou délku trvání jejich vytvoření.

Délka trvání úkolu je stanovena na základě předchozích zkušeností daného vývojáře, který má úkol splnit. Pokud podobný / stejný úkol již někdy dělal, je jeho časový odhad poměrně přesný. Problém však nastává v okamžiku, kdy vývojář má vytvořit prvek, se kterým nemá dosud žádné zkušenosti. V tom případě vývojář tipuje v rámci svých znalostí a zkušeností a délka trvání úkolu je jeho ničím nepodloženým, čistým odhadem.

Návrh řešení: *zpřesnění odhadů délky trvání úkolů pomocí třibodového odhadu.*

Třibodový odhad je v projektovém řízení využíván v problematice neurčitosti při odhadování pracnosti dílčích činností. Manažeři předkládají své odhady vypočítané pomocí tří veličin, které zastupují tři možné předpoklady dokončení činnosti [4]:

1. *Optimistická doba trvání činnosti; „a“*

Optimistická doba trvání předpokládá, že vše proběhne podle plánu s minimálními problémy. Tato situace nastane v 1 % případů.

2. *Pesimistická doba trvání činnosti; „b“*

Tento časový odhad předpokládá, že nic nepůjde podle plánu, vyskytne se maximum problémů. Tato situace nastane v 1 % případů.

3. *Nejpravděpodobnější doba trvání činnosti; „m“*

Nejpravděpodobnější doba trvání činnosti je doba, která by, v myslí manažera, měla být zaznamenávána znovu a znovu, v případě, že se daná činnost běžně opakuje.

Očekávaná doba trvání činnosti T_e (střední hodnota beta rozdělení) je definována rovnicí [4]:

$$T_e = \frac{a + 4m + b}{6}$$

Předtím, než tyto tři doby trvání mohou být skombinovány do jediného výrazu pro očekávanou dobu, musí být provedeny dva předpoklady. První předpoklad je, že standardní odchylka σ , je jedna šestina rozsahu požadavků času. Tento předpoklad vychází z teorie pravděpodobnosti, kde koncovými body křivky jsou tři standardní odchylky od průměru. Druhý předpoklad vyžaduje, aby rozdělení pravděpodobnosti doby potřebné pro činnost bylo vyjádřené jako beta rozdělení. [3]

Pro vývojáře Juicyo s.r.o. to znamená, že za optimistickou dobu trvání „ a “ budou dosazovat hodnotu takové doby trvání, která odpovídá splnění úkolu za naprosto ideálních podmínek, bez jakýchkoliv překážek a komplikací.

Za pesimistickou dobu trvání „ b “, budou dosazovat hodnotu, která vyjadřuje délku činnosti v případě opakujících se nepředvídatelných překážek a z nich vyplývajících komplikací.

Nejpravděpodobnější doby trvání činnosti „ m “ pro vývojáře bude časový interval, ve kterém by danou činnost s nejvyšší pravděpodobností rutinně odváděli při normálních podmínkách.

Po dosazení tří výše zmíněných odhadů do vzorce, vyjde odhad očekávané doby trvání úkolu, jež odráží jak míru rizika v úkolu, tak i závažnosti dopadu optimistických a pesimistických rizik. Ve výsledku by zavedení metody počítání odhadů délky trvání úkolů pomocí tříbodového odhadu mělo vést k zpřesnění odhadů dob trvání úkolů.

Očekávaná zlepšení:

- zpřesnění odhadů dob trvání úkolů,
- přesnější plánování Sprintů.

Příklad použití třibodového odhadu:

Vývojář má odhadnout délku trvání úkolu, s jehož plněním nemá žádné zkušenosti. Normálně by odhadl dobu trvání úkolu na 7 hodin. Vývojář se však rozhodne použít třibodový odhad. Za optimistickou dobu trvání zvolí hodnotu 5 a za pesimistickou dobu trvání hodnotu 12. Hodnoty dosadí do vzorce:

$$T_e = \frac{5 + 4 \cdot 7 + 12}{6}$$

$$T_e = 7,5$$

Očekávaná doba trvání činnosti T_e vyšla 7,5 hodiny, tj. o půl hodiny delší čas, než který vývojář původně odhadoval.

Půl hodina se v tomto případě může zdát jako zanedbatelné číslo, ale v případě, že by se tato odchylka vyskytla například u deseti úkolů v jednom projektu a stejná situace by se opakovala i u dalších dvou rozpracovaných projektů, z půl hodiny je rázem 15 hodin. To jsou téměř dva celé pracovní dny, což už tak zanedbatelné není.

4.4.3 Zlepšení ve sledování vytížení lidských zdrojů

Problém

Řízení IT projektů pomocí Scrum se od klasického projektového řízení liší tím, že vývojářské týmy se samy organizují. Vedení firmy sice zpočátku určí, kdo na projektu bude pracovat, ale organizace vývoje zakázky je pak výhradně v rukách pověřeného týmu. To samozřejmě vede k tomu, že vedení firmy ztrácí přehled o tom, jak který vývojář je ve skutečnosti časově vytížený.

Podobná situace panuje i ve firmě Juicy s.r.o. Vedení firmy přiřazuje vývojáře k určitým projektům, ale následně už nesleduje, jak je který zaměstnanec je vytížený. Přiřazení lidských zdrojů k určitým projektům má vedení pouze „ve své hlavě“. Účastníci na tvorbě zakázky se sice dají dohledat v softwaru OpenProject, ale dále neexistuje žádný dokument, ve kterém by bylo přehledně zaznamenáno, kolik projektů má vývojář k sobě přiřazeno.

Co se týče vytíženosti vývojářů během probíhajících Sprintů, tak o to se vedení firmy, z důvodů obtížného sledování, vůbec nezajímá. Spoléhá na to, že vývojáři si při Sprint Planning poradě nabere tolik práce, kolik si myslí, že zvládnou. Z toho vyplývá, že Time Management vývojářské práce je výhradně na vývojářích samotných. Situací, kdy by vývojář neodhadl své pracovní vytížení a nabral si více práce, než je schopen zvládnout, je velmi málo, v řádu jednotek. Ale i to málo občas stačí k nedodání smlouveného inkrementu produktu.

Výše uvedené poukazuje na to, že firmě chybí nástroj, kterým by šlo sledovat jak přiřazení vývojářů k projektům, tak i časovou vytíženost jednotlivých vývojářů.

Návrh řešení:

- 1) *Zřízení dokumentu sledující přiřazení lidských zdrojů k jednotlivým projektům.*
- 2) *Zavedení pracovního kalendáře pro všechny pracovníky firmy, ve kterém by bylo možné sledovat jejich alokaci a časové vytížení v průběhu Sprintů.*

4.4.3.1 Zřízení dokumentace sledující přiřazení lidských zdrojů k jednotlivým projektům

Pro přehledné sledování alokace jednotlivých vývojářů k projektům, je navrženo zřízení dokumentace, ve které budou jednotlivé alokace lidských zdrojů zaznamenávány.

Dokumentace bude obsahovat:

- jméno a příjmení,
- zkušenost vývojáře,
 - Senior, Junior;
- technické zaměření vývojáře,
 - Web Frontend, Web Backend, iOS, Android, Windows Phone, Unity 3D, Dart, IoT, AI;
- projekty, na kterých vývojář právě pracuje,
- role v projektovém týmu.

Vedení dokumentace o alokaci jednotlivých vývojářů k projektům povede nejen k tomu, že vedení Juicyo s.r.o. bude mít jasný přehled o tom, na čem který vývojář dělá, ale zároveň z dokumentace půjde vyčíst, která technická zaměření jsou v projektech nejžádanější. Tento fakt může vedení firmy následně využít v případném náboru či školení zaměstnanců.

Očekávaná zlepšení:

- jasný přehled o alokaci vývojářů na projektech,
- sledování využití technických zaměření v projektech,
- cílenější vybírání či školení zaměstnanců.

4.4.3.2 *Zavedení pracovního kalendáře pro lidské zdroje*

Pro sledování časové vytíženosti jednotlivých vývojářů bylo firmě Juicymo s.r.o. doporučeno zavést pracovní kalendář pro jednotlivé lidské zdroje.

V navrhovaném nástroji budou nejdříve zadány základní údaje týkající se práce každého zaměstnance firmy, jako jsou například, délka pracovní doby, dny kdy zaměstnanec pracuje apod.

Následně bude pro každého zaměstnance vytvořena šablona pracovního kalendáře v podobě tabulky. Do svého pracovního kalendáře si každý vývojář podle vlastního uvážení rozvrhne přidělené úkoly ze Sprint Planning porady⁴, příklad viz Tabulka 7. Jednotkami zadávaných hodnot do kalendáře jsou stanoveny hodiny.

Z pracovních kalendářů zaměstnanců se bude dát vygenerovat i souhrnný, celofiremní pracovní kalendář, ve kterém budou uvedeni všichni zaměstnanci a který bude ukazovat celkovou časovou vytíženost jednotlivých zaměstnanců v jednotlivých dnech, příklad viz Tabulka 8.

Příklad použití pracovního kalendáře:

Vývojář Tomáš má osmihodinovou pracovní dobu. V Projektu 1 pracuje na 10 úkolech, jejichž součet dob trvání je celkem 19. Práci na úkolech si rozvrhne do 8 dnů; viz Tabulka 7, řádek *Projekt 1*. V Projektu 2 má na práci 6 úkolů rozvržených do 6 dnů s celkovým součtem doby trvání 21 hodin; viz Tabulka 7, řádek *Projekt 2*. V Projektu 3 musí udělat 9 úkolů, s celkovou dobou trvání 17 hodin, které si rozvrhl do 7 dnů; viz Tabulka 7, řádek *Projekt 3*.

Po zadání pracovní náročnosti každého dne se v tabulce vývojáři vypočítá celková dob rozvržené denní práce (viz Tabulka 7, řádek *Celkem*) a zvýrazní dny, kdy vývojář přesahuje délku své pracovní doby (viz Tabulka 7, řádek *Celkem* / sloupec 24.2.),

⁴ Na Sprint Planning poradách si tým stanovuje plán pro další Sprint a rozděluje mezi vývojáře úkoly vyplývající z User Stories. Z plánu nadcházejícího Sprintu se dá zjistit, jak který vývojář bude časově vytížený.

a dny, které nejsou dostatečně pracovní vytiženy (viz Tabulka 7, řádek *Celkem* / sloupec 1.3.). Vývojář si dle toho může pracovní plán upravit.

V případě, že by většina dní byla v červených číslech, je zřejmé, že si vývojář nabral více práce, než bude schopen udělat. V tomto případě musí začít řešit se svými týmovými kolegy, zda by někdo nemohl udělat jemu přidělenou práci za něj. V případě, že by nikdo nemohl jeho práci převzít, musí tato skutečnost (nabrání si více práce, než je vývojář schopen udělat) být projednána na poradě Sprint Retrospective a tým se musí při plánování dalšího Sprintu z této chyby poučit.

V jiném případě, kdy by se například rozeběhl Projekt 4, vývojář si je schopen vypočítat, že si může nabrat pouze tolik úkolů, které by mu celkem zabraly pouze 7 hodin práce (součet řádku *Nevyužito*, viz Tabulka 7).

Tabulka 7 – Pracovní kalendář vývojáře

Tomáš	23.2.	24.2.	25.2.	26.2.	27.2.	28.2.	29.2.	1.3.	2.3.
Projekt 1	2	4	2	4	2	2	-	1	2
Projekt 2	2	5	3	0	3	6	-	0	2
Projekt 3	4	1	2	4	0	1	-	3	2
Celkem	8	10	7	8	5	9	0	4	6
Nevyužito	0	-2	1	0	3	-1	-	4	2

Zdroj: Vlastní zpracování

Tabulka 8 – Pracovní kalendář souhrnný

	23.2.	24.2.	25.2.	26.2.	27.2.	28.2.	29.2.	1.3.	2.3.
Jan	6	8	5	7	8	0	0	3	7
Tomáš	8	10	7	8	5	9	0	4	6
Martin	3	5	6	8	0	0	2	5	7
Jana	7	8	8	8	0	0	5	9	9

Zdroj: Vlastní zpracování

Předpoklady pro správné fungování Systému:

- poctivé dodržování vyplňování hodnot do systému,
- vytváření si detailnějšího rozvrhu prací již na poradě Sprint Planning.

Očekávaná zlepšení:

- zlepšení Time Managementu pracovníků,
- zlepšení odhadů při plánování Sprintů – tým si nabere tolik práce na Sprint, kolik má na vytvoření dané práce doopravdy času,
- vedení firmy získá přehled o využití pracovníků,
- vedení firmy získá přehled o efektivnosti vedení projektů.

5 Výsledky a diskuse

Společnost Juicyo s.r.o. praktikuje agilní přístup Scrum od roku 2014. Juicyo s.r.o. mělo při implementaci Scrum tu výhodu, že před jeho implementací nemělo zavedený jiný styl projektového řízení, a tudíž nemělo odkud převzít „špatné návyky“, které by odporovaly definicím Scrum.

Na základě rozboru řízení firmy, bylo zjištěno, že od doby, kdy firma metodiku Scrum zavedla, nejen, že si tuto metodiku dobře osvojila, ale zároveň se ji odvážíla začít upravovat tak, aby co nejlépe vyhovovala jejich potřebám při co nejlepším zachování definicí Scrum.

Rozbor ukázal, že firma Juicyo s.r.o. se definice Scrum nedrží pouze v jednom případě, a to v pravidelném pořádání Daily Scrums. Tyto denní porady se ve firmě konají velmi sporadicky z důvodů vykonávání vývojářských prací mimo firemní kancelář. Pro řešení současného stavu bylo firmě navrženo zřízení nového komunikačního kanálu v softwaru Slack. Toto řešení by mělo být náhradou za fyzické konání Daily Scrums a vést k lepší synchronizovanosti vývojářského týmu.

Detailní rozbor stavů, kterými Inkrement prochází při vývoji, ukázal, jak si Juicyo s.r.o. přizpůsobilo klasické stavy vývoje. Juicyo s.r.o. se uchýlilo k rozřazení klasického stavu vývoje Inkrementu „Ongoing“ na tři stavy: „In Progress“, „Ready for Test“ a „Needs Info“. Tímto krokem, nejen že si zajistilo přehlednější řízení vývoje Inkrementu a jasnější kontrolu toho, jak si tým v daném Sprintu počíná, ale zároveň zapojením stavu „Ready for Test“ dosáhlo kontroly jakosti vytvářeného Inkrementu.

Společnost si rovněž poradila s nutností zajistit přístupová práva k projektové dokumentaci nejen Scrum týmu, ale zároveň i dalším osobám, které na projektu mají podíl. Zavedla dvě další projektové role, Watcher a Stakeholder, které, ač mají při vytváření produktu malé pravomoci, mohou průběh projektu kontrolovat.

V celkovém pohledu firma uchopila metodiku Scrum velmi dobře a až na pravidelné konání Daily Scrums ji poctivě dodržuje. Firma využila prostor pro různé způsoby řízení pomocí Scrum a tento prostor si vhodnými přizpůsobeními upravila dle svých potřeb. Stále zde však zůstávají příležitosti na zlepšování. Juicyo s.r.o. se v současné době například vůbec nezajímá o management lidských zdrojů. V této oblasti

byla pro společnost navržena dokumentace pro sledování přiřazení lidských zdrojů k jednotlivým projektům a nástroj, pracovní kalendář, pro sledování vytíženosti lidských zdrojů.

Další oblastí, kde se nachází potenciál na zlepšení, je ve zpřesnění odhadů délek trvání úkolů, a tím spojeného přesnějšího plánování Sprintů. Pro tuto oblast bylo společností Juicyo s.r.o. představeno stanovení odhadů pomocí tříbodového odhadu.

Vytvořené návrhy nástrojů na zlepšení dosavadního řízení IT projektů byly diskutovány s Ing. Jukinem, spolumajitelem firmy Juicyo s.r.o. a manažerem Technického oddělení. Během tohoto rozhovoru proběhla nejprve prezentace zmiňovaných návrhů a poté byly jednotlivé návrhy rozebírány z hlediska jejich použitelnosti v praxi.

O zřízení nového komunikačního kanálu v softwaru Slack se Ing. Jukin vyjádřil, že navrhovaný způsob vedení Daily Scrums bude ve firmě zaveden na testovací období dvou Sprintů. Po skončení testovacího období proběhne evaluace této metody komunikace, a pokud se osvědčí, bude v ní firma pokračovat i nadále.

Druhý návrh, používání tříbodového odhadu pro zpřesnění délky trvání úkolů, byl přijat Ing. Jukinem jak se zájmem, tak ale i s menšími pochybnostmi, zda navrhovaná metoda bude mít nějaký významnější efekt: „Uvidíme, jak se odhadování délky trvání úkolů pomocí tříbodového odhadu odrazí na statistikách odhadů Sprintů a výkonnosti jednotlivých týmů.“ Ing. Jukin však potvrdil, že tříbodový odhad bude představen a vysvětlen všem pracovníkům firmy. Pracovníkům bude doporučeno tuto metodu používat hlavně v případech, kdy stojí před dosud neprováděnými úkoly.

Návrh na zlepšení sledování lidských zdrojů byl Ing. Jukinem přijat s neutrálním ohlasem. „V současné době Juicyo s.r.o. nedisponuje žádným nástrojem, ani nepoužívá žádnou metodu, kterou by sledovalo alokaci a vytížení vývojářů, protože tuto aktivitu shledalo vedení firmy za časově náročnou,“ připustil Ing. Jukin. „Případ, že by si časovou vytíženost hlídal každý vývojář sám, se zdá být dobrým kompromisem. Nedokážu si však představit vést tyto záznamy pomocí, například excelovských tabulek. Pokud bychom ve firmě zavedli sledování lidských zdrojů, museli bychom nejdříve navrhnout user-friendly software, s jehož pomocí by se tato činnost efektivně dala provozovat.“

V celkovém hodnocení Ing. Jukin navrhovaná řešení označil za: „řešení zapadající do konceptu řízení firmy“ a „použitelná pro zavedení do praxe“.

Z výše uvedeného vyplývá, že vypracované návrhy mají velký potenciál se stát přínosem v praxi řízení firmy. Výsledky nasazení těchto nástrojů by bylo vhodné zpracovat v dalších pracích.

6 Závěr

Tato diplomová práce byla zaměřena na problematiku agilního přístupu Scrum v řízení IT projektů, konkrétně na popis a rozbor stávajícího stavu agilních přístupů ve vybrané firmě. Na základě rozboru byly pro firmu vypracovány vlastní návrhy na zlepšení stavu řízení IT projektů.

V první, teoretické části, byly popsány základní charakteristiky a pravidla agilního přístupu Scrum. Bylo vysvětleno, že Scrum je postaven na třech pilířích, a to transparentnosti, kontrole a přizpůsobení. V práci byly rovněž popsány události definované metodikou Scrum, tj. Sprint a porady Sprint Planning, Sprint Review, Sprint Retrospective a Daily Scrums. V práci byly představeny i role, které Scrum rozeznává, jako Product Owner, Scrum Master a člen vývojářského týmu. Další Scrum tématem, kterým se teoretická část práce zabývala, byly artefakty Scrum: Product Backlog, Sprint Backlog a Inkrement. V závěru teoretické části práce byly představeny nástroje využívané v agilním vývoji softwaru, jako je Úkol, User Story a Epics.

Ve vlastní části práce byla představena firma Juicyo s.r.o., která figuruje na trhu vývoje IT technologií od roku 2013, a byl popsán současný stav řízení jejích IT projektů. Následně bylo řízení IT projektů pomocí agilního přístupu Scrum podrobena rozboru. Rozbor byl primárně zaměřen na případy, ve kterých se řízení firmy odchylovalo od definicí a pravidel Scrum. Byly zpracovány výstupy, ve kterých bylo detailně popsáno, proč firma přistoupila na dané změny a jakým způsobem tyto změny používá při řízení IT projektů.

Na základě výstupů rozboru, byla firmě navrhnutá opatření na zlepšení dosavadního stavu řízení IT projektů. Nad návrhy byla vedena diskuze s vedením firmy Juicyo s.r.o. a získána zpětná vazba a náměty pro případné zpracování dalších prací, například prací zabývajících se zjištěním dopadů navržených opatření pro zlepšení řízení IT projektů, či vytvořením softwaru pro sledování vytížení lidských zdrojů v projektech, které se řídí metodikou Scrum.

Práce jako celek přinesla nejenom pohled na agilní přístup Scrum, ale také praktickou ukázkou toho, jak se s pomocí Scrum dají řídit IT projekty v existující společnosti. V práci byl prezentován postup od prvotního sběru podkladových dat, přes

představení společnosti Juicymo s.r.o., popsání a následného rozboru řízení IT projektů ve firmě, až po vlastní návrhy na zlepšení současného řízení.

7 Zdroje

Tištěné dokumenty:

- [1] HENRIK KNIBERG. *Scrum and xp from the trenches: how we do scrum*. [S.l.: C4Media Inc.], 2007. ISBN 9781430322641
- [2] HENRIK KNIBERT & MATTIAS SKARIN a FOREWORDS BY MARY POPPENDIECK & DAVID ANDERSON. *Kanban and Scrum: making the most of both*. S.l.: C4Media, Inc, 2010. ISBN 9780557138326.
- [3] HILLIER, Frederick S a Gerald J LIEBERMAN. *Introduction to operations research*. 4th ed. Oakland: Holden-Day, 1986. ISBN 0-8162-3871-5.
- [4] KERZNER, Harold. *Project management: a systems approach to planning, scheduling, and controlling*. 10th ed. Hoboken, New Jersey: John Wiley & Sons, c2009. ISBN 978-0-470-27870-3.
- [5] SCHWABER, Ken. *Agile project management with Scrum*. Redmond, Wash.: Microsoft Press, c2004, xix, 163 p. ISBN 073561993x.

Internetové zdroje:

- [6] Agile Project Management. *EMag Issue* [online]. 2014, 2014(18), 46 [cit. 2016-03-01]. Dostupné z: <http://www.infoq.com/minibooks/emag-agile-project-management>
- [7] Blog: It's a Sprint Review Not a Sprint Demo! - Innolution. *Home of Corporate On-site Scrum & Agile Training & Coaching from Author Ken Rubin - Innolution* [online]. Innolution, LLC., 2014 [cit. 2016-03-01]. Dostupné z: <http://www.innolution.com/blog/its-a-sprint-review-not-a-sprint-demo>
- [8] Difference between Product backlog Grooming, Sprint Planning and Elaboration blog. *Agile and IT Management Courses - Capri Academy* [online]. Solihull, 2016 [cit. 2016-03-01]. Dostupné z: <http://www.capriconsulting.co.uk/difference-between-productbacklog-grooming-sprint-planning-and-elaboration/>

- [9] Guide to Agile Practices. *Guide to Agile Practices* [online]. Agile Alliance and Institut Agile, 2016 [cit. 2016-03-01]. Dostupné z: <http://guide.agilealliance.org/guide/backlog-grooming.html>
- [10] HUNDERMARK, Peter. Do Better Scrum. *InfoQ* [online]. 2015, 2014(3), 76 [cit. 2016-03-01]. Dostupné z: <http://www.infoq.com/minibooks/do-better-scrum>
- [11] KALTENECKER, Siegfried. Leading Self-Organising Teams. *InfoQ* [online]. 2015, 2015, 143 [cit. 2016-03-01]. Dostupné z: <http://www.infoq.com/minibooks/leading-self-organising-teams>
- [12] Scrum - Events. *Tutorials for SAP Lumira, PHP7, Python3, SAP SD, Network Security, TurboGears, DocumentDB, SAP Fiori, VLSI Design, Web Socket, HCatalog, SAP ABAP, SAS, Crystal Report, Database Testing, MVVM, Web2Py, ReactJS, MS Project, Silverlight, Foundation, Django, Apache Storm, ETL Testing, Pure.CSS, Entity Framework, Cordova, Materialize, Windows 10, Less, Statistics, Embedded Systems, Penetration Testing, Jenkins, WebRTC, SASS, Codeigniter, QlikView, SAP HANA, WxPython, Social Intelligence, Zookeeper, I* [online]. Hyderabad Telangana, 2016 [cit. 2016-03-01]. Dostupné z: http://www.tutorialspoint.com/scrum/scrum_events.htm
- [13] Scrum Community - Scrum Alliance. *Home - Scrum Alliance* [online]. SCRUM ALLIANCE®, Inc., 2015 [cit. 2016-03-01]. Dostupné z: <https://www.scrumalliance.org/community/articles/2015/february/product-backlog--better-ordered-than-prioritized>
- [14] Scrum History | Scrum Guides. *Home | Scrum Guides* [online]. 2014 [cit. 2016-03-01]. Dostupné z: <http://www.scrumguides.org/history.html>
- [15] Scrum Product Owner: The Agile Product Owner's Role. *Scrum Training Agile Training from ScrumMaster Mike Cohn* [online]. Broomfield: Mountain Goat Software, 2016 [cit. 2016-03-01]. Dostupné z: <https://www.mountaingoatsoftware.com/agile/scrum/product-owner>
- [16] Scrum User Stories - International Scrum Institute. *SCRUM-INSTITUTE.ORG - USD 29 ACCREDITED SCRUM CERTIFICATIONS* [online]. International Scrum

- Institute, 2016 [cit. 2016-03-01]. Dostupné z: http://www.scrum-institute.org/Scrum_User_Stories.php
- [17] Scrum Values | Agile Manifesto | Scrum Principles - Scrum Alliance. *Home - Scrum Alliance* [online]. SCRUM ALLIANCE®, Inc., 2016 [cit. 2016-03-01]. Dostupné z: <https://www.scrumalliance.org/why-scrum/core-scrum-values-roles>
- [18] SCHWABER, Ken a Jeff SUTHERLAND. *The Scrum Guide™: The Definitive Guide to Scrum: The Rules of the Game* [online]. 2014, 2013, 16 [cit. 2016-03-01]. Dostupné z: <http://www.scrumguides.org/docs/scrumguide/v1/scrum-guide-us.pdf>
- [19] Sprint Review Meeting - International Scrum Institute. *SCRUM-INSTITUTE.ORG - USD 29 ACCREDITED SCRUM CERTIFICATIONS* [online]. International Scrum Institute, 2016 [cit. 2016-03-01]. Dostupné z: http://www.scrum-institute.org/Sprint_Review_Meeting.php
- [20] Step 1: Get Your Backlog In Order! | All About Agile. *All About Agile | Agile Development Made Easy!* [online]. 2007 [cit. 2016-03-01]. Dostupné z: <http://www.allaboutagile.com/how-to-implement-scrum-in-10-easy-steps-step-1-get-your-backlog-in-order/>
- [21] Task in Scrum Definition - Innolution. *Home of Corporate On-site Scrum & Agile Training & Coaching from Author Ken Rubin - Innolution* [online]. Innolution, LLC., 2016 [cit. 2016-03-01]. Dostupné z: <http://www.innolution.com/resources/glossary/task>
- [22] The Daily Scrum Meeting. *Scrum Training Agile Training from ScrumMaster Mike Cohn* [online]. Broomfield: Mountain Goat Software, 2016 [cit. 2016-03-01]. Dostupné z: <https://www.mountaingoatsoftware.com/agile/scrum/daily-scrum>
- [23] The ScrumMaster. *Scrum Training Agile Training from ScrumMaster Mike Cohn* [online]. Broomfield: Mountain Goat Software, 2016 [cit. 2016-03-01]. Dostupné z: <https://www.mountaingoatsoftware.com/agile/scrum/scrummaster>

- [24] User Stories: An Agile Introduction. *Agile Modeling (AM) Home Page: Effective Practices for Modeling and Documentation* [online]. Ambyssoft Inc., 2014 [cit. 2016-03-01]. Dostupné z: <http://www.agilemodeling.com/artifacts/userStory.htm>
- [25] *Veřejný rejstřík a Sbirka listin - Ministerstvo spravedlnosti České republiky* [online]. Praha: Ministerstvo spravedlnosti, 2015 [cit. 2016-03-16]. Dostupné z: <https://or.justice.cz/ias/ui/rejstrik-firma.vysledky?subjektId=218437&typ=PLATNY>
- [26] Visual AGILExicon® - Innolution. *Home of Corporate On-site Scrum & Agile Training & Coaching from Author Ken Rubin - Innolution* [online]. Innolution, LLC., 2016 [cit. 2016-03-01]. Dostupné z: <http://www.innolution.com/resources/visual-agilexicon>
- [27] What is user story? - Definition from WhatIs.com. *Software Quality information, news and tips - SearchSoftwareQuality* [online]. TechTarget, 2015 [cit. 2016-03-01]. Dostupné z: <http://searchsoftwarequality.techtarget.com/definition/user-story>