

**Vysoká škola ekonomická v Praze**

**Fakulta informatiky a statistiky**

**Katedra informačních technologií**

Študijný program: Aplikovaná informatika

Odbor: Informační systémy a technologie

**Zvyšovanie vyspelosti procesov  
testovania v agilných projektoch –  
kvalitatívny empirický výskum**

**DIPLOMOVÁ PRÁCA**

Študent : Bc. Martin Homola

Vedúci : Ing. et Ing. Michal Doležel

Oponent : doc. Ing. Alena Buchalcevová, Ph.D.

**2017**

## **Prehlásenie:**

Prehlasujem, že som diplomovú prácu spracoval samostatne a že som uviedol všetky použité pramene a literatúru z ktorej som čerpal.

V Prahe dňa 23. apríla 2017

.....

Bc. Martin Homola

## **Pod'akovanie**

Vrelé ďakujem za odborné rady, pevné nervy, priateľský a ústretový prístup patrí pánovi Ing. et Ing. Michalovi Doleželovi, ktorý mi pomohol preniknúť do sveta agilného testovania a jeho krás.

Ďakujem taktiež každému členovi jednotlivých tímov, ktorý sa ochotne zúčastnil prípadovej štúdie za jeho čas, úprimnosť, ústretovosť a otvorenosť.

## **Abstrakt**

Diplomová práca je zameraná na analýzu vývoja procesu testovania z povedomia jednotlivých členov agilných projektových tímov v skúmaných spoločnostiach. Analýza je podkladom pre identifikáciu problémových oblastí a podoblastí jednotlivých tímov a spoločných znakov vývoja procesu testovania. Následne sú na spoločné problémové oblasti a podoblasti navrhnuté zlepšenia zo skúseností autora, odbornej literatúry a návrhov členov tímov. Získané poznatky sú generalizované a je vytvorený model oblastí ovplyvňujúcich vyspelosť testovania v agilne riadených projektoch.

Práca je rozdelená do štyroch častí. Prvá časť obsahuje definovanie základných pojmov skúmanej oblasti. Približuje agilnú vyspelosť, agilnú vyspelosť testovania a podrobne vysvetľuje pojem agilného testovania. Približuje agilný tím a rolu testera v ňom, dôvod procesu testovania, faktory jeho úspešnosti, praktiky agilného testovania a typy testov.

V druhej časti sú priblížené použité metódy a priebeh výskumu. Tretia časť zobrazuje poznatky výskumu. Popisuje skúmané tímy, ich organizačný kontext, vývoj testovacieho procesu, problémy tímu, spoločné problémy s návrhmi na ich zlepšenie, spoločné znaky vývoja testovacieho procesu a model oblastí ovplyvňujúcich vyspelosť agilného testovania. Na konci tejto časti sa nachádzajú výskumné poznatky o dôležitosti testerov v agilných tímoch. Poslednými časťami sú diskusia a záver, kde autor vysvetľuje prepojenie poznatkov výskumu s teóriou, hodnotí úspešnosť naplnenia cieľov, prínosy práce a navrhuje varianty rozšírenia práce.

## **Kľúčové slová**

agilné testovanie, agilný tím, agilný tester, praktiky agilného testovania, vyspelosť agilného testovania, zlepšovanie procesu testovania, dôležitosť testerov

## **Abstract**

Diploma thesis is focused on the analysis of testing process development derived from agile project team members awareness. This analysis is the foundation for individual problem areas and sub-areas identification and also common areas for the testing process development of researched teams. Afterwards, improvement plans for these areas are proposed from the authors experience, literature and the team members input. The research findings are generalised and model areas that influence agile testing maturity is created.

This thesis consists of four parts. The first part contains basic definition of the researched area terms, and explains agile maturity, agile testing maturity and agile testing in detail. It defines what the agile team is and the testers role in it. It also contains the purpose of testing, testing success factors, agile testing practices and agile test types.

In the second part, the research methods and the research process is explained. The third part contains the outputs of the research. It describes research teams, their organisational context, their testing process development, team problems, common problems and improvement plans for these problems. It also contains the common areas of the testing process development and model of areas that influence agile testing maturity. The final part of this section contains the research outputs about the importance of testers in agile teams. A discussion and conclusion are at the end that contain connections between the theory and the research findings and evaluates the success of determined goals, the thesis outputs and the proposals of future work

## **Keywords**

agile testing, agile team, agile tester, agile testing practices, agile testing maturity, testing process improvement, importance of testers

# Obsah

<b>1</b>	<b>ÚVOD 1</b>	
1.1	Ciele práce.....	2
1.2	Cieľová skupina .....	2
1.3	Použité metódy .....	2
1.4	Štruktúra práce .....	2
1.5	Komentovaná rešerš informačných zdrojov .....	3
<b>2</b>	<b>TEÓRIA .....</b>	<b>7</b>
2.1	Agilné testovanie .....	7
2.1.1	Agilný tím .....	10
2.1.2	Rola testera v agilnom tíme .....	11
2.1.3	Kategorizácia testov v agilnom vývoji.....	13
2.1.4	Kľúčové faktory úspešnosti .....	15
2.1.5	Dôvod procesu testovania.....	18
2.1.6	Praktiky budovania istoty .....	20
2.2	Vypelosť agilného vývoja.....	22
2.3	Vypelosť agilného testovania .....	23
<b>3</b>	<b>METÓDY.....</b>	<b>24</b>
3.1	Výskumné metódy .....	25
3.2	Tvorba otázok rozhovoru .....	27
3.3	Príprava a priebeh rozhovorov .....	28
3.4	Spracovanie dát .....	29
3.5	Analýza dát.....	30
<b>4</b>	<b>POZNATKY VÝSKUMU.....</b>	<b>31</b>
4.1	Predstavenie tímu 1 .....	32
4.1.1	Organizačné prostredie .....	32
4.1.2	Identifikácia tímu .....	32
4.2	Predstavenie tímu 2 .....	33
4.2.1	Organizačné prostredie .....	33
4.2.2	Identifikácia tímu .....	34
4.3	Predstavenie tímu 3.....	34
4.3.1	Organizačné prostredie.....	34
4.3.2	Identifikácia tímu .....	35
4.4	Predstavenie tímu 4.....	35
4.4.1	Organizačné prostredie.....	35
4.4.2	Identifikácia tímu .....	36

---

<b>4.5 Predstavenie tímu 5</b> .....	<b>36</b>
4.5.1 Organizačné prostredie.....	36
4.5.2 Identifikácia tímu.....	37
<b>4.6 Vývoj testovacieho procesu v tímoch</b> .....	<b>38</b>
4.6.1 Vývoj testovacieho procesu tímu 1.....	38
4.6.2 Vývoj testovacieho procesu tímu 2.....	40
4.6.3 Vývoj testovacieho procesu tímu 3.....	41
4.6.4 Vývoj testovacieho procesu tímu 4.....	42
4.6.5 Vývoj testovacieho procesu tímu 5.....	42
4.6.6 Spoločné znaky testovacieho procesu.....	44
<b>4.7 Identifikácia problémov jednotlivých tímov</b> .....	<b>48</b>
4.7.1 Identifikované problémy tímu 1.....	49
4.7.2 Identifikované problémy tímu 2.....	50
4.7.3 Identifikované problémy tímu 3.....	51
4.7.4 Identifikované problémy tímu 4.....	52
4.7.5 Identifikované problémy tímu 5.....	53
<b>4.8 Namapovanie spoločných problémov</b> .....	<b>54</b>
<b>4.9 Návrh na zlepšenie namapovaných problémov</b> .....	<b>56</b>
4.9.1 Automatizácia.....	57
4.9.2 Spolupráca a komunikácia.....	61
4.9.3 Projektové obmedzenia.....	65
4.9.4 Organizačné obmedzenia.....	67
4.9.5 Proces agilného vývoja.....	70
4.9.6 Myslenie člena tímu.....	73
4.9.7 Schopnosti člena tímu.....	75
4.9.8 Zdieľanie znalostí.....	76
4.9.9 Pracovný postup zlepšenia problémovej oblasti.....	78
<b>4.11 Model oblastí ovplyvňujúcich vyspelosť agilného testovania</b> .....	<b>79</b>
<b>4.12 Spoločné znaky prínosu testera v agilnom tíme</b> .....	<b>82</b>
<b>5 DISKUSIA</b> .....	<b>83</b>
<b>6 Záver</b> 86	
6.1 Zodpovedanie výskumných otázok.....	86
6.2 Prínosy práce a naplnenie cieľov.....	87
6.3 Varianty rozšírenia práce.....	88

---

<b>Príloha A: Otázky na skúmanie vyspelosti agilného testovania .....</b>	<b>90</b>
<b>Príloha B: Popis problémových oblastí a podoblastí.....</b>	<b>92</b>
<b>Príloha C: Formulár informovaného súhlasu .....</b>	<b>95</b>
<b>Príloha D: Výstup z analýzy dát v MAXQDA Analytics Pro.....</b>	<b>96</b>
<b>Terminologický slovník.....</b>	<b>100</b>
<b>Bibliografia.....</b>	<b>104</b>



---

## Zoznam obrázkov

Obrázok 1 – Štruktúra práce (autor)	2
Obrázok 2 – Znalosť v tvare T (Guest 1991) preložené autorom	10
Obrázok 3 – Agilné testovacie kvadranty (Crispin a Gregory 2008)	13
Obrázok 4 - Pyramída automatizácie testov (Gregory a Crispin 2014)	16
Obrázok 5 - Piliere testovania (Evans v Gregory a Crispin 2014)	18
Obrázok 6 – Postup výskumníka (autor)	24
Obrázok 7 – Porovnanie priebehu kvantitatívneho a kvalitatívneho výskumu Hendla (2016) preložené autorom	26
Obrázok 8 – Postup zakotvenej teórie Hendla (2016) preložené autorom	27
Obrázok 9 – Indikátor postupu výskumu (autor)	31
Obrázok 10 – Indikátor postupu výskumu krok 1 (autor)	38
Obrázok 11 – Indikátor postupu výskumu krok 2 (autor)	48
Obrázok 12 – Problémové oblasti tímu 1 (autor)	49
Obrázok 13 – Problémové oblasti tímu 2 (autor)	50
Obrázok 14 – Problémové oblasti tímu 3 (autor)	51
Obrázok 15 – Problémové oblasti tímu 4 (autor)	52
Obrázok 16 – Problémové oblasti tímu 5 (autor)	53
Obrázok 17 – Indikátor postupu výskumu krok 3 (autor)	54
Obrázok 18 – Individuálne problémové podoblasti skúmaných tímov (autor)	55
Obrázok 19 – Indikátor postupu výskumu krok 4 (autor)	56
Obrázok 20 – Indikátor postupu výskumu krok 5 (autor)	79
Obrázok 21 - Model oblastí ovplyvňujúcich vyspelosť agilného testovania (autor)	81
Obrázok 22 – Počet okódovaných segmentov podľa skúmaných tímov (autor)	96
Obrázok 23 – Prehľad kategórií prvej úrovne (autor)	97
Obrázok 24 – Prehľad kategórie budúcnosť (autor)	98
Obrázok 25 – Prehľad kategórií súčasnosť a minulosť (autor)	99

---

## Zoznam tabuliek

Tabuľka 1 – Výsledky vyhľadávania zdrojov (autor)	4
Tabuľka 2 – Porovnanie vybraných aspektov tradičných a agilných metodík (Buchalcevo <span>á</span> 2005) a (Bureš et al. 2016)	8
Tabuľka 3 – Časová náročnosť transkripcie (autor)	29
Tabuľka 4 – Špecifikácia opýtaných členov tímu 1 (autor)	33
Tabuľka 5 – Špecifikácia opýtaných členov tímu 2 (autor)	34
Tabuľka 6 – Špecifikácia opýtaných členov tímu 3 (autor)	35
Tabuľka 7 – Špecifikácia opýtaných členov tímu 4 (autor)	36
Tabuľka 8 – Špecifikácia opýtaných členov tímu 5 (autor)	37
Tabuľka 9 – Mapovanie spoločných problémových oblastí (autor)	54
Tabuľka 10 – Návrhy zlepšenia podoblastí oblasti Automatizácia (autor)	57
Tabuľka 11 – Návrhy zlepšenia podoblastí oblasti Spolupráca a komunikácia (autor)	61
Tabuľka 12 – Návrhy zlepšenia podoblastí oblasti Projektové obmedzenia (autor)	65
Tabuľka 13 – Návrhy zlepšenia podoblastí oblasti Organizačné obmedzenia (autor)	67
Tabuľka 14 – Návrhy zlepšenia podoblastí oblasti Proces agilného vývoja (autor)	70
Tabuľka 15 – Návrhy zlepšenia podoblastí oblasti Myslenie člena tímu (autor)	73
Tabuľka 16 – Návrhy zlepšenia podoblastí oblasti Schopnosti člena tímu (autor)	75
Tabuľka 17 – Návrhy zlepšenia podoblastí oblasti Zdieľanie znalostí (autor)	76
Tabuľka 18 – Mapovanie spoločných oblastí a znakov na prvky modelu (autor)	80
Tabuľka 19 – Spoločné znaky prínosu testera v agilnom tíme (autor)	82
Tabuľka 20 – Prepojenie výsledkov výskumu vyspelosti agilného testovania s teóriou (autor)	84
Tabuľka 21 – Spoločné znaky prínosu testera v agilnom tíme (autor)	85
Tabuľka 22 – Prínosy práce - výstupy z kvalitatívnych dát (autor)	87
Tabuľka 23 – Prínosy - metodické postupy (autor)	88
Tabuľka 24 – Popis problémových oblastí skúmaných tímov (autor)	92
Tabuľka 25 – Popis problémových podoblastí skúmaných tímov (autor)	93

# 1 ÚVOD

Od vzniku agilného manifestu prešiel nejaký čas a dnes už veľa spoločností využíva agilné praktiky vo svojej praxi z dôvodu využitia ich výhod a rýchlej reakcie na meniace sa požiadavky zákazníkov. Zákazníci požadujú kvalitný produkt, no agilný manifest nedefinuje akým spôsobom túto kvalitu dosiahnuť. Keďže som v súčasnosti členom agilného tímu a mojou úlohou je radiť v otázkach kvality, zaujímam sa o to, aké zlepšenia je potrebné uskutočniť, aby sme ju spolu s tímom dokázali zvýšiť.

Riadenie kvality je neodmysliteľnou súčasťou každého softvérového projektu a zaručuje, aby bol zákazníkovi dodaný produkt s ktorým bude spokojný. V prípade zlého riadenia kvality hrozí spoločnosti nielen zvýšenie nákladov na opravy chýb, ale taktiež strata zákazníka a poškodenie jej dobrého mena. Preto spoločnosti zistili, že investovať do tohto procesu je veľmi dôležité. Jednou z častí, ktorá ovplyvňuje kvalitu je i testovací proces. Ja zastávam v tíme pozíciu test inžiniera spolu s ďalšími kolegami a zaujímame sa o zlepšenie našich aktuálnych procesov testovania, keďže v nich stále objavujeme určité možnosti zlepšenia. Agilné testovanie je však odlišné od tradičného vodopádového v spôsobe vykonávania testovacích techník, kedy je za testovanie zodpovedný celý tím a je potrebná vyššia miera zapojenia a spolupráce medzi testerami, vývojármi i zákazníkmi. Táto odlišnosť ma núti pozerat' sa na spôsoby zlepšovania procesov testovania z iného uhlu ako u tradičných vodopádových metodík. V súčasnosti existuje mnoho modelov agilnej vyspelosti, ktoré hovoria o zlepšovaní agilných praktík. Málo z nich sa však zaoberá práve procesom testovania. To bol popud, ktorý ma prinútil zamyslieť sa nad tým, ako vieme definovať vyspelosť agilného testovania.

V kapitole 1 sú definované ciele práce, cieľová skupina, štruktúra práca a komentovaná rešerš informačných zdrojov. Kapitola 2 približuje základné pojmy skúmanej problematiky akými sú agilné testovanie, agilná vyspelosť, vyspelosť agilného testovania. Pojem agilné testovanie je podrobne rozpracovaný i z pohľadu testovacích praktík, ktoré sa pri agilnom vývoji používajú a tiež približuje pohľad na rolu testera v agilnom tíme. V kapitole 3 sú následne popísané metódy spolu s postupom krokov, ktoré som pri uskutočnení výskumu nasledoval. Kapitola 4 obsahuje identifikáciu skúmaných tímov, približuje analýzu vývoja procesu testovania, problémových oblastí a podoblastí pre každý tím. Tieto oblasti sú následne mapované na spoločné problémové oblasti. Z výstupov analýzy je zostavený návrh riešení na každú spoločnú problémovú podoblasť a následne je zo zistených poznatkov vytvorený model oblastí ovplyvňujúcich vyspelosť agilného testovania. Taktiež je skúmaná otázka dôležitosti testerov v agilnom tíme, ktorá je zodpovedaná vo výskumnej časti prostredníctvom vytvorenia prínosov agilného testera. V kapitole 5 a 6 je diskutovaná zhoda zistených poznatkov s teoretickými poznatkami, naplnenie cieľov práce spolu s hodnotením prínosov a možnosti rozšírenia práce.

## 1.1 Ciele práce

Hlavným cieľom tejto práce je nájsť spôsob, akým sa dajú zlepšiť procesy testovania v agilne riadených projektoch vývoja softvéru. Čiastkovým cieľom je identifikácia silných a slabých stránok aspoň v jednom projekte zaoberajúcom sa agilným vývojom. Na základe tejto identifikácie je cieľom navrhnúť zlepšenia procesov testovania vo vymedzených oblastiach. Ďalším cieľom je zo zistených poznatkov vytvoriť metodický postup, alebo model, pre zlepšenie procesov testovania v agilných projektoch.

## 1.2 Cieľová skupina

Výstupy diplomovej práce majú pomôcť agilným tímom a ich manažérom so zlepšením procesov testovania vo svojich projektoch. Zároveň môžu poskytnúť čitateľovi nový pohľad na vyspelosť agilného testovania, ktorý môže aplikovať pri svojej práci. Primárne je však určená pracovníkom zaoberajúcim sa vývojom softvéru za použitia agilných praktík.

## 1.3 Použité metódy

Hlavnou metódou ktorú v práci používam je kvalitatívny výskum a za stratégiu kvalitatívneho výskumu som vybral zakotvenú teóriu (grounded theory), ktorá obsahuje procedúry na vytvorenie novej alebo málo spracovanej teórie. Cieľom zakotvenej teórie je generovanie teórie namiesto testovania a overovania existujúcej teórie. Táto stratégia má pôvod v sociológii, ale stále viac preniká i do ďalších disciplín, vrátane informatiky. Keďže zvyklosťou v disciplínach využívajúcich kvalitatívne metódy výskumu je odlišná štruktúra práce, uvádzam popis metód ako samostatnú kapitolu 3.

## 1.4 Štruktúra práce

Práca je štrukturovaná podľa používaných štruktúr v disciplínach zaoberajúcich sa sociálnym výskumom a je členená nasledovným spôsobom:



Obrázok 1 – Štruktúra práce (autor)

## 1.5 Komentovaná rešerš informačných zdrojov

Táto podkapitola poskytuje prehľad o informačných zdrojoch, ktoré som v mojej diplomovej práci použil a ktoré sa zaoberali podobnou či príbuznou témou. Pri vyhľadávaní týchto zdrojov som postupoval nasledujúcim spôsobom. Na začiatku som si definoval kľúčové slová vychádzajúce z cieľov mojej práce. Tieto kľúčové slová som zostavil v anglickom a českom jazyku kvôli širšiemu priestoru vyhľadávania. Prehľadovaný zoznam kľúčových slov bol nasledovný:

1. Agile maturity / Agilní vyspělost
2. Agile maturity model / Model agilní vyspělosti
3. Agile testing / Agilní testování
4. Agile testing maturity / Vyspělost agilního testování
5. Agile process maturity / Vyspělost agilního procesu
6. Maturity in agile testing / Vyspělost v agilním testování
7. Testing maturity / Vyspělost testování

Po definícii kľúčových slov nasledoval výber najznámejších elektronických databáz informačných zdrojov. Vybrané boli databáze:

1. **ProQuest Central** – megadatabáza, ktorá obsahuje plné texty novín a časopisov, dizertačných prác, tržné správy, profily firiem, väčšinou v anglickom jazyku.
2. **EBSCO** – platforma pre prístup k databázam článkov z oblasti ekonomiky, managementu, účtovníctva, financií, obchodu, humanitných a sociálnych vied, vedy a kultúry, zdravotníctva, väčšinou v anglickom jazyku.
3. **Theses.cz** – národný register záverečných prác slúžiaci školám v Českej a Slovenskej republike na odhaľovanie plagiátov prevažne v českom a slovenskom jazyku.
4. **Google Scholar** – vyhľadávač odbornej literatúry prevádzkovaný spoločnosťou Google, ktorá prostredníctvom služby Google Search predstavuje najpoužívanejší vyhľadávač na svete. Prehľadávajú sa články, odborné práce, abstrakty, vedecké spoločnosti, online repozitáre, univerzity a ďalšie stránky. Väčšina výsledkov je v anglickom jazyku.
5. **Forrester** – spoločnosť poskytujúca prieskumy v oblasti ekonomiky a technológií v anglickom jazyku.

Vo vybraných databázach som následne uskutočnil vyhľadávanie definovaných kľúčových slov, ktoré prebiehalo v troch fázach. Prvou fázou bolo vyhľadávanie bez zadania akýchkoľvek dodatočných vyhľadávacích kritérií. Vo výsledkoch vyhľadávania uvedených v Tabuľke 1 reprezentuje túto fázu prvé číslo pred lomítkom. Po uskutočnení tejto fázy som zistil, že vo výsledkoch vyhľadávania sa neobjavil iba celý hľadaný výraz, ale i samostatné slová, ktoré výraz tvorili, čím som získal veľa nerelevantných výsledkov.

Druhá fáza zahŕňala preto vyhľadávanie iba celého výrazu. Vo výsledkoch vyhľadávania uvedených v Tabuľke 1 reprezentuje túto fázu druhé číslo za lomítkom. U niektorých obecných kľúčových slov bolo však množstvo výsledkov vyhľadávania stále veľké. Uskutočnil som preto tretiu, dodatočnú fázu vyhľadávania, kde som zúžil výsledky druhej fázy. Vyhľadávacie kritériá som zmenil na vyhľadávanie zdrojov nie starších ako rok 2015. V prípade, že výsledkov bolo stále veľa, zvýšil som hranicu na rok 2016. Vo výsledkoch vyhľadávania uvedených v Tabuľke 1 reprezentuje túto fázu tretie číslo za lomítkom s indikátorom časového obmedzenia v zátvorke.

Podfarbením jednotlivých riadkov tabuľky som rozlíšil anglické a české kľúčové slová pre jednoduchšiu orientáciu a porovnávanie. Slabo modrá farba zobrazuje anglické kľúčové slovo a biela farba české kľúčové slovo.

**Tabuľka 1 – Výsledky vyhľadávania zdrojov (autor)**

Kľúčové slovo	Databáza	ProQuest	EBSCO	Theses.cz	Google Scholar	Forrester
Agile maturity		12 870 / 28	39 / 7	199 / 2	41 900 / 271 / 47 (2016)	908 / 24
Agilní vyspělost		0 / 0	0 / 0	254 / 0	117 / 0	0 / 0
Agile maturity model		8 992 / 5	25 / 3	181 / 0	34 400 / 167 / 32 (2016)	725 / 0
Model agilní vyspělosti		0 / 0	0 / 0	232 / 0	5 / 0	0 / 0
Agile testing		34 714 / 300 / 67 (2016)	173 / 46	469 / 24	137 000 / 1420 / 162 (2016)	1021 / 55
Agilní testování		0 / 0	0 / 0	895 / 7	535 / 1	0 / 0
Agile testing maturity		5 551 / 0	927 / 0	10 / 0	29 200 / 8	320 / 2
Vyspělost agilního testování		0 / 0	0 / 0	167 / 0	74 / 0	0 / 0
Agile process maturity		9 832 / 4	2 / 0	165 / 1	38 400 / 26	862 / 0
Vyspělost agilního procesu		0 / 0	0 / 0	235 / 0	124 / 0	0 / 0
Maturity in agile testing		5 551 / 0	927 / 0	109 / 0	29 300 / 0	320 / 2
Vyspělost v agilním testování		0 / 0	0 / 0	165 / 0	72 / 0	0 / 0
Testing maturity		173 198 / 77 / 10 (2015)	235 / 10	922 / 6	1 040 000 / 758 / 41 (2016)	926 / 6
Vyspělost testování		0 / 0	0 / 0	962 / 1	3 920 / 0	0 / 0

Z Tabuľky 1 je zrejmé, že najviac výsledkov bolo vyhľadanych v anglickom jazyku. V českom jazyku sa kľúčovým slovám venovali len vysokoškolské práce.

Obecnejšie kľúčové slová ako Agile testing, Testing maturity, Agile maturity a Agile maturity model poskytli viac výsledkov ako konkrétnejšie kľúčové slová, ktorými boli Agile process maturity, Agile testing maturity a Maturity in agile testing.

### **Zistenia, ktoré upravili vyhľadane počty zdrojov uvedených v Tabuľke 1:**

- Pri tvorbe kľúčových slov bolo zvolené české slovo „vyspělost“ namiesto „zralost“ kvôli vyššiemu počtu vyhľadanych zdrojov.
- Prehľadované boli i zdroje článkov, pričom články sa nezaoberali kľúčovým slovom. Z tohto dôvodu sa mi zúžil počet relevantných výsledkov.
- U výskumov od Forresteru a vyhľadávaní cez Google Scholar boli do vyhľadávania zaradené všetky články z tej istej publikácie, pričom len jeden sa zaoberal daným kľúčovým slovom. Tento dôvod taktiež viedol k zúženiu počtu relevantných výsledkov.
- Pri vyhľadávaní kľúčového slova ako Agile maturity boli do výsledkov vyhľadávania zaradené i kľúčové slová ako Agile maturity model. Ďalší dôvod ktorý viedol k zúženiu počtu relevantných výsledkov.

### **Rešerš informačných zdrojov:**

Knihy *Agile testing* (Crispin a Gregory 2008) a *More agile testing* (Gregory a Crispin 2014) od autoriek Janet Gregoryovej a Lisy Crispinovej poskytujú zosumarizované najlepšie praktiky z disciplíny agilného testovania. Knihy slúžia k rozšíreniu testovacích znalostí, pomáhajú s prispôbením agilného testovania na prostredie v ktorom tím pôsobí a tiež zlepšeniu testovacích procesov a tímového zapojenia.

Diplomová práca (Mrázek 2014) sa zaoberá analýzou procesov testovania IT oddelenia medzinárodnej finančnej inštitúcie. Na základe tejto analýzy autor identifikuje problémové oblasti IT oddelenia a následne navrhuje kroky pre zefektívnenie procesov testovania. Podobná odborná práca od Kučeru (2007) sa sústreďuje na optimalizáciu testovacích procesov použitím modelu CMMI z ktorého sú vyhľadane oblasti a doporučenia na problematické procesy testovania.

Bakalárska práca (Blažková 2013) definuje a popisuje agilné testovanie a prácu agilného testera. Porovnáva tiež tradičné (rigorózne) metodiky s metodikami agilnými. Následne navrhuje rozšírenie systému na hodnotenie metodík METES o kritériá pokrývajúce testovací proces. Práca má slúžiť ako príručka pre začínajúcich agilných testerov.

V diplomovej práci (Vachalec 2013) sa autor zaoberá riadením kvality a testovania pri vývoji softvéru. Popisuje metriky, ktoré sa pri riadení kvality využívajú. Sú analyzované rozdiely v zaistovaní kvality medzi tradičnými a agilnými metodikami a navrhnuté kritériá

podľa ktorých sa metodiky vo vzťahu ku kvalite dajú porovnávať. Vysvetľuje tiež pojem inžiniera kvality a rozširuje metodiku MMSP (Metodika pre Malé Softwarové Projekty) v oblasti testovania. Rojo (2014) vo svojej bakalárskej práci približuje testovanie v agilných projektoch a tiež rozširuje metodiku MMSP o skúmanú oblasť.

Králová (2012) vo svojej diplomovej práci navrhuje lokalizovanú metodiku testovania pre podľa medzinárodných praktík a štandardov podľa ISTQB určenú pre české organizácie. Metodika je navrhnutá v nástroji EPF Composer a slúži na definovanie procesov testovania pre rôzne organizácie. Autorka metodiku hodnotí na základe štandardizovaného procesného rámca pre zlepšovanie procesov testovania a navrhuje oblasti pre ďalšie rozšírenie metodiky. V teoretickej časti uvádza prehľad štandardov, ktoré majú vzťah k testovaniu, objasňuje účely certifikácie v oblasti testovania a analyzuje možnosti využitia materiálov pre prípravu na certifikáciu ISTQB.

Došek (2012) sa zameriava v diplomovej práci na model Test Maturity Model Integration (TMMi). Hľadá možnosti a obmedzenia jeho využitia a zhŕňa existujúce skúsenosti s modelom. V práci autor navrhuje praktický postup pre ohodnotenie procesov podľa modelu. Okrem TMMi obsahuje úvode do modelov Test Process Improvement (TPI), Capability Maturity Model Integration (CMMI) a Critical Testing Processes (CTP). V práci sú všetky štyri modely hodnotené a sú navrhnuté situácie v ktorých je jednotlivé modely vhodné použiť.

Diplomová práca (Chodura 2012) sa zaoberá agilným prístupom k vývoju a pozíciou testovania podľa Lean Software Development. Na začiatku identifikuje problémy a nevýhody testovacích procesov v tíme na ktoré následne aplikuje lean princípy, poukazuje na ich pozitívne vlastnosti a hodnotí výhody, ktoré tímu tento prístup priniesol.

Kulhánková (2015) popisuje rozdiely medzi tradičným a agilným testovaním. V praktickej časti sa venuje vytvoreniu vlastnej metodiky na základe prístupu Behaviour Driven Development, ktorá pomáha riešiť nedostatky a bariéry metodiky Scrum vo veľkých podnikoch.

Z odborných článkov sú dôležité komparačné štúdie modelov agilnej vyspelosti, ktoré porovnávajú vybrané agilné modely podľa autormi zostavených kritérií a hodnotia ich úspešnosť. Ďalšie prehľadované články a knihy prinášajú pohľad na agilné testovanie, agilnú vyspelosť a vyspelosť agilného testovania. Poznatky z týchto zdrojov sú spracované v kapitole 2. Žiadna z českých alebo slovenských odborných prác sa však nezaobrá témou vyspelosti agilného testovania, preto som sa rozhodol túto problematiku skúmať.



## 2 TEÓRIA

V tejto kapitole definujem pojmy zostavené na základe uskutočnenej rešerše zdrojov. Tieto pojmy sú základom pre zostavenie výskumu a venujú sa skúmanej problematike. Kapitola čerpá hlavne zo zdrojov (Crispin a Gregory 2008) a (Gregory a Crispin 2014), preto tieto citácie v ďalšom texte už neuvádzam ak sa nejedná o doslovnú citáciu. V prípade, že text pochádza z iného zdroja používam citáciu tohto zdroja.

### 2.1 Agilné testovanie

Crispin definuje agilné testovanie porovnaním tradičného vodopádového vývoja s agilným. V tradičných metodikách testovanie čaká na ukončenie fáze vývoja. Podľa autorky je naopak agilný testing iteratívny a inkrementálny. To znamená, že testery testujú každý prírastok kódu, akonáhle je dokončený. Iterácia môže trvať týždeň až mesiac. Tím programuje a testuje časť kódu a uisťuje sa, že táto časť funguje správne. Potom sa presunie do ďalšej časti, ktorá musí byť naprogramovaná. Programátori sa nikdy nedostanú o krok pred testerov, pretože *užívateľský príbeh* (user story) nie je dokončený, dokým nie je otestovaný. User story alebo story je jednou z úrovní detailu produktu pri plánovaní iterácie, ktorá predstavuje malý, testovateľný kus požadovanej funkcionality a môže byť dokončený v priebehu jedného až troch dní.

Autorka tiež definuje princípy agilného testovania, ktoré vychádzajú z princípov agilného manifestu (<http://agilemanifesto.org/>) a extrémneho programovania (<http://www.extremeprogramming.org/>). Sú nimi:

1. Poskytovanie nepretržitej spätnej väzby
2. Doručovanie hodnoty zákazníkovi
3. Umožnenie komunikácie tvárou v tvár
4. Mať odvahu
5. Zachovávať jednoduchosť
6. Neustále zlepšovanie
7. Reakcia na zmenu
8. Samo-organizácia
9. Sústredenosť na ľudí
10. Pôžitok z práce (Crispin a Gregory 2008, str. 21)

V agilnom testovaní sa väčšina akceptačného testovania uskutočňuje samotným tímom ako vyplýva z prieskumu (Ambler 2012). Nezaobídeme sa preto bez tímového prístupu pri ktorom je tím sám zodpovedný za testovanie.

Tvrdí to aj Linz (2014) podľa ktorého tu neexistuje test manažér a zodpovednosť za testovanie a vývoj je na celom produktovom tíme. Priblíženie agilného tímu obsahuje podkapitola 2.1.1. Testovanie však vyžaduje špecifické skúsenosti a preto je vhodné disponovať človekom, ktorý má prax v tejto oblasti a vie tak tím naučiť odskúšané testovacie praktiky.

Kulhánková (2015) tvrdí, že agilné testovanie neznamenaá zmenu spôsobu testovania z hľadiska rozsahu, ale z hľadiska spôsobu vykonávania rovnakých aktivít a snaží sa o čo najväčšiu paralalizáciu testovania a vývoja. Na objasnenie rozsahu testovania slúži podkapitola 2.1.3 Kategorizácie testov v agilnom vývoji.

Podľa Bureša a kolektívu (2016) agilné testovanie neexistuje, pretože sa v ňom používajú rovnaké metódy, techniky a koncepty ako v tradičných projektoch, len inak použité. Uvádza, že kvalita je celotímovým záujmom a predpokladá sa zdieľaná zodpovednosť. Taktiež sa zmiňuje o tom, že vytváranie akejsi metodiky agilného testovania je proti agilným praktikám a lepšie je sa zamerať na zdieľanie znalostí prostredníctvom komunit testerov. Podľa Doležela (2017) komunitou testerou rozumieme neformálnu sieť test profesionálov so záujmom o testovacie znalosti. Porovnanie niektorých aspektov tradičných a agilných metodík zobrazuje tabuľka 2.

**Tabuľka 2 – Porovnanie vybraných aspektov tradičných a agilných metodík (Buchalcevová 2005) a (Bureš et al. 2016)**

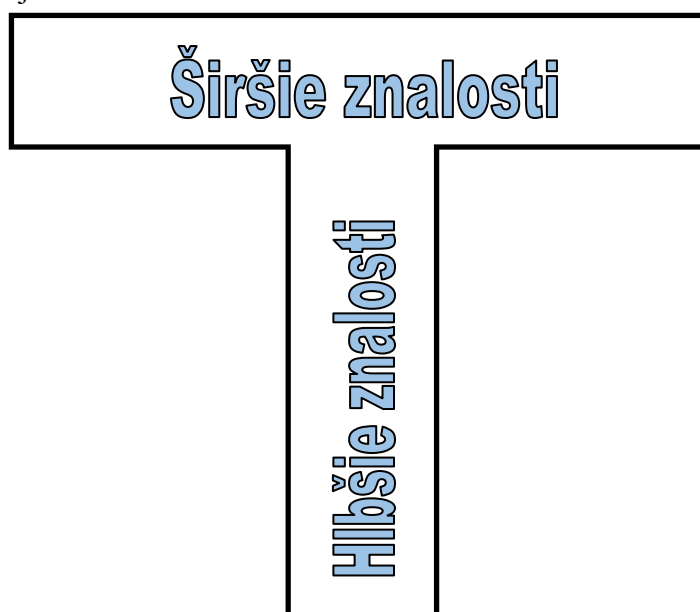
Hľadisko	Tradičné metodiky	Agilné metodiky
<b>Náplň metodiky</b>	Procesy, zameriavajú sa na explicitnú znalosť a pozerajú na ľudí ako na sekundárny faktor	Praktiky, zameriavajú sa na „tacit“ znalosti, chápu ľudí ako kľúčový faktor úspechu
<b>Podrobnosť metodiky</b>	Procesy a činnosti sú popísané veľmi podrobne	Zameriava na činnosti, ktoré vytvárajú hodnotu a eliminuje činnosti, ktoré hodnotu neprinášajú
<b>Kvalita</b>	Zameranie na kvalitu procesov a predpoklad, že kvalitné procesy povedú ku kvalitnému výsledku	Zameranie na hodnotu pre zákazníka a vysokú kvalitu produktu
<b>Predvídateľnosť</b>	Predpokladá predvídateľnosť budúcnosti, dôraz na anticipáciu	Predpokladá nepredvídateľnosť budúcnosti, dôraz na adaptáciu na zmeny

Hľadisko	Tradičné metodiky	Agilné metodiky
<b>Definovateľnosť procesu vývoja software</b>	Vývoj software je definovaný proces, je možné ho bez problémov opakovať	Vývoj software je empirický proces, nemôže byť konzistentne opakovaný, ale vyžaduje konštantné monitorovanie a adaptáciu
<b>Participácia zákazníka na projekte</b>	Len v začiatkových a koncových fázach	Zákazník je riadiacim subjektom behom celého projektu, pri každej iterácii zákazník môže meniť priority funkcií
<b>Štýl vedenia</b>	Tradičný štýl formovaný na základe nedôvery, direktívne vedenie, kontrola	Vodcovstvo a spolupráca, je formovaný na základe dôvery a rešpektu
<b>Forma komunikácie</b>	Prevažne písomná	Dôraz na komunikáciu tvárou v tvár
<b>Dokumentácia</b>	Rozsiahla dokumentácia	Rodstatná nie je dokumentácia, ale pochopenie
<b>Spôsob vývoja</b>	Skôr vodopádový, prípadne iteratívny a prírastkový s dlhými iteráciami	Prírastkový vývoj s krátkymi iteráciami
<b>Zdroj procesného know-how</b>	Štandardizácia, dobré praktiky v odvetví ako celku	Každý tím si nájde najlepšie vyhovujúci individuálny prístup, neformálne zdieľanie naprieč tímami v organizácii
<b>Úloha testerov</b>	„Strážcovia brány – ochrancovia užívateľov“	„Sprievodcovia programátorov v džungli kvality“
<b>Špecializácia testerov</b>	Špecialisti (manažér testovania, analytik testovania, tester, vývojár automatizovaných testov...)	Univerzalisti nazývaní napr. inžinieri testovania/kvality
<b>Externé vplyvy posilňujúce pozíciu</b>	Tradiície, Zákonné regulácie a audit (požiadavky na bezpečnosť a spoľahlivosť)	Nutnosť rýchlo reagovať na jednanie konkurencie

### 2.1.1 Agilný tím

Podľa Cottmeyera (2015) je agilný tím skupinou ľudí rôznych pracovných rolí, ktorá disponuje všetkými potrebnými znalosťami na vyprodukovanie fungujúceho a otestovaného produktového prírastku. Odborná literatúra aj praktici tvrdia, že v agilnom procese by sa mali jednotlivé pracovné role rozmazať a jednotlivé role by mali medzi sebou zdieľať znalosti. V prípade, že tím identifikuje potrebu znalosti, ktorá chýba na jeho efektívne fungovanie, je potrebné ju doplniť. Vo väčšine prípadov táto znalosť v tíme je a preto by mal byť každý člen aktívny aby ju vo vhodnom okamihu využil a zvýšil tak tímovú výkonnosť.

Učením sa znalostí od iných rolí bude každý člen tímu vnímať dôležitosť kvality a tím môže prispieť k zvýšeniu celkovej kvality vyvíjanej aplikácie. Takto sa obohacujú znalosti v tvare T každého člena tímu. Na to aby tím pracoval efektívne potrebujú členovia širšie znalosti v oblastiach na ktoré nie sú špecializovaní. Táto šírka znalostí umožňuje spoluprácu so špecialistami iných pracovných rolí. Zároveň však potrebujú hlbšie znalosti a prax vo svojej špecializácii čím môžu do tímu priniesť znalosti, ktoré v ňom chýbajú. Obrázok 2 zobrazuje znalosť v tvare T.



Obrázok 2 – Znalosť v tvare T (Guest 1991) preložené autorom

U testerov širšie znalosti predstavujú základné porozumenie systémovej architektúry, obecných konceptov programovania, dotazovania do databází a tiež schopnosť pracovať s vývojovými prostrediami a kontinuálnou integráciou, ktorá umožňuje často kontrolovať zmeny v kóde tvorbou automatických regresných testov. Na druhej strane, ostatné role potrebujú základné znalosti testovania, aby splnili požiadavok na celotímové testovanie. Hlbšie znalosti predstavujú u testerov testovacie praktiky a skúsenosti s testovaním. Niektoré role by mali mať tiež hlbšie znalosti o požiadavkách biznisu. Biznis v tomto pojatí predstavuje zákaznícky tím a je tak uvádzaný i v ďalšom texte.

## 2.1.2 Rola testera v agilnom tíme

Prínos agilného testera vieme zistiť z činností, ktoré vykonáva a aké znalosti pri tom používa. V agilnom testovaní sú na testera kladené jednak požiadavky na „mäkké znalosti“, ktoré predstavujú osobnostné atribúty človeka v interakcii s inými ľuďmi. Nevieme povedať, že by sme si ich osvojili naplno, pretože sa v nich stále zdokonaľujeme. Tester okrem nich však potrebuje i „tvrdé, technické znalosti“ na zaistenie dodania požadovaných funkcií. Gregory a Crispin (2014) uvádzajú nasledujúce kvality, ktoré do tímu jeho rola prináša.

### „Mäkké znalosti“ agilného testera:

- **facilitátor** – niekto kto prijme k spolupráci ľudí s odlišnými rolami pričom sa sústreďia na spoločný cieľ.
- **riešiteľ problémov** – pomáha tímu adresovať problémy tak, aby sa nesústreďil len na symptómy problémov, ale riešil ich príčiny.
- **poskytovanie a prijímanie spätnej väzby** – má poskytovať konštruktívnu spätnú väzbu, ktorá pomôže vybudovať dôveru v tíme. Pri prijímaní spätnej väzby sa má zamyslieť nad jej významom pred vytváraním si záverov.
- **znalosť biznis domény** – porozumenie biznisu na úrovni znalostí v tvare T, čím môžu biznisu lepšie pomôcť ohľadom budúcnosti produktu.
- **koučovacie a načúvacie schopnosti** – keďže každý v agilnom tíme je zodpovedný za testovanie, tester má viesť ostatných členov v riešení svojich vlastných problémov s testovaním a kvalitou. Mal by tiež načúvať sťažnostiam a novým nápadom.
- **odlišné myslenie** – mal by rozpoznať, ktorý štýl myslenia je vhodný v konkrétnej situácii a použiť ho aby vedel reflektovať realitu.
- **organizujúci** – vedieť zorganizovať si svoj čas a sústreďiť sa na úlohy.
- **kolaboratívny** – nebáť sa požiadať o pomoc s testovaním v prípade zahltenia testovacími úlohami

**„Tvrde znalosti“ agilného testera:**

- **riadenie vývoja príkladmi** – vytváranie automatických testov na príklady chovania novej funkcie definované biznisom.
- **automatizačné a kódovacie znalosti** – základná schopnosť čítať kód, porozumenie kódovacím štandardom, schopnosť rozoznať čo môže byť automatizované, obecná znalosť systémovej architektúry a iné.
- **obecné technické znalosti** – znalosti ako čítanie logov, základné dotazovanie do databází, práca s príkazovým riadkom a iné.
- **vývojové prostredia** – schopnosť aktualizácie, zostavenia buildu na lokálnej stanici pre flexibilnejšie testovanie a skúšanie programu.
- **testovacie prostredia** – tím by mal mať viac testovacích prostredí na rôzne účely a tester by mali poznať ich nastavenia. Mali by vedieť kedy požiadať o inú konfiguráciu alebo nový server.
- **kontinuálna integrácia a systém na správu kódu** – rozumieť konfiguráciám kontinuálnej integrácie, základnej práci so systémom na správu kódu a stratégii vetvenia.
- **atribúty kvality testovania** – schopnosť vedieť poradiť ohľadom zlepšenia atribútov kvality, ktoré sú dôležité, ale ťažko testovateľné ako výkonnosť, bezpečnosť. Taktiež sem patrí podpora exploratívneho testovania v tíme.
- **techniky testovacieho návrhu** – znalosť techník ktoré pomôžu s rozhodnutím čo a ako testovať.

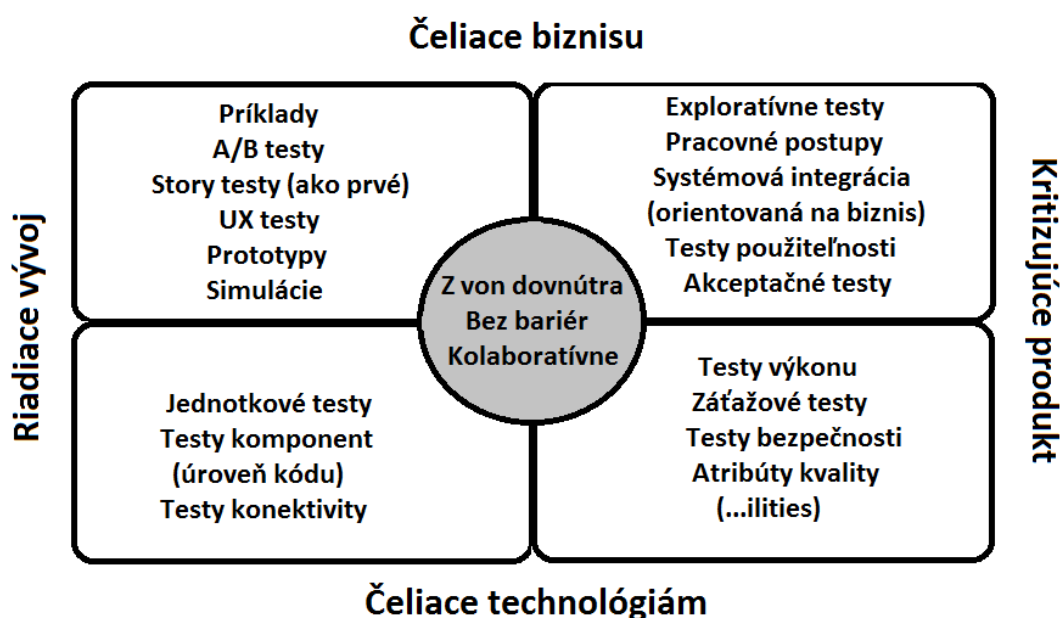
Spoločnosť vie pomôcť s podporou týchto znalostí tak, že môže vytvoriť komunitu testerov ku ktorej budú mať tester v nej pracujúci, ale i externí odborníci v tejto oblasti prístup. Mala by umožniť testerom zdieľať skúsenosti, názory, nápady. Spoluprácou testerov v spoločnosti môže táto platforma vzniknúť a postupne sa rozširovať i mimo spoločnosti.

Spomenuté znalosti kladú i dôraz na nové požiadavky na kandidáta zaujímajúceho sa o pozíciu agilného testera. V spoločnostiach, ktoré využívajú tradičné metodiky sú po testeroch požadované z veľkej časti technické znalosti, no väčšinu požiadaviek na agilného testera vieme zaradiť skôr medzi „mäkké znalosti“. Aj keď však nájdeme vhodného uchádzača, nemusí to znamenať, že tester bude úspešný, pretože tímová dynamika môže ovplyvniť jeho chovanie.

### 2.1.3 Kategorizácia testov v agilnom vývoji

Agilné testovanie má rovnaký rozsah ako testovanie v tradičných projektoch, pričom pridáva nové typy testov. V tejto podkapitole sa preto venujem zobrazeniu rozsahu testovania za pomoci agilných testovacích kvadrantov zobrazených na obrázku 3.

Na ľavej strane nájdeme testy, ktorými sa snažíme predchádzať chybám včasným testovaním. Z tohto pohľadu sa tu sústreďme na riadenie vývoja. Pravá strana skôr obsahuje testy, ktoré chyby odhaľujú a tiež tu objavujeme chýbajúce funkcie. Sústreďme sa tu teda na kritizovanie produktu. Testy čeliace biznisu nachádzajúce sa v horných kvadrantoch nám hovoria, aké funkcie biznis požaduje a čo je pre neho dôležité. Učíme sa nimi potreby biznisu, ktoré následne transformujeme na kód. Tieto testy sa ukazujú (čelia) biznisu a sú pre neho zrozumiteľné. Testy čeliace technológiám sú viac interné, nečelia biznisu a sú zrozumiteľné len členom tímu s technickými znalosťami.



Obrázok 3 – Agilné testovacie kvadranty (Crispin a Gregory 2008)

### **Testy čeliace technológiám, ktoré riadia vývoj**

Tieto testy sú základom pre ostatné typy a podporujú tím v tom, by vyprodukoval kvalitný kód. Zaisťujú internú kvalitu aplikácie a umožňujú rýchle overenie, či v kóde pri programovaní novej funkcie nie je chyba. Zároveň sa tým predchádza vzniku chýb v aplikácii. Zodpovednosť za tieto testy nesú vývojári. V prípade, že v tíme chýbajú znalosti s tvorbou tohto typu testov je vhodné túto znalosť doplniť či už podporou manažementu so zabezpečným školením alebo využitím schopností iného člena, ktorí tieto znalosti má, alebo prijatím nového špecialistu, ktorý touto znalosťou disponuje. Tento kvadrant najviac maximalizuje návratnosť investícií do testovania. Návratnosť investície totiž predstavuje prospech získaný z testovacej aktivity vážený oproti nákladom na túto aktivitu a na najnižšej úrovni budú vynaložené úsilie a náklady na testy najnižšie.

### **Testy čeliace biznisu, ktoré riadia vývoj**

Na rozdiel od tradičných dokumentov s požiadavkami, ktoré sa používajú v tradične riadených projektoch do tohoto kvadrantu patria príklady a testy napísané jazykom biznisu, ktoré vznikli v spolupráci s ním a odrzkadľujú jeho potreby. Zodpovednosť za tieto testy majú väčšinou tester, ktorí pátrajú a pomáhajú biznisu vyjadriť spokojnosť s produktom a tiež vytvárať príklady chceného a nechceného chovania vyvíjanej aplikácie. Vytvárajú sa tu nástroje, ktoré zistia spokojnosť biznisu, ako napríklad kontrolné zoznamy a z týchto zoznamov sa vytvoria testy čeliace biznisu. Vieme nimi predchádzať chybám v podobe vzniku neporozumení a domnienok ešte pred vytvorením kódu. Je vhodné ich zautomatizovať aby podporili rýchlu spätnú väzbu a tým tak mohol dodať hodnotu v čo najkratšom čase.

### **Testy čeliace biznisu, ktoré kritizujú produkt**

Tieto testy vytvárajú pridanú hodnotu výsledného produktu a poskytujú uistenie, či tím programuje funkcionality, ktorá uspokojí zákazníka. Patria sem akceptačné testy, ktoré demonštrujú biznisu dodávaný produkt. Veľkým prínosom tohto kvadrantu je exploratívne testovanie, ktoré dopĺňa automatizáciu a využíva ľudský intelekt a vnímanie na otestovanie systému z rôznych vopred nenaplánovaných uhlov pohľadu. Ďalšími testami sú testy dokumentácie a reportov a regresné testy, ktoré verifikujú, že chovanie testovaného systému sa nezmenilo.

### **Testy čeliace technológiám, ktoré kritizujú produkt**

Tento kvadrant obsahujú testy, ktoré sa ťažko plánujú. Vývojársky tím by mal určiť, či tieto testy sú schopné dodať, alebo je potrebné prizvať externého odborníka. Patria sem „ility“, takto označované podľa koreňa anglických výrazov pre testy bezpečnosti, udržateľnosti, interoperability, kompatibility, spoľahlivosti a inštalovateľnosti. Testy výkonu, škálovateľnosti, záťaže by sa mali uskutočňovať od začiatku projektu. Netreba zabudnúť na plánovanie testov pamäť, aby nedochádzalo k neskorším problémom v tejto oblasti.



## 2.1.4 Kľúčové faktory úspešnosti

Zo skúseností s agilným vývojom a testovaním pozbieraných od autoriek kolekcie *The Agile Testing Collection* (Crispin a Gregory 2008; Gregory a Crispin 2014) vieme určiť faktory, ktoré pomôžu agilným testerom uspieť v agilnom tíme a agilným tímom zase pomôžu s doručením vysoko kvalitného produktu. Tieto faktory sú kritické pre úspech nielen agilných ale ostatných projektov. Ich aplikovanie je vhodné i pre veľké podnikové prostredia s globálne distribuovanými tímami. Globálne distribuované tímy tvorí skupina ľudí, ktorí spolupracujú naprieč časom, priestorom a cez hranice ako jednotlivých štátov, tak i jednotlivých organizácií prostredníctvom počítačových sietí s využitím informačných a komunikačných technológií. (Bedrnová et al. 2012) Patrí sem:

### Celotímový prístup

V agilnom tíme je za testovanie a zvyšovanie kvality zodpovedný celý tím. Táto spoločná zodpovednosť zvyšuje širšie znalosti členov tímu a tak vie tím reagovať rýchlejšie na nečakané situácie. Testeri disponujú hlbšími znalosťami o testovacích technikách a mali by ich zdieľať s ostatnými členmi tímu. Tester by sa mal účastniť už plánovacích stretnutí alebo diskusie o dizajne nových funkcií. V prípade, že sa týchto stretnutí neúčastní je vhodné zaviesť plánovacie stretnutie, ktoré zahŕňa testera, programátora a biznis experta kde sa upresnia požiadavky na produkt.

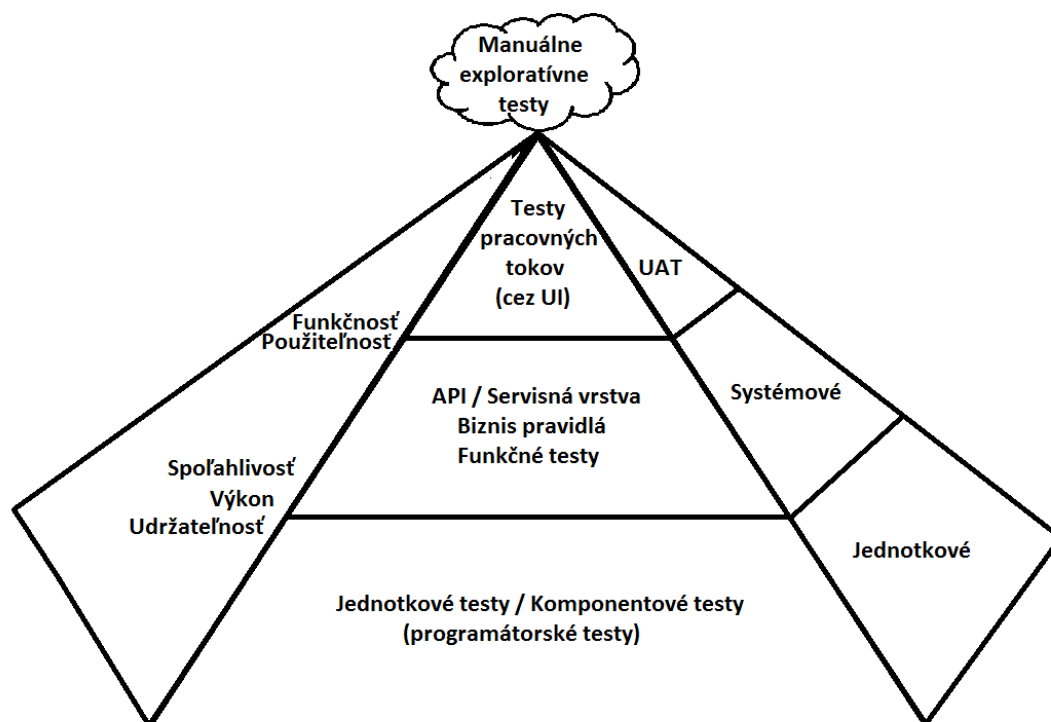
### Myslenie agilného testera

V mysli agilného testera by nemalo byť nič, čo zabraňuje spolupráci s ostatnými členmi tímu podieľajúcimi sa na testovacom procese. Nemal by byť „strážcom kvality“ ako je tomu u tradičných projektov. Sústredenie členov tímu by malo byť na ciele, ktoré si stanovili a mali by robiť všetko čo je v ich silách aby ich naplnili. S úspešným dosiahnutím cieľov majú pomôcť agilné princípy, ktoré sa majú pripomínať a následne vypracovať akčný plán zlepšenia.

### Automatizácia regresných testov

V agilnom testovaní sa snažíme zaistiť čo najrýchlejšiu spätnú väzbu testov, aby sme rýchlo vedeli reagovať na nájdené chyby a mali viac času na ich opravu. Automatizácia poskytuje záchrannú sieť na odchytenie chýb pred produkciou, pomáha riadiť vývoj a udržuje technický dlh na minime. V prípade, že tím produkuje softvér bez použitia najlepších praktík môže sa objaviť technický dlh. Jedná sa o funkcionality, ktoré tím nebol schopný dodať. V prípade, že automatizácia sa v tíme neuskutočňuje a komplexita aplikácie rastie, môže nastať situácia, že množstvo regresných testov narastie do takej výšky, že sa nestihnú otestovať všetky prípady a neodhalia sa tak vzniknuté regresné chyby. Týmto spôsobom sa na testovacie aktivity vynakladá množstvo času a nedá sa sústrediť na iné typy testovania, čím testovanie zaostáva za programovaním.

S rozhodnutím, čo automatizovať pomôže tímu pyramída automatizácie testov zobrazená na obrázku 4. Je použiteľná pre väčšinu tímov a projektov. Jednotlivé vrstvy pyramídy indikujú rýchlosť spätnej väzby poskytnutej testami.



Obrázok 4 - Pyramída automatizácie testov (Gregory a Crispin 2014)

Na najnižšej vrstve sa nachádzajú jednotkové testy. Tieto testy poskytujú najrýchlejšiu spätnú väzbu už na lokálnej stanici vývojára pri písaní kódu. Servisná vrstva alebo vrstva aplikačného rozhrania testuje biznis pravidlá bez použitia užívateľského rozhrania, čím poskytuje rýchlejšiu spätnú väzbu ako testy na vyššej úrovni pyramídy, ktoré bežia cez užívateľské rozhranie. Doplnkom automatizácie sú manuálne exploratívne, ktoré odhalia chyby s ktorými automatické testy nepočítajú. Je však potrebné nájsť cestu ako poskytnúť rýchlu spätnú väzbu z tohoto typu testov.

Nástroje, ktoré môžeme použiť na uskutočnenie jednotlivých úrovní obsahuje pravá strana pyramídy. Neexistuje teda len jeden nástroj, ktorý otestuje všetko a pre každý nástroj je definovaná iná zodpovednosť. Ľavá strana obsahuje atribúty kvality, aby tím nezabudol na ich otestovanie. V prípade, že atribút je zaradený do úrovne, tím začne premýšľať nad tým, aké testy má uskutočniť pre ich zaistenie.

Neexistuje jedno správne riešenie automatizácie testov. Riešenie je nutné nájsť celotímovým experimentovaním. Dospieť k spoločne prospešnému záveru ako pokračovať môže vyžadovať dávku odvahy. Tímy, ktoré sú v agilnom vývoji nové by mali mať skúseného praktika, ktorý vie pomôcť s výberom riešení, stanovením metrík (ukazovateľov stavu systému), nastavením štandardov v tíme a budovaním komunity na zdieľanie skúseností.

### Poskytnutie a prijatie spätnej väzby

Agilný vývoj je efektívnejší oproti tradičnému hlavne vďaka rýchlej spätnej väzbe, ktorú vo vývoji aplikuje. Čím rýchlejšia spätná väzba je, tým rýchlejšie vieme, či produkt je v súlade s očakávaniami zákazníkov. Automatizované testy v tejto oblasti veľmi pomáhajú. Ďalej sa na jej zrýchlení podieľajú i výsledky exploratívneho testovania a reporty reálnych užívateľov. Taktiež musí byť v záujme biznisu, poskytovať rýchlu spätnú väzbu, pretože bez ich spolupráce tím nevie, či je na „správnej ceste“. Dôležité je i vypýtať si spätnú väzbu na svoju prácu, programátori by sa nemali báť vypýtať si viac informácií ak nerozumejú požiadavkám. Patrí sem i diskusia so zákazníkom o splnení kritérií, ktoré požadoval.

### Vybudovanie základu kľúčových praktík

Sem radíme praktiky, ktoré vznikli dávno pred agilnými praktikami a bez nich nevieme dodať vysoko kvalitný produkt.

1. **Kontinuálna integrácia** – overenie integrácie automatickým buildom
2. **Testovacie prostredia** – vytvorenie testovacieho prostredia
3. **Riadenie technického dlhu** – predchádzanie automatizáciou a refaktoringom funkcií
4. **Vytváranie inkrementov** – rozdelenie práce na menšie kúsky kódu
5. **Programovanie a testovanie sú súčasťou jedného procesu** – testy sa píše pred kódom a vývojári sa ich snažia pokryť
6. **Synergia kľúčových praktík** - kombináciou praktík sa zvyšuje kvalita využitím synergického efektu

### Spolupráca so zákazníkmi

Tento faktor zahŕňa príklady zostavené zo spolupráce medzi testerom a biznis expertom, ktoré definujú a prioritizujú užívateľské požiadavky a sú následne spracované na testy riadiace vývoj. Testerí pomáhajú s interpretáciou a pochopením požiadaviek medzi zákazníckym a vývojovým tímom, pretože zákazník väčšinou nevie čo od systému očakáva a má iné predstavy o finálnom produkte.

### Nadhľad

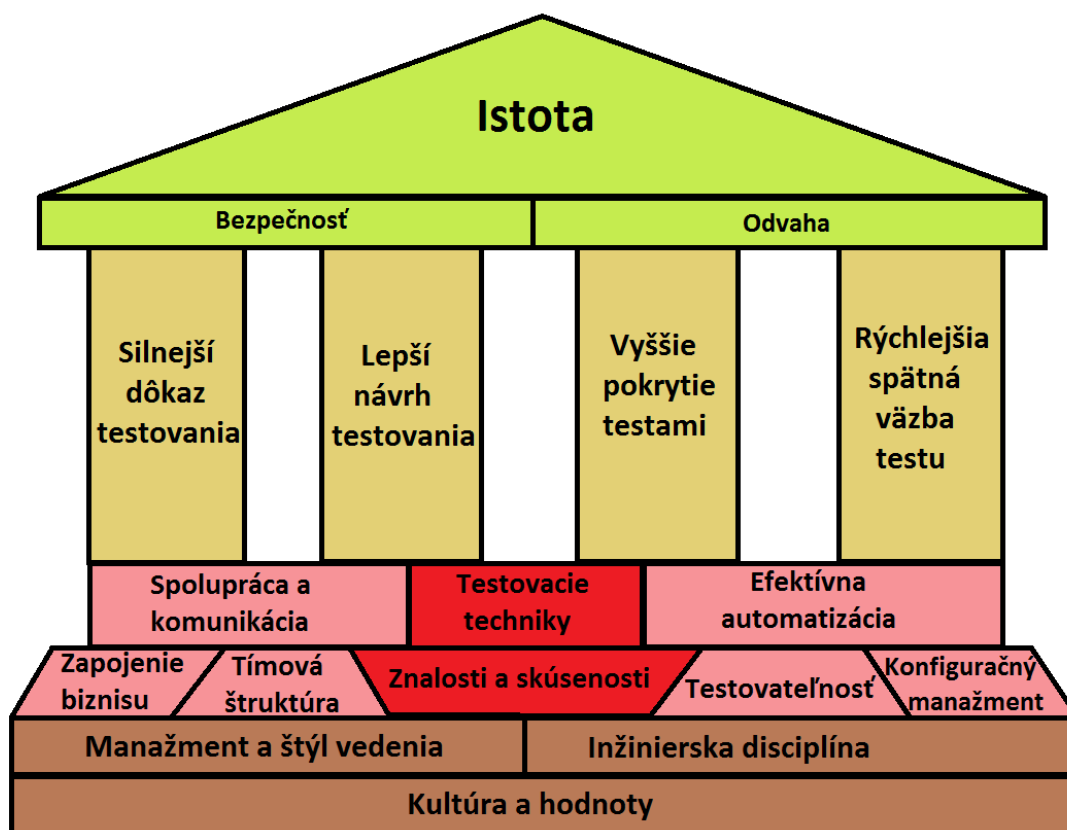
Testerí majú užívateľský pohľad na produkt, čím získavajú nadhľad oproti vývojárom, ktorí sú sústredení na technickú implementáciu. S nadhľadom si tím uvedomuje, na čo všetko sa pri implementácii zabudlo. Testerí majú pomôcť tímu získať prehľad v tom, ako súčasné príbehy zapadajú do celkovej hodnoty, ktoré biznisu prinášajú.

## 2.1.5 Dôvod procesu testovania

Autorom konceptu pilierov je britský coach agilnej kvality David Evans, ktorý v nich prehľadne sumarizuje dôvod procesu testovania. Podľa Evansa testovanie samo o sebe neprináša kvalitu, ale informáciu o kvalite. Cieľom je dosiahnuť istotu v kvalitný produkt, ktorý dodávame zákazníkovi. Zlepšenie procesu testovania túto istotu podľa neho zvyšuje. Hodnotným produktom efektívneho testovania je preto istota. „*Istota spočíva na odvahe a bezpečnosti. Odvaha bez bezpečnosti vedie k nerozvážnosti a bezpečnosť bez odvahy k stagnácii a neaktivite.*“ (Gregory a Crispin 2014, str. 403)

Piliere, ktoré podporujú istotu v testovaní sú:

1. **Silnejší dôkaz testovania** – Ako dobre zainteresované strany rozumejú tomu čo bolo testované a kedy?
2. **Lepší návrh testovania** – Ako primerane, účelné a efektívne sú používané testovacie scenáre?
3. **Vyššie pokrytie testami** – Ako veľmi je systém pokrytý testami v porovnaní s modelmi rizík?
4. **Rýchlejšia spätná väzba testu** - Ako rýchlo identifikujeme problémy a ich nápravné akcie?



Obrázok 5 - Piliere testovania (Evans v Gregory a Crispin 2014)

Piliere predstavujú kvality, ktoré vieme vylepšovať. Viac vylepšení znamená vyššiu istotu v to, že tím dodáva kvalitný produkt. Dosiahnuť však rovnováhu jednotlivých pilierov nie je jednoduché, pretože investícia do jedného, môže znamenať újmu druhého. Podporou pilierov sú *základy*, ktoré obsahujú tri oblasti obsahujúce niekoľko podoblastí:

### 1. Tímové základy

- a. **Spolupráca a komunikácia** – zahŕňa „mäkké znalosti“ člena používané v tíme k druhým členom i mimo tímu voči zákazníkom
- b. **Zapojenie biznisu** – na fungovanie spolupráce je nutné zapojiť zástupcov biznisu do práce s členmi produktového tímu
- c. **Tímová štruktúra** – spoluprácu tiež podporuje výber štruktúr, ktoré podporujú tímovú spoluprácu podporuje ich v sebaorganizácii

### 2. Základy schopností

- a. **Testovacie techniky** – zahŕňajú techniky ako exploratívne testovanie a techniky pre lepší návrh testovania
- b. **Znalosti a skúsenosti** – predstavuje znalosti a skúsenosti tímu s testovacími technikami, čo zaručuje ich efektívnosť

### 3. Technické základy

- a. **Efektívna automatizácia** – schopnosť automatizovať rýchlo a efektívne väčšinu testov
- b. **Konfiguračný manažment** – efektívna automatizácia závisí na konfiguračnom manažmente, ktorý zahŕňa správu verzií, konfigurácie testovacích prostredí a dát, identifikácia dôležitých artefaktov vo vývojovom a testovacom procese
- c. **Testovateľnosť** – zahŕňa systémovú architektúru, ktorá podporuje automatizované testy

*Základy* stoja na *organizačnom podloží* do ktorého patria tri oblasti:

1. **Manažment a štýl vedenia** – patria sem schopnosti manažérov, ktorí by mali motivovať a koučovať svojich zamestnancov
2. **Inžinierska disciplína** – táto oblasť zahŕňa podporu tímov technickými nástrojmi a prostriedkami, ktoré pri svojej práci členovia používajú a tiež zabezpečenie školení a umožniť osobný rast na zaistenie vhodných a aktuálnych znalostí
3. **Kultúra a hodnoty** – patrí sem organizačná kultúra a hodnoty, ktoré spoločnosť uznáva zamestnanci sú s nimi stotožnení

## 2.1.6 Praktiky budovania istoty

Autorky kolekcie *The Agile Testing Collection* Crispin a Gregory (2008) (2014) tvrdia, že neexistujú žiadne zaručené najlepšie praktiky, no veria v kľúčové praktiky, ktoré obohatia najviac agilné tímy. Tieto praktiky majú zvýšiť tímový pocit istoty s dodaním kvalitného produktu, ktorý uspokojí zákazníkov. So získaním istoty potom príde odvaha na experimenty a kontinuálne zlepšovanie. Medzi tieto praktiky radíme:

### Používanie príkladov

Radíme sem vývoj riadený akceptačnými testami, špecifikácie na základe príkladov, vývoj riadený požadovaným chovaním, vývoj riadený testami a kontinuálnu integráciu. Jedná sa o praktiky kedy za spolupráce biznis experta a testera sa vytvoria príklad požadovaných a nepožadovaných chovaní produktu, ktoré pomôžu predísť odmietnutiu zo strany zákazníka ešte pred implementáciou. Ak komplexita produktu nedovoľuje uviesť príklady požadovaného chovania, je vhodné vyhradiť čas na skúmanie chovania, ktoré se potom za pomoci biznisu odsúhlasí a pretaví do testov riadiacich vývoj.

### Exploratívne testovanie

Tento typ testovania slúži na rýchle odhalenie chýb. Neslúži na prevenciu ako iné typy testovania. Je doplnkom automatizácia a využíva percepciu testera na odhalenie nových nápadov ako funkcie otestovať a odhaliť tak rizikové miesta, ktoré automatické testy neodhalili. Mal by poskytovať rýchlu spätnú väzbu a preto je potrebné nájsť spôsob akým ho čo najrýchlejšie uskutočniť a výsledky interpretovať.

### Testovanie funkcií

Je nutné myslieť na oblasti, ktoré nie sú vo funkcionalite zahrnuté, no z hľadiska užívateľov sú veľmi dôležité. Jedná sa o výkonnosť, bezpečnosť, spoľahlivosť a iné atribúty kvality. Tím má identifikovať, ktoré atribúty sú pre zákazníka dôležité a na ne sústrediť svoju pozornosť už pri plánovaní funkcií. Je možné, že tím pri plánovaní stratí nadhľad a pozabudne na tieto dôležité atribúty.

### Kontinuálne učenie

Ak chce tím dodať zákazníkovi produkt s ktorým budú spokojní mal by si vyhradiť čas na učenie. Pri sústredení sa na implementáciu novej funkcie tím často nemá čas na učenie, pretože je to zdĺhavý proces, ktorý sa nedá zrýchliť. Prostredie, ktoré podporuje učenie je charakterizované dostatočným časom na experimenty, rýchly cyklus experimentu s jeho hodnotením spolu s rýchlou reakciou na toto hodnotenie a pocit osobného bezpečia v tíme.

### **Citlivosť na kontext**

Projektový kontext a kontext prostredia ovplyvňujú aplikáciu agilných princípov do práce tímu. Nemôžeme ignorovať systém v ktorom sa tím nachádza a preto by mal tento kontext spoznať a nájsť cestu ako sa mu prispôbiť. Typickými „brzdami“ aplikácie agilných princípov sú zafixované procesy, ktoré nemôžu byť napríklad kvôli zákonu alebo bezpečnosti zmenené. Jedná sa napríklad o zákonné regulátorne požiadavky na vyvíjaný softvér vyplývajúce z požiadaviek vďaka ktorým je kladený väčší dôraz na dokumentáciu a definované procesy.

### **Udržanie sa v realite**

Tím by mal byť schopný reálne stanoviť svoje limity a reálne očakávania na základe sebapoznania. Každý člen je schopný vykonať v určitom časovom úseku iný rozsah práce. Ak tester zistí, že je nad jeho sily zvládnuť napísať všetky testy pred vývojom, mal by mu vývojár pomôcť s testovaním namiesto vývoja nových funkcií, ktoré testom nie sú pokryté. Vhodným spôsobom je zviditeľnenie limitu rozpracovanej práce aby tím mohol prispôbiť svoje kapacity aktuálnej potrebe. Zákazníkom tak tím vie poskytnúť reálny pohľad na stav aplikácie a nie ich zneistiť nereálnymi predstavami.

### **Vytvorenie zdieľanej vízie**

Úspech nezaručuje žiadna z agilných testovacích praktík a je len na tíme samotnom, čo všetko si z týchto praktík vyberie a aplikuje. Tím by mali viesť agilné hodnoty a princípy. Z daného vyplýva, že neexistuje jediné obecné riešenie aplikovateľné pre každý tím ale tímy by mali experimentovať a zabezpečiť rýchle spätno-väzobné cykly, ktoré pomôžu v uistení správneho smeru na dodanie kvalitného produktu.

## 2.2 Vyspelosť agilného vývoja

Pojmom vyspelosť agilného procesu sa zaoberali rôzne štúdie v súvislosti s modelmi agilnej vyspelosti, ktorých bola zostavená celá rada. Komparačné štúdie však poukazujú na to, že aktuálne dostupné modely nie sú použiteľné.

V komparačnej štúdií modelov agilnej vyspelosti Adalı a kolektív (2016) prišiel k záveru, že väčšina modelov a nástrojov na meranie agilnej vyspelosti využíva agilné praktiky na zobrazenie stupňa agility, ktorá predstavuje schopnosť vytvárať a reagovať na zmenu za účelom zisku v turbulentnom biznis prostredí. Nevieme z nich však určiť ako úspešné tieto metódy budú.

Deekonda a Sirigudi (2016) zistili, že dostupné modely agilnej vyspelosti nie sú efektívne v adresovaní potrieb agilných praktikov a to z dôvodu, že ich presne nedodržiavajú. Taktiež tvrdí, že implementácia agilných praktík je špecifická pre organizáciu a typ projektu.

Podľa Schweigerta a kolektívu (2013c) v súčasnosti nevieme nájsť spoločne akceptovaný model agilnej vyspelosti, ale pochybuje o tom, či by novo vzniknutá „metodika“ neohrozila princípy agilného vývoja a nepriblížila sa tak viac k tradičnému modelu vývoja softvéru.

Fontana a kolektív (2014) tvrdí, že praktici nevidia vyspelosť agilného vývoja softvéru ako definíciu procesu či kvantitatívne merateľných schopností. Vyspelosť agilného vývoja znamená skôr podporu subjektívnych schopností, ako je napríklad spolupráca, komunikácia, oddanosť, starostlivosť, zdieľanie a samoorganizácia.

Elssamadisy (2007) je kritický ohľadom agilnej vyspelosti, pretože nepovažuje agilitu za cieľ vývoja softvéru. Podľa Rothmana (2011) sa musí agilná vyspelosť vysporiadať i s kultúrnymi problémami akými sú mindset, vzdialenosť moci a vyhýbanie sa neurčitosti.

*„Ak sa pozrieme na agilnú vyspelosť, je zrejmé, že zlepšenie agilného vývoja a manažmentu nie je len problémom formálnej spôsobilosti alebo stupňov vyspelosti, ale tiež problémom hodnôt, emócií a kultúry.“* (Schweigert et al. 2013c)



## 2.3 Vypelost' agilného testovania

Na definíciu agilnej vypelosti testovania som použil dva zdroje, ktoré sa ňou zaoberali. Prvým zdrojom je verejne dostupný článok pojednávajúci o oblastiach modelu vypelosti agilného testovania (Giudice 2013). Ďalším zdrojom je verejne prístupná webová prezentácia (Shirly 2011).

Podľa Giudiceho (2013) vieme agilnú vypelost' testovania skúmať v piatich oblastiach:

1. Oblasť **správanie testovacieho tímu** je zameraná na zistenie úrovne tímovej spolupráce. Taktiež mapuje zodpovednosti za jednotlivé úrovne testov.
2. Oblasť **organizácia agilného testovania** sa sústreďí na skúmanie prístupu manažéra testovania k agilným praktikám z pohľadu štýlu vedenia, ktorý využíva a ako prispieva k ich podpore.
3. Oblasť **praktiky agilného testovania** skúma ako tester v agilných tímoch koordinujú testovacie aktivity a aké praktiky agilného testovania pri svojej práci tím používa.
4. Oblasť **automatizácia** zisťuje pokrytie automatizovateľných funkcionalít funkčnými a nefunkčnými regresnými testami. Taktiež sa zameriava na zistenie úrovne praktík používaných pri návrhu testov a kódu. Skúma i zodpovednosť za automatizáciu testov.
5. Oblasť **agilná vypelost'** overuje či tester majú agilný mindset (myslenie). Zisťuje ako sú oboznámený s agilnými princípmi a tiež skúma agilnú vypelost' celého tímu.

Shirly (2011) tvrdí, že vypelost' agilného testovania odpovedá aktuálnej vypelosti agilného tímu. Vo svojom modeli vypelosti agilného testovania preto kombinuje vývojový model tímu inšpirovaný od Tuckmana a Jensa (1977) s dvadsiatimi testovacími kategóriami v ktorých vie tím testovacie procedúry rozvíjať. Niektoré procedúry tím nevie dosiahnuť pokým nedosiahne vyššiu tímovú vypelost'. Procedúry teda vieme zlepšovať vždy na konkrétnej úrovni tímovej vypelosti. Kategórie však predstavujú vývoj naprieč jednotlivými úrovňami tímovej vypelosti.

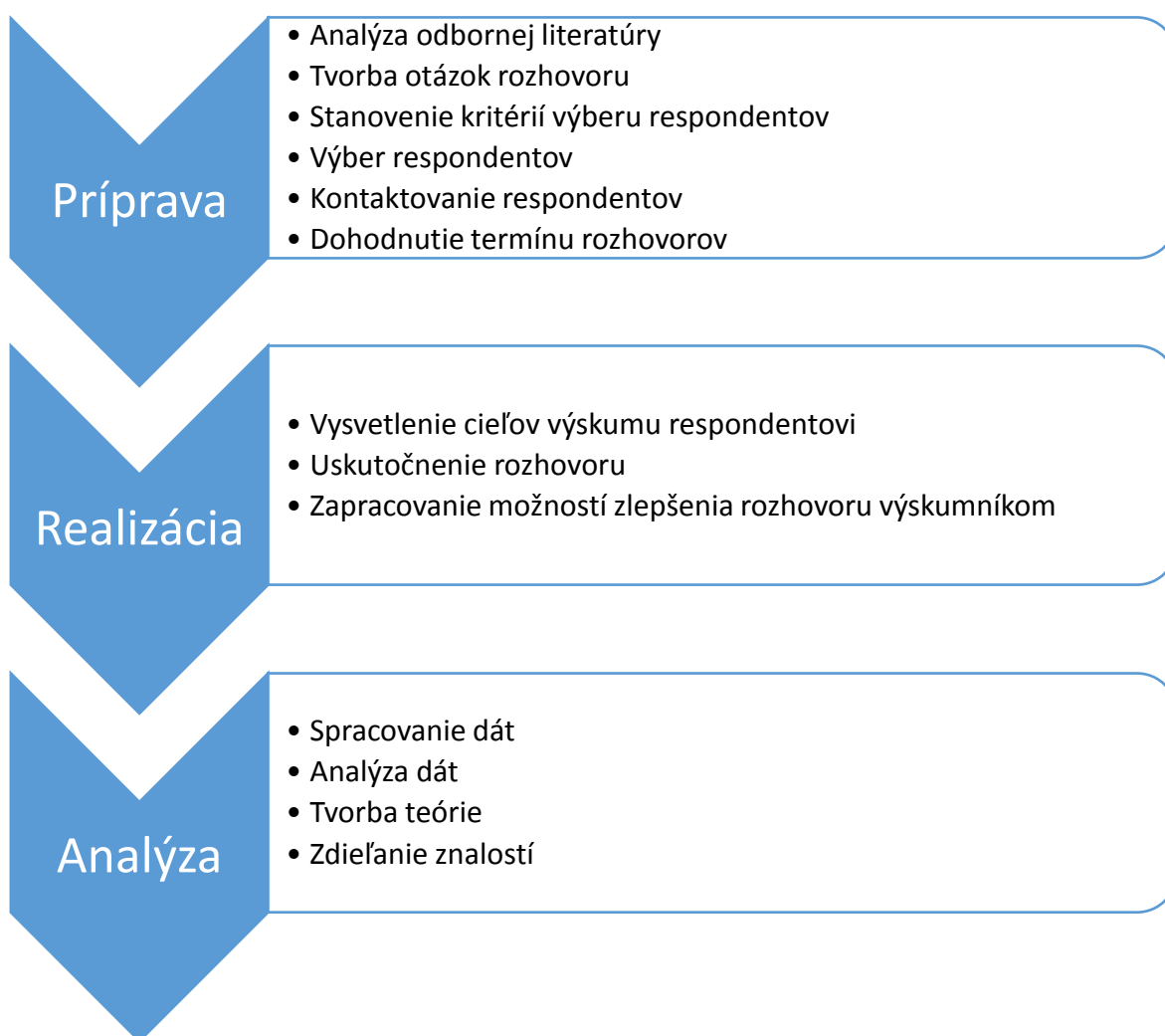
Postup pri zistení agilnej vypelosti testovania je taký, že tím ohodnotí všetky kategórie podľa priradených definícií v odpovedajúcich úrovniach tímovej vypelosti. Zistí tým aktuálny stav a rozhodne sa, čo s ním urobí. Poskytuje i návod pre manažerov testovania aký štýl vedenia majú používať v jednotlivých úrovniach vypelosti tímu. Model je verejne dostupný avšak málo prispôsobiteľný a nevychádza z aktuálnych potrieb členov tímu, čím nezaručuje úspešnosť zlepšenia všetkých uvedených testovacích kategórií.

## 3 METÓDY

Kapitola popisuje použité výskumné metódy a postup výskumníka pri príprave, uskutočnení a analýze rozhovorov. Prvá podkapitola približuje koncepty kvalitatívneho výskumu a zakotveneí teórie, ktoré boli pri výskume použité spolu so stanovenými výskumnými otázkami. Ďalej je popísaný proces tvorby otázok rozhovoru s odkazom na ich znenie.

Nasleduje popis prípravy rozhovorov v ktorej je uvedený spôsob výberu respondentov na základe kritérií, spôsob ich kontaktovania a dohodnutia termínov rozhovorov. Sú popísané základné údaje o realizácii rozhovorov akými sú počet respondentov, časové obdobie výskumu, dĺžka rozhovorov a ďalšie.

Po získaní kvalitatívnych dát je zobrazený postup ich spracovania s uvedenou časovou náročnosťou a následne je približený spôsob analýzy dát a tvorby teórie. Na záver je tiež spomenuté zaistenie validity výskumu, spoľahlivosti otázok a spôsobu zamedzenia predporozumeniu výskumníka. Obrázok 6 obsahuje dielčie kroky výskumníkovho postupu.



Obrázok 6 – Postup výskumníka (autor)

### 3.1 Výskumné metódy

Za hlavnú metódu skúmania bol vybraný kvalitatívny výskum. Podľa Creswella (1998, str. 12) je kvalitatívny výskum definovaný ako process hľadania porozumenia založený na rôznych metodologických tradíciách skúmania daného sociálneho alebo ľudského problému. Výskumník vytvára komplexný, holistický obraz, analyzuje rôzne typy textov, informuje o názoroch účastníkov výskumu a uskutočňuje skúmanie v prirodzených podmienkach.

Ďalšie zdroje podľa Hendla (2016) uvádzajú, že v kvalitatívnom výskume dáta analyzujeme a interpretujeme induktívnym spôsobom. Výskumník na začiatku nepozná konečný tvar skladačky, ktorú zo získaných informácií skladá. Až postupne analýzou dát a nevynechaním ničoho, čo by mohlo s vysvetlením situácie pomôcť získava táto skladačka kontúry.

Na začiatku si výskumník zvolí výskumné otázky, ktoré sa snaží výskumom zodpovedať. Ja som si stanovil zodpovedanie nasledujúcich výskumných otázok vychádzajúcich z cieľov práce.

Hlavné výskumné otázky:

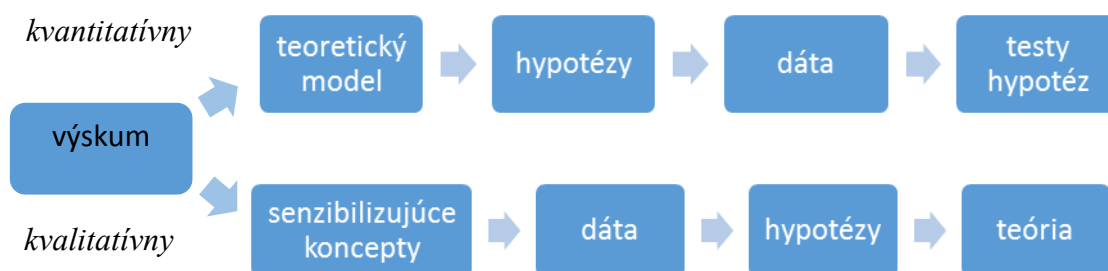
- Čo je vyspelosť v agilnom testovaní?
- Ako vieme zlepšiť vyspelosť agilného testovania?
- Aký stupeň vyspelosti dosahujú skúmané tímy?

Doplnkové výskumné otázky:

- Sú testerí potrební v agilných tímoch?

Podľa Hendla (2016) môže výskumník výskumné otázky modifikovať alebo dopĺňať v priebehu výskumu, behom zberu a analýzy dát. Kvalitatívny výskum sa preto považuje za pružný typ výskumu kedy v jeho priebehu nevznikajú len výskumné otázky, ale tiež hypotézy i nové rozhodnutia, ako modifikovať zvolený výskumný plán a pokračovať pri zbere dát a ich analýze. Vyhľadávajú a analyzujú sa všetky informácie, ktoré prispievajú k osvetleniu výskumných otázok. Výskumník následne uskutočňuje deduktívne a induktívne závery z ktorých môže navrhnúť teóriu o pozorovanom fenoméne.

Obrázok 7 porovnáva priebeh kvantitatívneho a kvalitatívneho výskumu.



**Obrázok 7 – Porovnanie priebehu kvantitatívneho a kvalitatívneho výskumu Hendla (2016) preložené autorom**

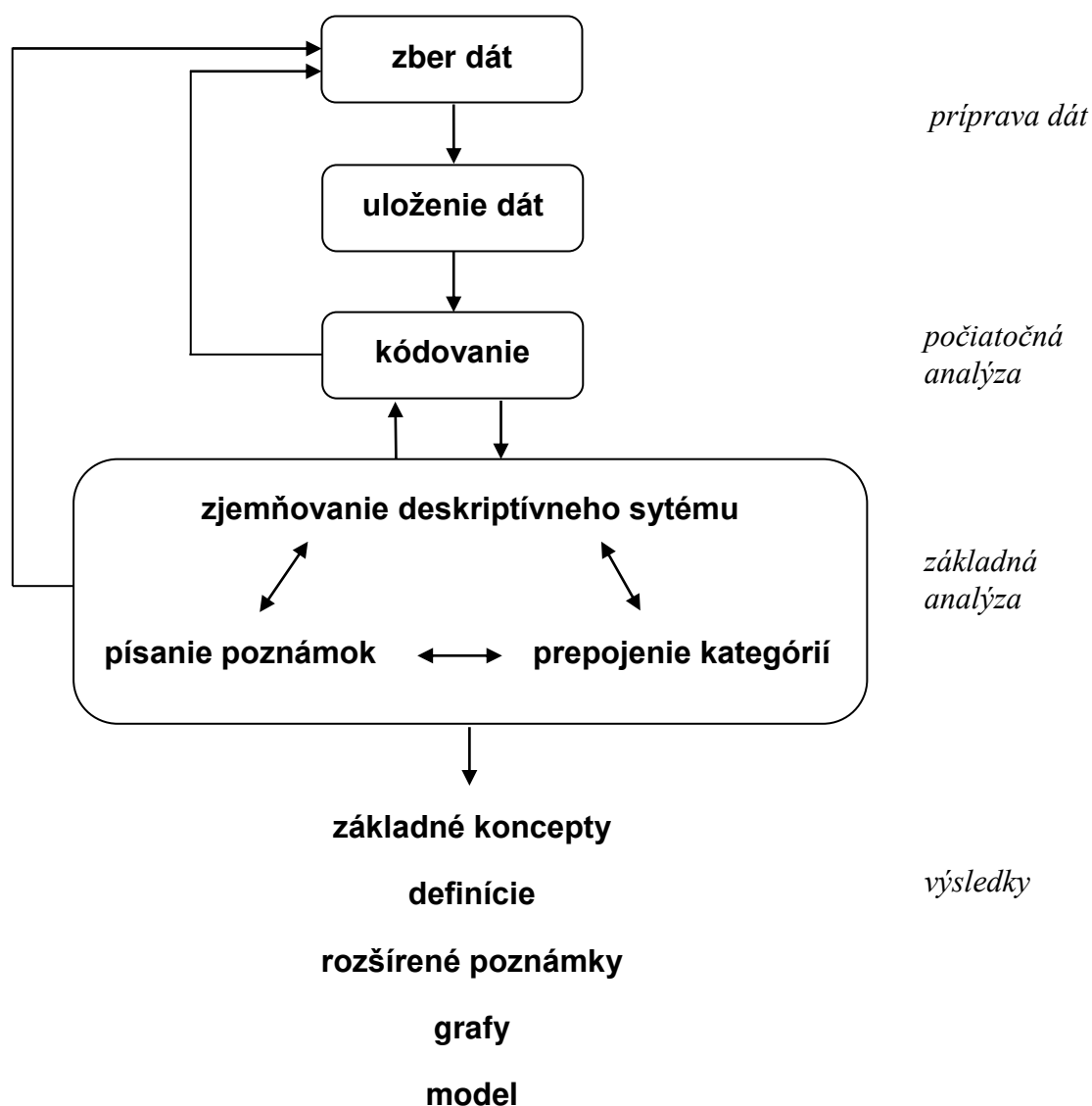
Cieľom kvalitatívneho výskumu za použitia konceptov zakotvanej teórie je podľa Stola (2016) generovanie teórie namiesto testovania a overovania existujúcej teórie, čo vyplýva z obrázku 7. Senzibilizujúce koncepty (teoretické pojmy), sa podľa Hendla (2016) navrhujú na základe konceptualizácie dát a to priradením označení udalostiam a javom.

Za stratégiu kvalitatívneho výskumu a spôsob analýzy získaných dát som zvolil zakotvenú teóriu (grounded theory) hlavne preto, že poskytuje procedúry pre vytváranie teórie v oblastiach, ktoré sú málo teoreticky spracované. Tiež vieme nájsť dostatok príkladov jej použitia. Cieľom výskumu, ktorý vychádza zo stratégie zakotvanej teórie, je návrh teórie pre fenomény v určitej situácii, na ktoré je zameraná pozornosť výskumníka. Vznikajúca teória je zakotvená v dátach získaných behom štúdie. V konkrétnom výskume bádatelia modifikujú alebo vypúšťajú niektoré kroky celého postupu.

Zakotvená teória sa používa pri objavovaní a rozvíjaní nových teórií. Keďže nesie v sebe prvky neistoty a intuície nevieme jej proces algoritmizovať a postup procesu tento aspekt tiež zohľadňuje. Proces zakotvanej teórie podľa zobrazuje obrázok 8.

Základnými prvkami zakotvanej teórie podľa Hendla (2016) sú koncepty (teoretické pojmy). Ďalej sú to kategórie, ktoré sú abstraktnejšie ako koncepty a ukazujú podobnosti alebo rozdiely. Jedná sa v podstate o zoskupenie konceptov do kategórií. Posledným prvkom sú prepozície zobrazujúce zobecnené vzťahy na úrovniach kategória-koncept a kategória-kategória.

Na odkrytie významu dát sa v zakotvanej teórie používa kódovanie v troch fázach. Prvou fázou je **otvorené kódovanie**, kedy výskumník prechádza dáta a priradzuje im označenia na nízkej úrovni abstrakcie. Tento zoznam sa postupne prepracúva v ďalšej analýze. Druhou fázou je **axiálne kódovanie** v ktorom výskumník hľadá súvislosti medzi jednotlivými kategóriami a upravuje alebo pridáva ďalšie koncepty alebo kategórie. V poslednej fázi ktorú predstavuje **selektívne kódovanie** výskumník integruje výsledky výskumu a hľadá hlavné témy, ktoré budú základom novej teórie.



Obrázok 8 – Postup zakotvenej teórie Hendla (2016) preložené autorom

## 3.2 Tvorba otázok rozhovoru

Keďže v hlavných výskumných otázkach je skúmaná vyspelosť procesu testovania, ktorá je definovaná úrovňami vyspelosti na ktoré spoločnosť aspiruje, otázky rozhovoru sa zameriavali na zistenie vývoja procesu testovania s pohľadom na minulosť, súčasnosť a aspirácie tímov do budúcnosti. Otázky rozhovoru boli tiež inšpirované piliermi testovania od Davida Evansa uvedené v podkapitole 2.1.5. Piliere obsahujú oblasti zamerané na zvýšenie kvality testovacieho procesu, ktorým vieme zvýšiť vyspelosť testovania. Keďže produktom testovania je istota v kvalitný produkt, otázky boli zamerané na zistenie oblastí, ktoré vplyvajú na zvýšenie alebo zníženie istoty členov tímu v kvalitný produkt. Posledná sekcia otázok sa zameriavala na zisťovanie prínosu testerov a tým aj testovacieho procesu v tíme, čím sa potvrdzuje význam výskumu a možnosť jeho použitia v praxi.

Je dôležité zmieniť, že prvá verzia dotazníku mala omnoho viac sekcií, no pre časové obmedzenia výskumu a s prihliadnutím na sústredenosť respondentov bolo množstvo otázok z prvej verzie zúžené a vo verzii dva rozdelené na sekciu základných a doplnkových otázok. Základné otázky boli kladené každému respondentovi a doplnkové otázky mali usmerniť respondenta pri odbehnutí od témy.

K zmene znenia otázok rozhovoru a pridávaniu ďalších doplnkových otázok dochádzalo v priebehu rozhovorov alebo po uskutočnení rozhovoru, a to zo zistení, ktoré rozhovor priniesol, čo je podľa Hendla (2016) v súlade s konceptom zakotvenej teórie, kedy sa predpokladá postupné prispôsobenie výskumného plánu a tým i otázok rozhovoru novým zisteniam výskumníka, ktoré v procese výskumu nastali. Pridané doplnkové otázky boli kladené v prípade, že respondent danú oblasť nespomenul, ale pre výskumné potreby bol pohľad na tieto oblasti potrebný z dôvodu vylúčenia alebo potvrdenia ich významnosti. Druhá verzia otázok rozhovoru je uvedená v Prílohe A.

### 3.3 Príprava a priebeh rozhovorov

Výber bol založený na následne uvedených štyroch kritériách. Prvým kritériom bolo to, že skúmané tímy sa zaoberajú agilným vývojom softvéru. Toto kritérium vychádza zo zamerania a cieľov práce na skúmanie agilných procesov a ich odlišnosti od procesov tradičných vodopádových.

Druhým kritériom bolo použitie spoločnej metodiky vývoja softvéru. Všetky skúmané tímy používajú Scrum ako metodiku na riadenie vývoja softvéru, preto som sa pri výbere počtu respondentov riadil rolami, ktoré táto metodika definuje. Za výskumnú vzorku som teda považoval scrum mastera, product ownera a scrum tím vo vytvorenom zložení z dvoch vývojárov, dvoch testerov, interakčného dizajnéra. Väčšie zastúpenie rovnakých rolí v scrum tíme bolo vytvorené z dôvodu získania rôznych pohľadov na skúmanú problematiku. V prípade, že všetky role neboli v súčasnosti k dispozícii, vybral som člena tímu, ktorý v súčasnosti zastáva inú pozíciu, no v minulosti sa venoval tej, ktorá mi do vzorku chýbala. Ak však rola v tíme chýbala a nebol tam ani nikto kto ju v minulosti zastával, nie je vo tímovom vzorku zahrnutá.

Tretím bola dlhoročná skúsenosť respondenta v oblasti agilného vývoja pričom pri nízkom počte respondentov v tíme boli vybraní i respondenti s menšími skúsenosťami v agilnom vývoji. Dĺžka skúseností sa pohybovala od 0,5 roka až po 12 rokov.

Štvrtým kritériom bol súhlas s nahrávaním rozhovoru. Toto kritérium splnili všetci respondenti, ktorí boli vopred informovaný o priebehu rozhovorov (viď Príloha C). Podrobné údaje o respondentoch sú uvedené v kapitolách 4.1, 4.2, 4.3, 4.4 a 4.5.

Tímy boli autorom práce kontaktované prostredníctvom e-mailovej komunikácie v ktorej bol pripojený i Formulár informovaného súhlasu s uskutočnením rozhovorov. Termíny uskutočnenia rozhovorov boli dohodnuté podľa pracovnej disponibility respondentov.

Rozhovory prebiehali od 6.2.2017 do 2.3.2017 za účasti primárneho výskumníka (autora) a vždy jedného respondenta. Miestom rozhovorov bola spoločnosť v ktorej tím pôsobí. Opýtaných bolo 23 respondentov a dĺžka rozhovorov sa pohybovala medzi 20 – 50 min. Cílená dĺžka rozhovorov bola 45 minút, no niektorí respondenti neboli výreční, alebo mali rýchlejšiu reč ako ostatní.

Otázky rozhovoru boli pôvodne definované v slovenskom jazyku kvôli zníženiu neporozumenia, ktorý cudzí jazyk prináša, pričom pre jedného respondenta boli upravené do anglického jazyka. Otázky boli otvorené a poskytovali respondentom slobodu v prezentácii svojich skúseností a názorov na skúmanú tému. Na konci rozhovoru bola položená doplnková otázka, skúmajúca, či respondent má ešte na mysli niečo, s čím by sa chcel podeliť.

### 3.4 Spracovanie dát

Z nahrávok bola následne autorom vytvorená transkripcia jednotlivých respondentov zoskupených podľa spoločností v ktorej pôsobia. Na transkripciu bol použitý *Express Scribe Transcription Software* (<http://www.nch.com.au/scribe/>). Pri prepise bola použitá čiastočná transkripcia na nezrozumiteľné časti nahrávok, nerelevantné informácie alebo slovné vaty. Pri prezentácii spracovaných dát v práci som použil mužský rod z dôvodu zvýšenia anonymizácie respondentov. Nasledujúca tabuľka 3 zobrazuje časovú náročnosť uskutočnenej transkripcie.

Tabuľka 3 – Časová náročnosť transkripcie (autor)

Identifikácia tímu	Počet respondentov	Priemerná časová náročnosť transkripcie v hodinách	Súhrnná časová náročnosť transkripcie v hodinách
Tím 1	7	4	28
Tím 2	6	4	24
Tím 3	4	4	16
Tím 4	4	4	16
Tím 5	2	4	8
<b>SPOLU</b>	<b>23</b>	<b>4</b>	<b>92</b>

### 3.5 Analýza dát

Spracované dáta boli analyzované za použitia softvéru pre analýzu kvantitatívnych a kvalitatívnych dát *MAXQDA Analytics Pro* (<http://www.maxqda.com/products/maxqda-analytics-pro>).

Pre analýzu dát boli použité koncepty kódovania Straussovej Grounded Theory, ktorá tri fáze kódovania uvedené v podkapitole 3.1.

Prvá fáza kódovania, definovaná ako otvorené kódovanie, bola uskutočnená pri prvom čítaní textov jednotlivých rozhovorov, pričom boli vytvárané kódy na základe toho o čom respondent hovorí a ako sa cíti. Tieto kódy boli následne za použitia *MAXQDA Analytics Pro* zoskupené podľa vývoja procesu testovania na tri hlavné kategórie minulosť, súčasnosť a budúcnosť a ostatné kategórie. Kategórie súčasnosť a budúcnosť obsahovala dáta vývojového procesu až po súčasnosť. Kategória budúcnosť obsahovala problémy a návrhy na zlepšenie identifikované respondentami. Ostatné kategórie obsahovali odpovede na otázky prínosu testerov, vnímanie kvality a iné.

V druhej fáze, ktorou je axiálne kódovanie boli v okódovaných segmentoch hľadané logické súvislosti a boli vytvorené obecnější kódy. Tieto kódy obsahovali kódy z otvoreného kódovania so spoločnými znakmi.

Treťou fázou bolo selektívne kódovanie. Táto fáza obsahovala definovanie centrálnych kategórií z ktorých boli následne odvodené spoločné znaky testovacieho vývoja v podkapitole 4.6.6 a spoločné problémy tímov v podkapitole 4.8. Výstupy z analýzy v *MAXQDA Analytics Pro* sú k dispozícii v Prílohe D.

Validita bola ohrozená rizikom neporozumenia medzi výskumníkom a respondentom. Na minimalizáciu tohto rizika bol cieľ výskumu vysvetlený jednotlivým účastníkom. Pre zaistenie spoľahlivosti otázok rozhovoru boli uskutočnené 3 pilotné rozhovory v tíme 1 a otázky boli prispôbené problémom, ktoré z nich vznikli. Na minimalizáciu výskumníkových predporozumení boli rozhovory nahrávané a transkribované. Predporozumením sa rozumie ovplyvnenie úsudku vlastnými skúsenosťami výskumníka z praxe, napríklad jeho pôsobením v agilnom tíme. Bez nahrávania a následnej transkripcie by tak výskumník mohol urobiť chybu pri dokumentácii výpovedí prostredníctvom kľúčových slov počas rozhovoru.

Posledným dôležitým bodom bolo zdieľanie výsledkov výskumu so zúčastnenými spoločnosťami a prípadné poskytnutie konzultácií s objasnením navrhnutých zlepšení alebo iných nejasností.



## 4 POZNATKY VÝSKUMU

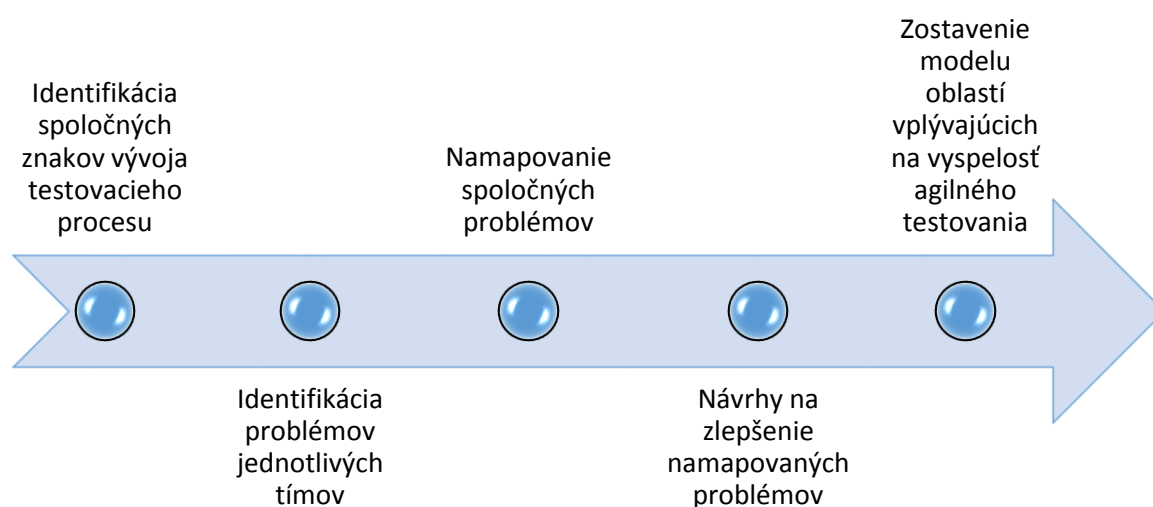
V tejto kapitole sa venujem priblíženiu podnikového prostredia v ktorom skúmané tímy pôsobia a tiež predstaveniu jednotlivých opýtaných členov tímu. Názvy spoločností a mená členov tímov sú anonymizované z dôvodu ochrany firemných a osobných údajov.

Prvým krokom je popis vývoja testovacieho procesu na základe kvalitatívnych dát a za využitia konceptov Zakotvenej teórie sú definované spoločné znaky vývojového procesu testovania, ktoré majú priaznivý vplyv na testovací proces a kvalitu produktu.

V každom tíme sú následne identifikované problémy, ktoré vyplynuli z rozhovorov. Následne sú identifikované spoločné problémy tímov v mapovacej tabuľke problémov. Ďalším krokom sú návrhy na zlepšenie jednotlivých problémov z pohľadu autora, literatúry a návrhov členov skúmaných tímov.

Z týchto spoločných problémov a spoločných znakov vývoja testovacieho procesu je zostavený model a oblasti, ktoré majú vplyv na vyspelosť agilného testovania.

Vizuálny indikátor postupu výskumnej kapitoly:



Obrázok 9 – Indikátor postupu výskumu (autor)

## 4.1 Predstavenie tímu 1

### 4.1.1 Organizačné prostredie

**Spoločnosť 1** je svetovou zdravotníckou organizáciou s dlhoročnými skúsenosťami vo výrobe inovatívnych produktov a poskytovaní služieb, ktoré zlepšujú životy po celom svete. Sústreďí sa na zlepšovanie svojich produktov prostredníctvom špecializovaných odborníkov, analýzy zákazníckych potrieb a podpory diverzity v spoločnosti. Keďže má vysokú zodpovednosť za bezpečnosť ľudských životov a vysoký etický záväzok, musí jej činnosť podliehať reguláciám. Podporuje tímovú spoluprácu a agilný vývoj softvéru. Jej podniková informatika poskytuje podporu prostredníctvom vývoja a použitia interných aplikácií, ktoré uľahčujú mnohé procesy spoločnosti. Informatika je koncentrovaná v niekoľkých centrách v rôznych kútoch sveta a zamestnáva interný personál, no využíva i externých zamestnancov.

### 4.1.2 Identifikácia tímu

**Tím 1**, ktorý pracuje v tejto spoločnosti sa zameriava na vývoj mobilnej aplikácie ktorej užívateľom sú interní zamestnanci. Zároveň sú k tejto aplikácii tímom vytvorené i webové aplikácie, bez ktorých by informatika nepodporila biznis procesy. Metodikou riadenia vývoja softvéru je Scrum. Dodržiavajú sa všetky Scrum ceremónie s tým, že tím si metodiku prispôbil podľa svojej potreby. Pri vývoji produktu tím využíva svoje skúsenosti a najlepšie praktiky pričom neustále hľadá zlepšenia. Zo skúmaných tímov je tento tím najväčší a tvorí ho až 15 členov pričom je rozdelený na 2 subtímy, ktoré majú medzi sebou závislosti.

Nasledujúca tabuľka 4 ukazuje základné údaje o opýtaných členoch tímu 1.

Tabuľka 4 – Špecifikácia opýtaných členov tímu 1 (autor)

Respondent	Skúsenosti s vývojom softvéru (roky)	Zastávané role	Pôsobenie v spoločnosti (roky)	Pôsobenie na projekte (roky)	Skúsenosti s agilným prostredím (roky)
Vývojár 1	4	front-end vývojár back-end vývojár	1	1	1
Vývojár 2	17	vývojár	3	1	4
Testlead	13	tester, testlead, vývojár webového rozhrania, test analytik, test automation engineer, scrum master	1	1	5
Tester	12	vývojár, integračný inžinier, support, automation engineer, test lead, test manager	1	1	12
Product owner	10	IT projektový manažér, konzultant, scrum master, product owner	2	1	2
Scrum master	8	projektový manažér, agilný kouč, scrum master	2	1	8
Interakčný dizajnér	4	front-end vývojár, UX dizajnér	2	1	1

## 4.2 Predstavenie tímu 2

### 4.2.1 Organizačné prostredie

**Spoločnosť 2** sa špecializuje na marketingové analýzy a on-line monitoring sociálnych sietí. Pomáha pracovníkom v oblasti on-line marketingu vytvárať lepšie marketingové kampane pochopením fungovania trhu na sociálnych médiách. Spoločnosť neustále monitoruje, analyzuje a porovnáva milióny profilov na najväčších sociálnych sieťach a pomáha identifikovať najzaujímavejší obsah pre zákazníkov. Má pobočky po celom svete v ktorých pôsobia zamestnanci rôznych národností. Zaoberá sa agilným vývojom softvéru za účelom predaja na lokálnom trhu i v zahraničí, poskytuje poradenstvo v oblasti informačných technológií a ďalšie služby. Podporuje tímovú spoluprácu, inovácie, otvorenosť. Zdieľa svoje poznatky s informatickou verejnosťou a nebojí sa hovoriť o svojich procesoch navonok.

## 4.2.2 Identifikácia tímu

**Tím 2**, ktorý pracuje v tejto spoločnosti vyvíja webovú aplikáciu, ktorá poskytuje zákazníkom náhľad na zlepšenie stratégie sociálnych médií. Aplikácia je prevažne frontendová a za backend sa považujú výpočty prebiehajúce nad databázami. Metodikou riadenia vývoja softvéru je Scrum. Tím si túto metodiku prispôbil podľa svojej potreby. Pri vývoji produktu tím využíva svoje skúsenosti a najlepšie praktiky pričom neustále hľadá zlepšenia. Nasledujúca tabuľka 5 ukazuje základné údaje o opýtaných členoch tímu 2.

Tabuľka 5 – Špecifikácia opýtaných členov tímu 2 (autor)

Respondent	Skúsenosti s vývojom softvéru (roky)	Zastávané role	Pôsobenie v spoločnosti (roky)	Pôsobenie na projekte (roky)	Skúsenosti s agilným prostredím (roky)
<b>Product owner</b>	7,5	vývojár, UX dizajnér, product owner	3,5	2	7,5
<b>Tester</b>	3	tester	2	2	3
<b>Testlead</b>	15	QA inžinier, QA manažér, SW development manažér	2	2	8
<b>Vývojár 1</b>	1	vývojár	1	1	1
<b>Vývojár 2</b>	10	tester, vývojár	10	1	10
<b>Vedúci softvérového inžinierstva / Scrum master</b>	9	vývojár, scrum master, vedúci softvérového inžinierstva	9	N*	5

\* hodnota N znamená, že respondentova rola nie je členom projektového tímu, pretože v súčasnosti zastáva manažérsku rolu nad skúmaným tímom

## 4.3 Predstavenie tímu 3

### 4.3.1 Organizačné prostredie

**Spoločnosť 3** prevádzkovateľ internetového portálu so širokým portfóliom služieb. Okrem nej na trhu pôsobia i ďalší prevádzkovatelia internetových portálov. Spoločnosť podporuje inovácie a nové nápady. Svoje riešenia považuje za kvalitné a jednoduché určené pre každodenné použitie. Je zameraná na rýchlu a efektívnu komunikáciu so svojimi užívateľmi. Podporuje nápady navrhované užívateľmi. Poskytuje otvorené a neformálne prostredie s tým, že stanovené pravidlá pracovníci rešpektujú. Vyvíja softvér pre lokálny trh bezplatne i za účelom predaja. V budúcnosti nepredpokladá presun na zahraničné trhy, čo vyplýva z jej misie kde vidí svoje postavenie ako čisto česká spoločnosť.

### 4.3.2 Identifikácia tímu

**Tím 3**, ktorý pracuje v tejto spoločnosti vyvíja webovú aplikáciu, ktorá je dostupná zákazníkom zdarma. Na mobilnej verzii tejto aplikácie pracuje iný tím. V súčasnosti je tento tím bez testerov. Členovia tímu chcú túto skutočnosť zmeniť, pretože sú nespokojní. Aktívne hľadajú túto rolu do svojich radov. Metodikou riadenia vývoja softvéru je Scrum. Tím si túto metodiku prispôbil podľa svojej potreby. Pri vývoji produktu tím využíva svoje skúsenosti a najlepšie praktiky pričom neustále hľadá zlepšenia.

Nasledujúca tabuľka 6 ukazuje základné údaje o opýtaných členoch tímu 3.

Tabuľka 6 – Špecifikácia opýtaných členov tímu 3 (autor)

Respondent	Skúsenosti s vývojom softvéru (roky)	Zastávané role	Pôsobenie v spoločnosti (roky)	Pôsobenie na projekte (roky)	Skúsenosti s agilným prostredím (roky)
Scrum master	4	tester, analytik, scrum master	1	1	4
Vedúci softvérového inžinierstva	25	vývojár, architekt, vedúci architektúry, vedúci vývoja, vedúci vývojového oddelenia	2	N*	2
Vývojár	10	vývojár	10	4	6
Product owner	14	produktový dizajnér, product owner, grafik, dizajnér,	0,5	0,5	8

\* hodnota N znamená, že respondentova rola nie je členom projektového tímu, pretože v súčasnosti zastáva manažérsku rolu nad skúmaným tímom

## 4.4 Predstavenie tímu 4

### 4.4.1 Organizačné prostredie

**Spoločnosť 4** sa zaoberá vývojom mobilných a webových aplikácií pre spoločnosti všetkých veľkostí a predáva ich väčšinou na americkom trhu. Spoločnosť ponúka široké portfólio aplikácií v kategóriách ako cestovanie, životný štýl, sociálne vyžitie, internetové podnikanie a iné. Sústreďí sa na individuálny talent aplikovaný v tímovej spolupráci. Má zaväzok v dodávaní excelentných riešení, kedy pracovníci hľadajú nové možnosti zlepšenia s nadšením a nikdy nie sú spokojný so súčasným stavom. Tími cieľia na prvotriedny užívateľský zážitok z aplikácie. Podporuje inovácie technológií, ktoré pomáhajú vo vytváraní produktov top kvality a spoluprácu so zákazníkom. V organizačnej štruktúre tejto spoločnosti nenájdeme projektových manažérov čím sú zákazníci v priamom kontakte s tímami. Odstraňuje zdĺhavé procesy, pretože sa sústreďuje na čo najrýchlejšie dokončenie práce.

#### 4.4.2 Identifikácia tímu

**Tím 4**, ktorý pracuje v tejto spoločnosti vyvíja webovú investičnú platformu podporujúcu skupinové financovanie nehnuteľností. Aplikácia je vyvíjaná v tuzemsku a je určená pre americký trh. Metodikou riadenia vývoja softvéru je Scrum. Tím si túto metodiku prispôbil podľa svojej potreby. Pri vývoji produktu tím využíva svoje skúsenosti a najlepšie praktiky pričom neustále hľadá zlepšenia. V súčasnosti na projekte pôsobia dvaja testerí v rámci QA (quality assurance) tímu. QA tímom sa tu rozumie testovací tím a toto označenie je používané i v ďalšom texte. Špecifickým rysom tímu 4 je, že jeden jeho člen zastáva niekoľko rolí naraz.

Nasledujúca tabuľka 7 ukazuje základné údaje o opýtaných členoch tímu 4.

**Tabuľka 7 – Špecifikácia opýtaných členov tímu 4 (autor)**

Respondent	Skúsenosti s vývojom softvéru (roky)	Zastávané role	Pôsobenie v spoločnosti (roky)	Pôsobenie na projekte (roky)	Skúsenosti s agilným prostredím (roky)
<b>Tester 1</b>	1	tester, product manažér	1	1	1
<b>Tester 2</b>	2,5	tester	0,5	0,5	0,5
<b>Vývojár</b>	6	kóder, vývojár,	1	1	2
<b>Product owner / Scrum master / Vývojár / Tester</b>	6,5	teamlead, product manažér, vývojár, tester, scrum master	1,5	1,5	6,5

## 4.5 Predstavenie tímu 5

### 4.5.1 Organizačné prostredie

**Spoločnosť 5** sa sústreďuje na vývoj inovačných technológií spoluprácou špičkových odborníkov z rôznych vedeckých oblastí. Jedná sa o medzinárodnú spoločnosť s pobočkami po celom svete, ktorá podporuje diverzitu v organizácii. Záväzkom organizácie je priniesť výhody technológií ľuďom a organizáciám, ktoré to potrebujú najviac. Technológie vytvára s ohľadom na šetrnosť životného prostredia. Vyvíja softvér pre zahraničný i lokálny trh bezplatne i za účelom predaja. Aplikácie sú zamerané na komunikáciu prostredníctvom internetu.

### 4.5.2 Identifikácia tímu

**Tím 5**, ktorý v tejto spoločnosti pôsobil sa zameriaval na vývoj desktopovej platformy na podporu elektronickej komunikácie. V súčasnosti je projekt na ktorom tím pracoval ukončený. Kvôli zlej dohľadateľnosti pôvodných členov tímu boli do tímového vzorku zapojení len dvaja členovia tvoriaci scrum tím. Metodikou riadenia vývoja softvéru bol Scrum. Tím si túto metodiku prispôbil podľa svojej potreby. Pri vývoji produktu tím využíval svoje skúsenosti a najlepšie praktiky pričom neustále hľadal zlepšenia.

Nasledujúca tabuľka 8 ukazuje základné údaje o opýtaných členoch tímu 5.

**Tabuľka 8 – Špecifikácia opýtaných členov tímu 5 (autor)**

Respondent	Skúsenosti s vývojom softvéru (roky)	Zastávané role	Pôsobenie v spoločnosti (roky)	Pôsobenie na projekte (roky)	Skúsenosti s agilným prostredím (roky)
<b>Vývojár</b>	12	vývojár	3,5	1	3,5
<b>Tester</b>	12	tester, test automation engineer	5	3	7

## 4.6 Vývoj testovacieho procesu v tímoch



Obrázok 10 – Indikátor postupu výskumu krok 1 (autor)

Táto podkapitola približuje vývoj testovacích procesov jednotlivých tímov ako „príbeh“ od vzniku tímu až po súčasnosť. Všetky tímy v súčasnosti existujú, až na tím 5, ktorý sa rozpadol. Nad kvalitatívnymi dátami z rozhovorov bol uskutočnený postup za použitia stratégie zakotvenej teórie a boli vytvorené kategórie spoločné pre vývoj testovacieho procesu vo všetkých tímoch. Tieto spoločné kategórie sú v ďalšom texte nazvané ako spoločné znaky vývojového procesu. Tieto znaky podporujú testovací proces a zvyšujú kvalitu produktu.

### 4.6.1 Vývoj testovacieho procesu tímu 1

Na začiatku sa testovanie uskutočňovalo manuálne bez automatizácie. Respondenti uviedli, že odhady práce boli založené na nereálnych dátach. Veľa technických úloh malo nulové ohodnotenie úloh. Podľa niektorých členov tímu prevládala na začiatku prehnaný optimizmus. Testovací proces sa uskutočňoval tak, že funkcia sa vyvinula a posunula sa z vývoja na test, tester a tiež product owner ich otestoval a prešlo sa na ďalšiu funkciu. O automatizácii sa len hovorilo.

Niektorí členovia tímu testovanie vnímali oddelene od vývoja a nemali záujem o pochopenie dôvodu testovania. Jedným z uvádzaných dôvodov bolo to, že členovia sa až tak dobre nepoznali a tiež spolupráca bola horšia. Začalo sa zdôrazňovať, že zodpovednosť za testovanie je zodpovednosť všetkých.

Testovací proces bol podľa niektorých členov zo začiatku chaotický a bez základných praktík ako je spätná väzba. Členovia sa aktívne snažili zlepšiť tento stav analýzou možností automatizácie či dokonca prelnaním rolí tester a vývojár, kedy tester odporúčal



vývojárom lepšie spôsoby programovania. Podľa niektorých členov sa prelínanie rolí nie úplne dosiahlo. Niektorí členovia považovali za problém zavedenie zmeny do praxe, kedy sa na nej dohodlo, ale nič sa neudialo.

Tím si uvedomil potrebu zmeny na základe vlastných skúseností akými boli nárast komplexity aplikácie a prvý release, kedy ešte veľa chýb nebolo opravených, alebo sa objavovali regresné chyby. Tím si preto uvedomil, že sa musí viac sústrediť na kvalitu a nepodľahnúť tlaku zákazníkov na dodávanie nových funkcií. Taktiež si tím uvedomil, že regulátorne prostredie je obmedzením, ktoré nemôžu brať na ľahkú váhu.

Po release jedna tím zmenil plán a zameril sa na míľniky, ktoré si stanovil sám. Taktiež sa začali vyvíjať stratégie kvality a v nej zahrnutá automatizačná stratégia, práca s požiadavkami a ďalšie. Začali sa robiť akčné plány na oblasti, ktoré členovia identifikovali za nutné vyriešiť hneď či už z pohľadu technického, nastavovania cieľov alebo zdieľania znalostí a koučingu. Členovia spomínajú, že retrospektívu začali využívať na hľadanie zlepšení.

Vývojári, interakční dizajnéri, product owner a scrum master sa snažili aktívne zmeniť myslenie biznisu tak aby sa sústredili na pre nich skutočnú hodnotu. Ďalšie zlepšenie, ktorí niektorí členovia považujú za dôležité je rozdelenie tímu na dva subtímy.

Istota v to, že tím dodáva kvalitný produkt bola v minulosti horšia, ľudia sa báli spustiť aplikáciu aby niečo „nevyplávalo na povrch“, tvrdí Scrum master. Dnes niektorí spájajú túto istotu s automatickými testami, ktoré v minulosti neboli vytvorené.

V súčasnosti sa uskutočňuje exploratívne testovanie, čo považujú členovia tímu za prínos. Taktiež sú automatizované jednotkové testy, integračné testy a niekoľko testovacích sád vrátane end-to-end testov v podobe testov užívateľského rozhrania spúšťané cez kontinuálnu integráciu. Väčšina týchto akceptačných testov je stále manuálna.

Testerí stále vnímajú to, že testovanie je oddelené od vývoja. Ostatní tvrdia, že vývojári a testerí sú prepojení už od začiatku vývoja funkcie. Tím sa začal na veci pozeráť reálnejšie a je prístupnejší k zmenám procesov. Podľa niektorých sa prijaté zmeny ešte stále neprijmú do praxe na 100%. Tiež sa zmenilo myslenie zainteresovaných strán a ich prístup. Všetci členovia sa však zhodujú, že sa testovací proces zlepšil. Sú spokojní so súčasným stavom, ale vidia oblasti pre jeho zlepšenie. Istota v kvalitný produkt sa zvýšila oproti minulosti, no niektorí členovia sú optimistickí a niektorí kvalite produktu neveria. Všetci však vidia potenciál pre zlepšenie.

Spolupráca medzi členmi tímu sa zlepšila, členovia sa spoznali a snažia sa pozeráť na veci i očami iných pracovných rolí v tíme. Niektorí však vnímajú potenciál pre zlepšenie spolupráce a komunikácie v tíme. Metriky, ktoré by tím sledoval, nemal ani nemá stanovené, ale zameriava sa skôr na spätnú väzbu od klientov, spätnú väzbu výsledkov automatizovaných testov či empirické poznanie. Niektorí členovia tímu považujú počet chýb za metriku s dobrou vypovedacou schopnosťou, niektorí s tým nesúhlasia.

## 4.6.2 Vývoj testovacieho procesu tímu 2

V start-upovej fáze vývoja spoločnosti neexistovalo testovanie. Vývojári sa dohodli na tom či aplikácia je schopná ísť do produkčného prostredia. Po príchode testera sa testovanie uskutočňovalo manuálne. Existoval pool testerov, ktorí sa pridelovali do viacerých tímov. Pool testerov budem v ďalšom texte považovať i za testovaciu organizáciu ako tomu bolo v tíme 1. Tieto pojmy vnímam ako synonymá k pojmu Testing Center of Excellence, ktorý je definovaný ako „*formálna, alebo neformálna skupina jedincov vyznávajúcich, šíriacich a ďalej rozvíjajúcich strategicky dôležité testovacie know-how.*“ (Doležel 2017, str. 10)

Produkt začal narastať a prichádzali väčší klienti, tak si tím uvedomil, že je dôležité zamerať sa na udržanie kvality. Preto sa postupne testovací tím rozrástol a začal byť zodpovedný za automatizáciu, pričom bol čas kedy ich písal i vývojár. Niektoré funkcie boli však zautomatizované len na polovicu. Tieto poloautomatizované testy sa neskôr zrušili. Zrušil sa tiež pool testerov a vytvorili sa špecialisti na automatizáciu testovania pôsobiaci vo vývojovom tíme.

Automatizácia sa riešila na úrovni akceptačných testov v nástroji Selenium. Unit a integračné testy neboli potrebné keďže charakter produktu nie je závislý na backende. Začali sa robiť akčné plány na oblasti, ktoré členovia identifikovali za nutné vyriešiť hneď či už z pohľadu technického alebo zdieľania znalostí a koučingu.

Testovanie bolo vnímané zo začiatku skôr ako príťaž, no tento pohľad sa zmenil keď si tím uvedomil hodnotu testera.

Tím sa aktívne snažil zrýchliť a optimalizovať svoj vývojový proces. Postupne zavádzal nové pravidlá, praktiky a neustále sa učil. Členovia zaviedli napríklad pravidlo, že keď sa začne regresné testovanie, tak musí byť ukončený vývoj na produkte. Predtým toto pravidlo nebolo, čo narušovalo regresné testovanie.

Taktiež sa snažili zaviesť vývoj riadený testami, ale neuspeli. Dôvodom je podľa niektorých členov veľká komplexita produktu do ktorej sa snažil tento proces zaviesť.

V rámci optimalizácie sa tím išiel poradiť v inej spoločnosti, ktorá zdieľala s tímom skúsenosti. Po zavedení tejto skúsenosti sa zrýchlil čas nasadzovania novej verzie do produkčného prostredia.

Istota v kvalitný produkt zo začiatku nebola žiadna. Čakalo sa na spätnú väzbu od zákazníka. Dnes sú si kvalitou členovia istejší. Niektorí tak súdia na základe lepšej spätnej väzby od zákazníka.

Dnes sú menej pokryté nové funkcie testami, pretože klienti tlačia na vývoj nových funkcií. Testovanie sa začalo sústreďovať na overovanie funkcií hodnotných pre zákazníka, ale podľa niektorých členov je to oblasť na zlepšenie. Vývojový proces sa podľa niektorých členov v súčasnosti nemení a je nastavený dobre.

Testerí nedisponujú takými technickými znalosťami ako vývojári, ale majú technické znalosti s automatizáciou akceptačných testov. Niektorí členovia tímu automatizáciu nepovažujú za nástroj pre zvýšenie kvality produktu, ale kvalitu môže naopak znížiť. Zrýchli sa doručovanie, ale na jej zvýšenie je dôležitý ľudský kontakt s aplikáciou.

Vývojári sa sústredia na svoju rolu a za testovanie sú zodpovední testerí. Všetci sú so súčasným procesom testovania spokojní ale vidia potenciál pre zlepšenie. Spolupráca a komunikácia medzi vývojármi a testerí je pružná.

Metriky, ktoré by tím sledoval, nemal ani nemá stanovené, ale zameriava sa skôr na spätnú väzbu od klientov, počet regresných chýb či empirické poznanie. Podľa niektorých členov záležia metriky na očakávaniach čo chce tím zlepšiť.

### 4.6.3 Vývoj testovacieho procesu tímu 3

Zo začiatku webovú aplikáciu testovali vývojári a product owner. Testerí v tíme neboli. Neskôr jeden tester testoval manuálne mobilnú aplikáciu. Začala sa testovať i webová časť exploratívne, no z testera sa stal po nejakom čase vývojár.

Vývoj prebiehal tak, že sa vyvinula funkcia a dala sa na test testerovi. Testovanie bolo oddelené od vývoja. Product owner napísal zoznam funkcií, ktoré by sa mali pred každým releasom testovať. Zo začiatku vývojári vnímali testovanie za nie úplne dôležité. Písala sa nová funkcionálna, no nikto nevedel, prečo sa to robí. S automatizáciou začali vývojári, keď si uvedomili potrebu unit testov pre nové funkcie. Niektoré staré funkcie neboli pokryté unit testami a len veľmi málo komponent je pokrytých integranými testami.

S komunikáciou a spoluprácou boli členovia tímu spokojní hlavne vďaka rýchlej spätnej väzbe a vzájomnej blízkosti. Tím si uvedomuje potrebu zmeny procesov na základe spätnej väzby od klientov, alebo zo skúseností z iných spoločností v ktorých členovia tímu v minulosti pracovali.

Istota v kvalitný produkt bola v minulosti nízka hlavne kvôli počtu chýb nájdených na produkcii. V súčasnosti nemajú respondenti istotu v kvalitu produktu ale z vývojári produktu veria na základe toho, že produkt má medzi zákazníkmi dobré meno a chyba sa preto toleruje lepšie.

V súčasnosti je tím bez testera ale aktívne túto rolu do tímu hľadajú. Vývojári boli už za túto rolu radi a chyba im v tíme. Predtým keď s touto rolou nemali skúsenosť, nechýbala im. Testerskú rolu prevzali znovu vývojári a product owner. Súčasný stav členovia vnímajú tak, že si vývojári odvykli od sústredenia sa na kvalitu. Proces testovania sa podľa niektorých respondentov skôr zhoršil ako zlepšil. Z vývojárskeho pohľadu je proces testovania na dobrej ceste z pohľadu jednotkových testov. Čo sa týka ostatných respondentov sú nespokojní so súčasným stavom testovania.

Tím aktívne hľadá zlepšenia procesov testovania, ktoré by im pomohli odstrániť neistotu ako napríklad zavádzanie beta verzií aplikácie, ako riadenej distribúcie za účelom získania spätnej väzby z trhu a tiež zavedenie predprodukčného prostredia, hľadanie skutočných potrieb užívateľov, rozšírenie povedomia o dôležitosti kvality produktu. Tím nemá stanovené žiadne metriky na sledovanie zlepšenia testovacieho procesu. Využíva empirické poznanie.

#### 4.6.4 Vývoj testovacieho procesu tímu 4

Zo začiatku žiadne testerská rola v tíme nebola, vývojári a product owner si testovali aplikáciu sami. Zo zvýšením komplexity produktu prišlo uvedomenie si potreby prácnejšieho testovania. Na začlenenie testerov do tímu musel však testerský tím dokázať klientovi svoju hodnotu, čo sa mu podarilo, a klient bol tak spokojný, že ich do tímu začlenil. Po príchode testerov do tímu sa začalo s exploratívnym testovaním.

Vývojári boli zodpovední za písanie unit a integračných testov, čo im zostalo až dodnes. Bez QA (Quality Assurance) sa klienti viac sťažovali a od jej zavedenia sa počet chýb na produkcii znížil. Vývojári na začiatku nevedeli čo od testingu očakávať, pretože s ním nemali skúsenosti a nedávali tomu veľké šance. Po začlenení QA do tímu sa toto vnímanie zmenilo, pretože sa mohli sústrediť viac na svoju prácu a QA tím im ukázal prípady na ktoré vývojári nemysleli.

Členovia tímu aktívne hľadali zlepšenia v oblastiach komunikácie, vytvoreni nových procesov testovania, testovacej stratégie, dĺžky Sprintu, usmernení klienta. V minulosti ľudia v tíme neboli toľko v kontakte. Dnes spolupráca a komunikácia je rýchla a efektívna i zásluhou vzájomnej blízkosti jednotlivých členov. Dôvera v kvalitu produktu bola v minulosti nízka. Po zavedení QA sa zvýšila. V súčasnosti sú si všetci členovia istí kvalitou produktu, ktorý nasadzujú do produkcie.

V súčasnosti sa používa exploratívne testovanie. QA tím považuje proces testovania za stabilný a nevidí, že by sa v súčasnosti niečo menilo. Všetci členovia tímu sú spokojní so súčasným stavom testovania ale vidia potenciál pre zlepšenie. Tím sa začal sústrediť viac na kvalitu keď si uvedomil, že do aplikácie sa registrujú reálni ľudia, pretože predtým nebola aplikácia online. Tím nemá stanovené žiadne metriky na sledovanie zlepšenia testovacieho procesu. Využíva empirické poznanie.

#### 4.6.5 Vývoj testovacieho procesu tímu 5

Testerov bolo menej a dohánali prácu vývojárov. Uskutočňovalo sa exploratívne testovanie celým tímom alebo časťou tímu. Testovanie teda prebiehalo manuálne. Na regresné testovanie sa využívali tímy v zahraničí. Členovia tímu tvrdia, že procesy boli dobre nastavené hlavne vďaka tomu, že v tíme boli skúsení ľudia už od vzniku tímu, ktorí tieto procesy nastavili na základe svojich skúseností.

Organizácia testerov bola oddelená od organizácie vývojárov, čo opýtaní považujú za problém. Obmedzením testovania bolo množstvo zariadení, ktoré bolo nutné overiť. Chyby s nízkou prioritou sa neuzatvárali. Pozeralo sa len na tie s vysokou prioritou. Posledný rok sa vytvárali tiež integračné testy. Na automatizáciu integračných testov boli alokovaní ľudia, no nepriniesla veľkú hodnotu. Výsledky týchto testov neprinášali veľkú hodnotu a boli vývojárom ľahostajné.

O zlepšení procesu sa dozvedeli na základe sledovania počtu chýb. Členovia tímu stále hľadali zlepšenia aj tam kde sa už nedalo čo zlepšiť. Išlo sa do detailov bez toho, že by sa muselo. Podľa testerov v tíme od začiatku bolo každému jasné, že kvalita je dôležitá. Z vývojárskeho pohľadu ich kvalita veľmi netrápila, ale boli zameraní na svoju prácu.

Tester považuje spoluprácu a komunikáciu za bezproblémovú no vývojár tvrdí, že nebol úplne vždy stav, že by sa tester spolupracoval od začiatku s vývojárom a product ownerom. Vývojár si nebol istý kvalitou produktu, ale tester je optimistický a veril mu.

Potom nastala zmena. Zrušili sa testerské role. Vývojár i tester nevnímajú túto zmenu pozitívne. Podľa výpovede nastal chaos a každý sa spoliehal, že testovanie sa urobí samo. Kvalita produktu išla dolu. Všetci vnímajú, že agilné je to, že sa môže čokoľvek zmeniť.

## 4.6.6 Spoločné znaky testovacieho procesu

### Kontinuálne učenie

Počas vývoja testovacieho procesu som vypozeroval vzory kontinuálneho učenia, ktoré sa v tímoch uskutočňuje v nasledujúcich fázach:

1. Uvedomenie potreby zmeny
2. Aktívne hľadanie zlepšení
3. Uvedenie zmeny do praxe

Tím sa na základe neustáleho hľadania zlepšení v rôznych oblastiach posúva dopredu a prispieva tým k vyspelosti vývojového, testovacieho procesu i celého tímu.

### Tím 1

*Keď sa zamyslím nad bojovaním s tými istými chybami dookola a diskutovaním na retrospektíve a zrazu si uvedomením toho, že robíme to isté stále, tím to začal adresovať.*

*Product owner*

*Chvála Bohu sme si to aspoň v nejaký čas uvedomili a začali sa sústrediť na to ostatné a to bol taký kick [štartovací impulz] myslím si, že do celého tímu, kedy si uvedomili, že tá kvalita je fakt zlá, a začali sa celý tím viac sústrediť na otvorené bugy, na bugy, ktoré sú tam, na uzatváranie bugov a na nerobenie bugov [prevenciu], lebo vedeli, že máme ďalšie tri sprinty a za tie sprinty už to proste nejde ďalej.*

*Scrum master*

*A myslím si, že som videl, že tester odporúčal iné spôsoby [...] spôsoby programovania vývojárom aby zlepšili výkonnosť.*

*Product owner*

*Proces stále vyladujeme ako sme sa bavili, že sa to stále nejak ako na retrospektíve rozoberá a stále sa to snažíme zlepšiť.*

*Vývojár*

*Musíme až narazíme na problém riešiť ako to riešiť lepšie, tak s tým testovaním tam u toho SIT [formálne testovanie uspokojujúce regulatorne požiadavky] sme riešili, že to zaberá strašne veľa času, tak proste ok, poďme to zautomatizovať, budeme si púšťať automatické testy a uvidíme proste či to môžeme použiť na to a dva týždne proste len tak seknúť [skrátit] na dva dni alebo deň alebo čo. [...] Takže takým štýlom sa vyvíjame, ešte tam nie sme, nemôžeme povedať, že dokážeme proste SIT spraviť za jeden deň [...] ale nejak sa to vyvíja postupne.*

*Vývojár*

## **Tím 2**

*O tom je celý agile vlastne. [...] Pre mňa agile nie je o nejakých akoby mílnikoch ale o kontinuálnych drobných zmenách, ktoré ťa vedú k niečomu lepšiemu. Snažíme identifikovať prečo. Čo tu bol za problém. Aha tu bol problém s deployom [nasadením], pretože sme tu niekde v testovacom prostredí vybuildovali [vytvorili] zľú verziu, prečo? Čo s tým urobíme. A zase ako sa zlepšuješ, zlepšuješ, zlepšuješ. [...]*

*Vedúci vývoja*

*Už sme sa tu niekoľkokrát snažili zaviesť [...] TDD [vývoj riadený testami], ale to sme nezvládli, pretože [...] sme sa snažili zaviesť nejaký takýto smer do už existujúceho, veľmi rozvinutého, košatého produktu, tak to bolo viac ako práce než vo výsledku sme tomu chceli venovať.*

*Vývojár*

*Keď som sem prišiel, to čo sa produkovalo, veľakrát v produkcii ani nikde nefungovalo, bolo to naozaj veľmi zlé. Mojou hlavnou úlohou bola naštartovať quality assurance, dať ten tím dohromady*

*Test manager*

*Takže sa stále snažíme optimalizovať a to či sa nám to darí alebo tak práve vidíme v tom, koľko tých bugov a v tej regresii nájde, ako dlho nám to trvá*

*Tester*

**Tím 3**

*Takže zatiaľ to zlepšenie vnímam v tom, že všetci kľúčoví ľudia v tíme si uvedomujú, že to musíme riešiť a že to musíme riešiť hneď a máme vlastne i podporu vedenia v tomto takže som rád [...]*

*Product owner*

*vymyslel som nejaký spôsob ako by sa to testovanie malo zlepšiť. Mám to i dohodnuté vlastne, svoje ciele kam to mám posunúť, zlepšenie kvality obecné [...] zlepšenie celkovej kvality softvéru je jeden z cieľov, ktoré mám nastavené tu na tento rok a intenzívne sa tomu venujem [...]*

*Vedúci vývoja*

*Ale ja som sa snažil keď som prišiel zistiť, prečo tí ľudia nemajú radi našu službu, lebo mal som pocit, že to ani nevieme, a myslím si, že už teraz to vieme lepšie a mojou snahou teraz je klásť im čo najmenej prekážok v naplňovaní ich cieľov na našej službe a pomáhať im prosto nejakým spôsobom používať, možno aj nejakým spôsobom rozšíriť ich obzory v zmysle nejakej inšpirácie [...]*

*Product owner*

*Tak dobre, ja som to pochopil a snažím sa postupne od toho vývoja cez ten náš test postupne urobiť ten automat aby sme ho v ideálnom stave dostiahli tu do toho preprodu, kde to už bude akože v produkcii a si myslím, že by bolo úplne skvelé keby sa nám to podarilo. [...]*

*Vedúci vývoja*

**Tím 4**

*Ja som si uvedomil, že je to dôležité v tej dobe keď akoby som nemal úplne čas strážiť zo všetkých strán všetky tasky na tej funkcionalite, ktorú sme vyrábali.*

*Vývojár*

*Testoval som rôzne mobilné aplikácie a zároveň som vymyslel nejaký proces.*

*Tester*

*No on potom akoby navrhol tie týždenné Sprints a že keď je vlastne [...] nejaký backlog kde sa plánuje všetko čo sa vlastne chce urobiť.*

*Tester*



*Museli sme mu najprv posielat' test stratégiu a vtedy sme si to nejak začali ujasňovať, že akoby ako by to malo fungovať a následne sme si sadli akože s celým vývojovým tímom a povedali sme si, že ok tak spravíme jeden Sprint odvtedy dovtedy a QA by malo začať testovať vtedy a vtedy. [...] Keď sme začali tie pravidelné meetingy v ten piatok. [...]*

*Tester*

## **Tím 5**

*My sme potom už riešili hodne detailné veci, pretože nám to bežalo v podstate, ale furt sme si mysleli, že to nie je možné, že by nebol nejaký problém, tak sme sa do toho šúrali a tak. Stále sme nejaké drobnosti riešili [...] ale to nebolo ani nutné riešiť.*

*Tester*

## 4.7 Identifikácia problémov jednotlivých tímov



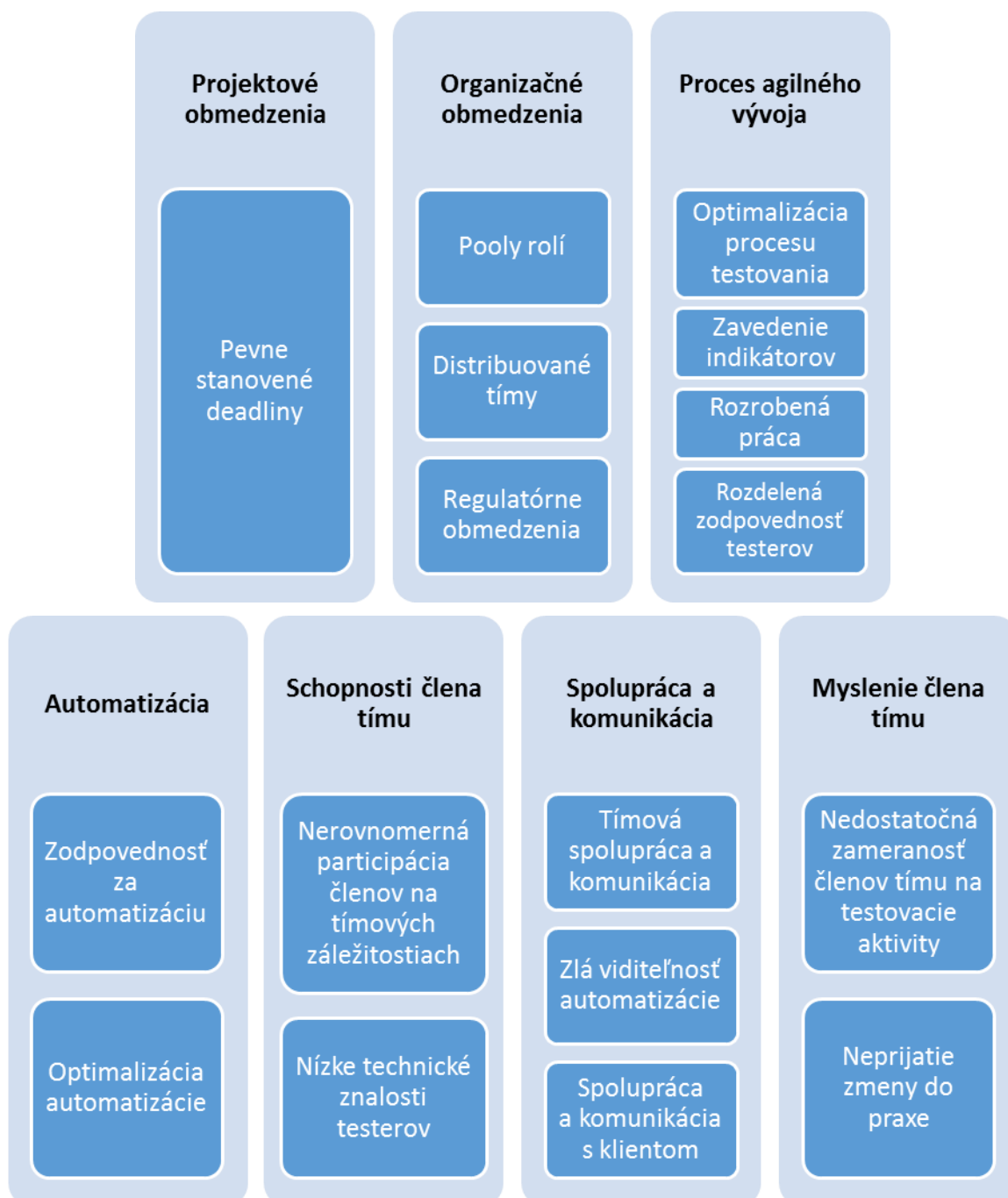
**Obrázok 11 – Indikátor postupu výskumu krok 2 (autor)**

Táto podkapitola popisuje problémy extrahované s rozhovorov s jednotlivými členmi skúmaných na základe konceptov zakotvenej teórie, kedy problémy nižšej úrovne boli generalizované a priradené do spoločných kategórií. V ďalšom texte sú tieto generalizované kategórie pomenované ako problémové oblasti a problémy nižšej úrovne ako problémové podoblasti.

Tieto oblasti môžu mať pozitívny alebo negatívny vplyv na testovací proces i kvalitu produktu preto je nutné nájsť rovnováhu v hľadaní zlepšení pre konkrétny tím. Čo funguje pre jeden tím, nemusí byť prospešné pre iný. Preto odporúčam, aby sa nad každou oblasťou tím zamyslel, ohodnotil jej opodstatnenosť a spoločne našiel optimálny plán na zlepšenie, ktorý mu bude vyhovovať.

### 4.7.1 Identifikované problémy tímu 1

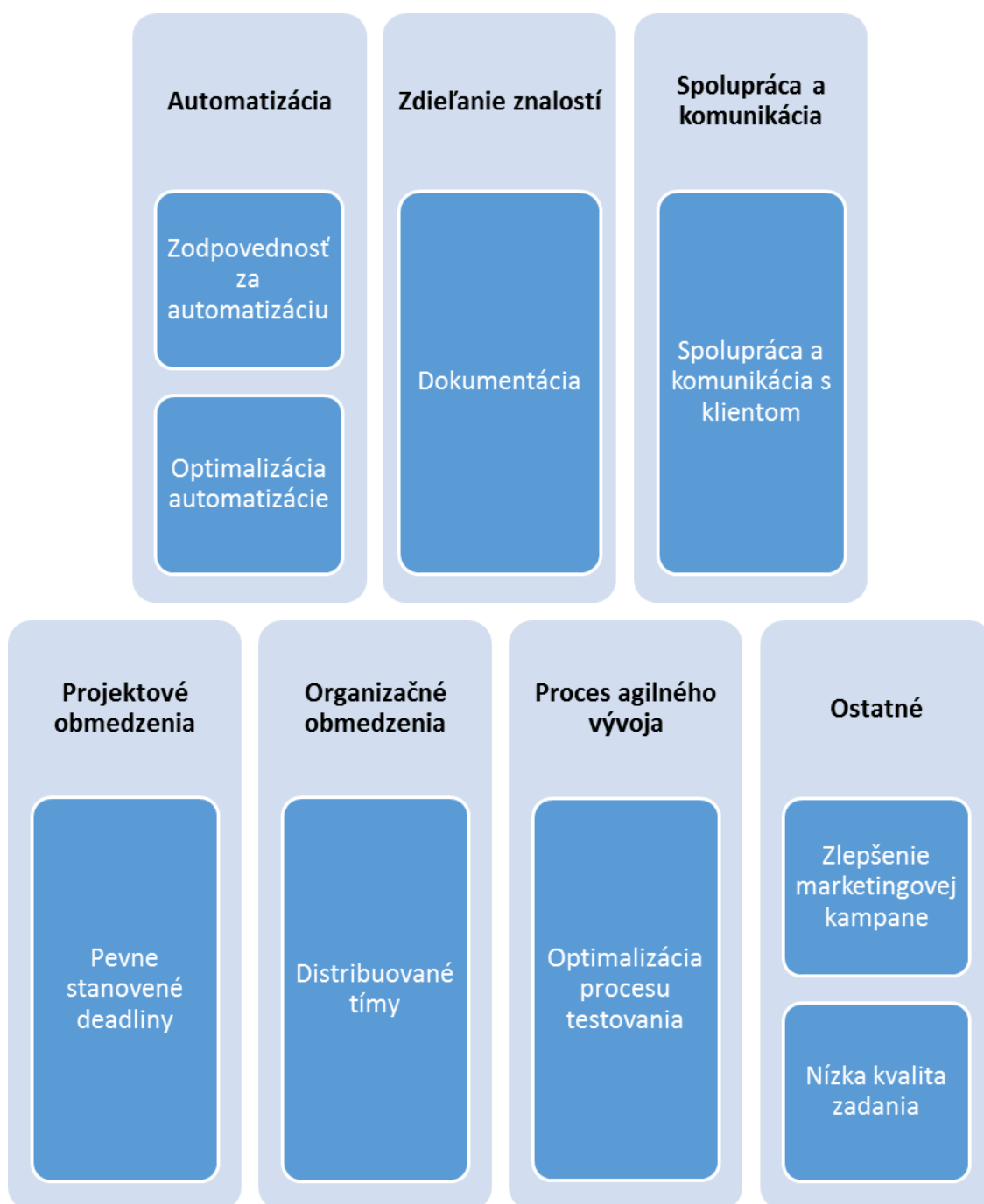
Nasledujúci diagram zobrazuje problémy extrahované z rozhovorov s jednotlivými členmi tímu 1. Členovia ich identifikovali sami ako pre nich dôležité oblasti nad ktorými by sa mal tím zamyslieť a rozhodnúť sa čo s nimi urobí. Podrobný popis problémových podoblastí obsahuje príloha C.



Obrázok 12 – Problémové oblasti tímu 1 (autor)

### 4.7.2 Identifikované problémy tímu 2

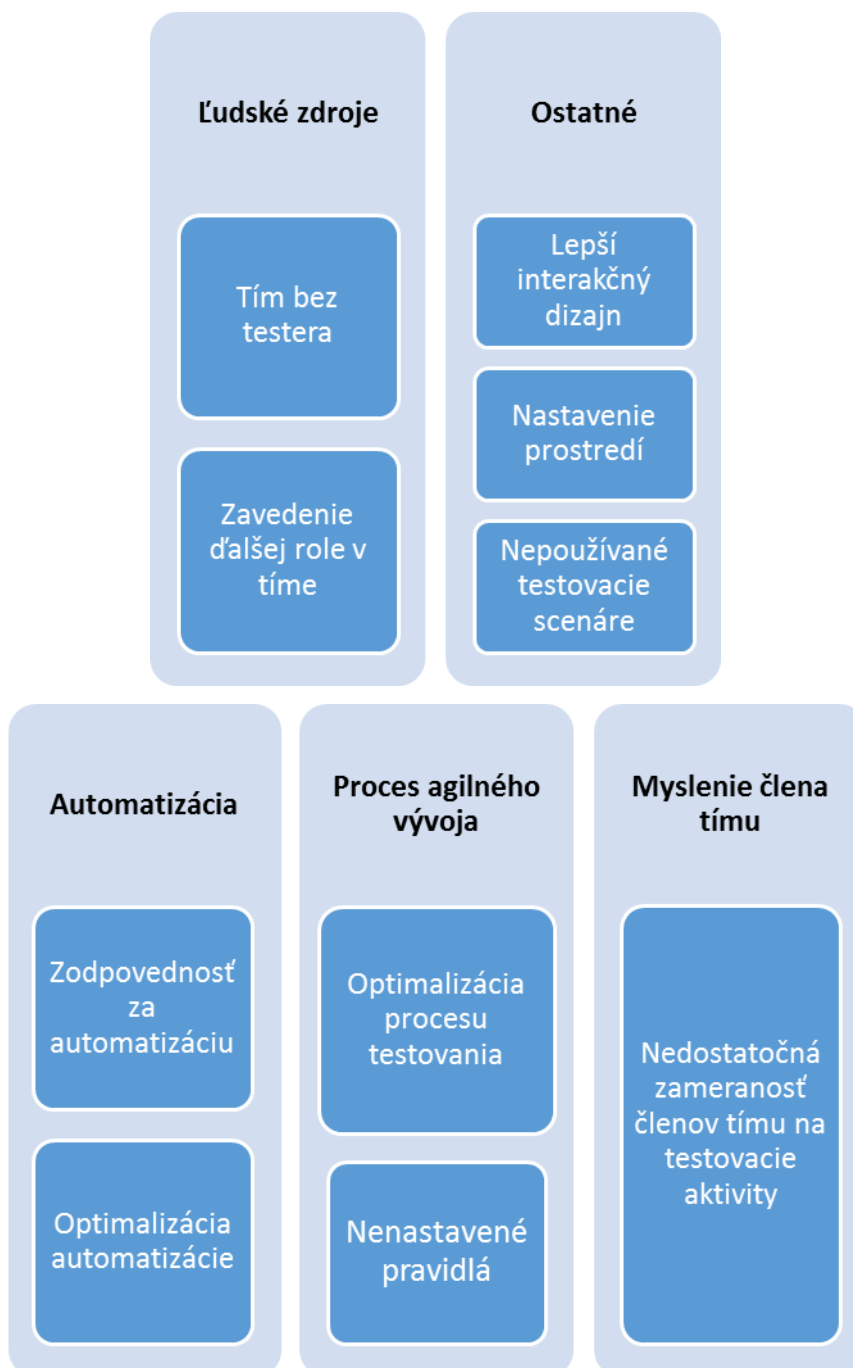
Nasledujúci diagram zobrazuje problémy extrahované z rozhovorov s jednotlivými členmi tímu 2. Členovia ich identifikovali sami ako pre nich dôležité oblasti nad ktorými by sa mal tím zamyslieť a rozhodnúť sa čo s nimi urobí. Podrobný popis problémových podoblastí obsahuje príloha C.



Obrázok 13 – Problémové oblasti tímu 2 (autor)

### 4.7.3 Identifikované problémy tímu 3

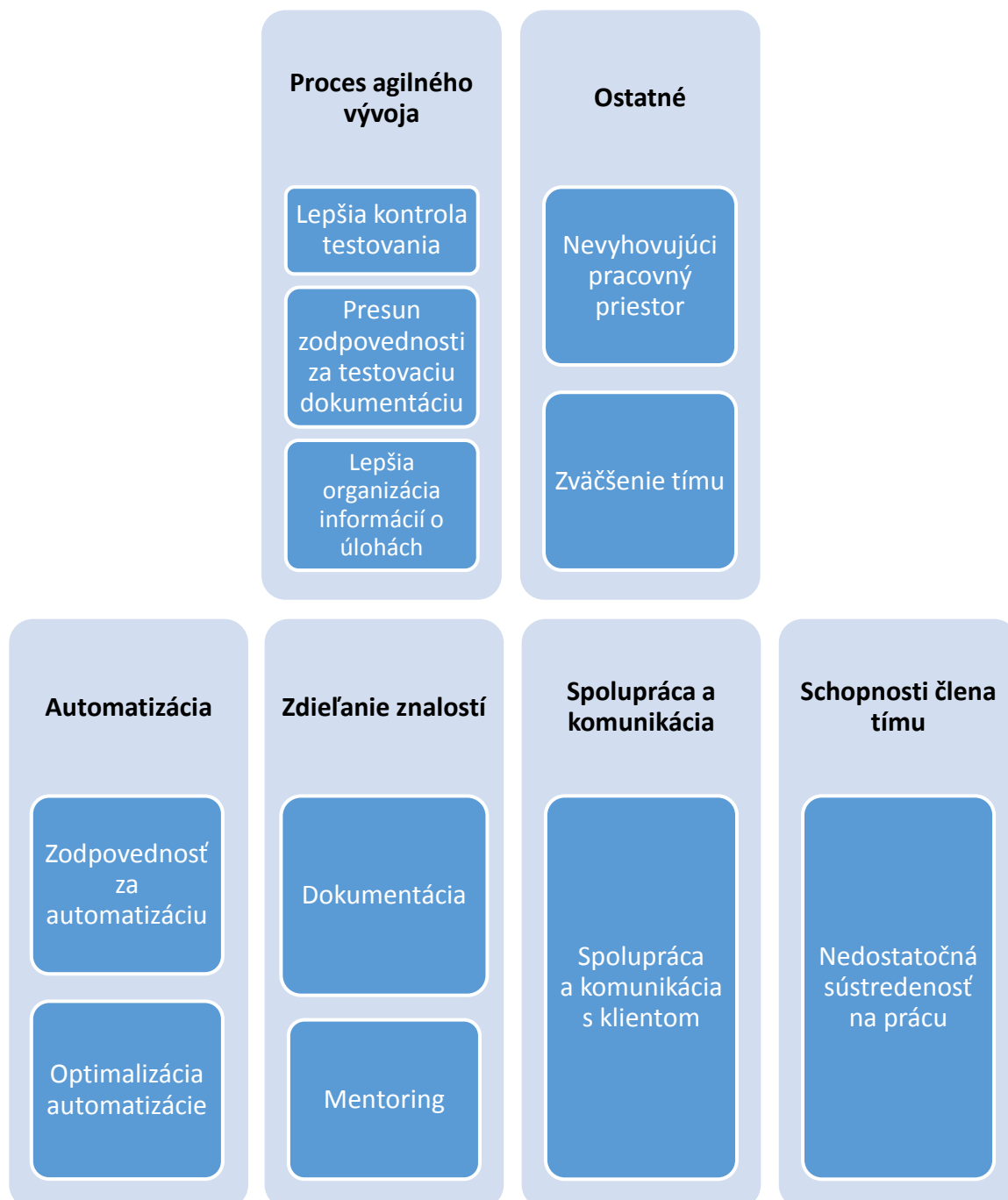
Nasledujúci diagram zobrazuje problémy extrahované z rozhovorov s jednotlivými členmi tímu 3. Členovia ich identifikovali sami ako pre nich dôležité oblasti nad ktorými by sa mal tím zamyslieť a rozhodnúť sa čo s nimi urobí. Podrobný popis problémových podoblastí obsahuje príloha C.



Obrázok 14 – Problémové oblasti tímu 3 (autor)

#### 4.7.4 Identifikované problémy tímu 4

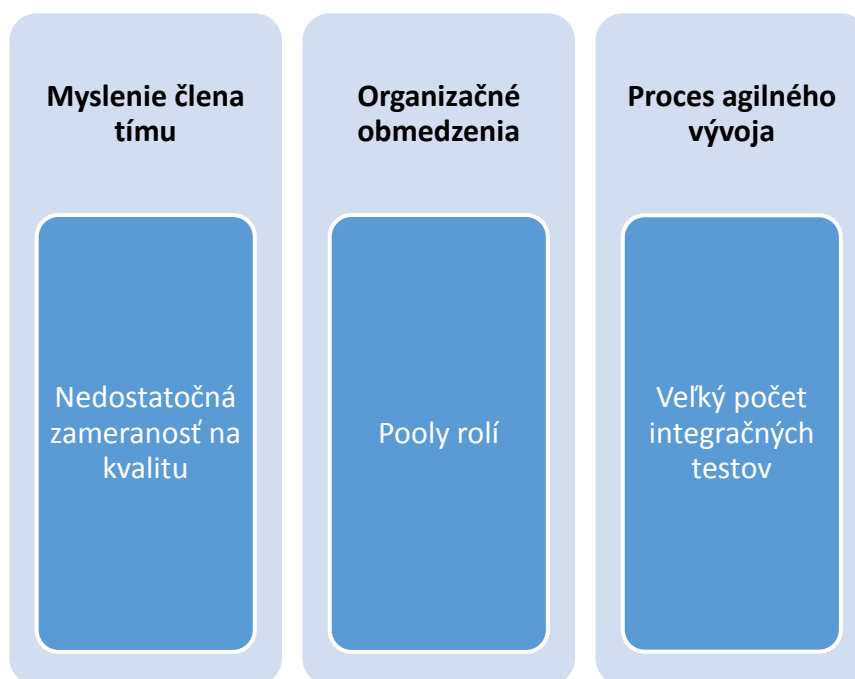
Nasledujúci diagram zobrazuje problémy extrahované z rozhovorov s jednotlivými členmi tímu 4. Členovia ich identifikovali sami ako pre nich dôležité oblasti nad ktorými by sa mal tím zamyslieť a rozhodnúť sa čo s nimi urobí. Podrobný popis problémových podoblastí obsahuje príloha C.



Obrázok 15 – Problémové oblasti tímu 4 (autor)

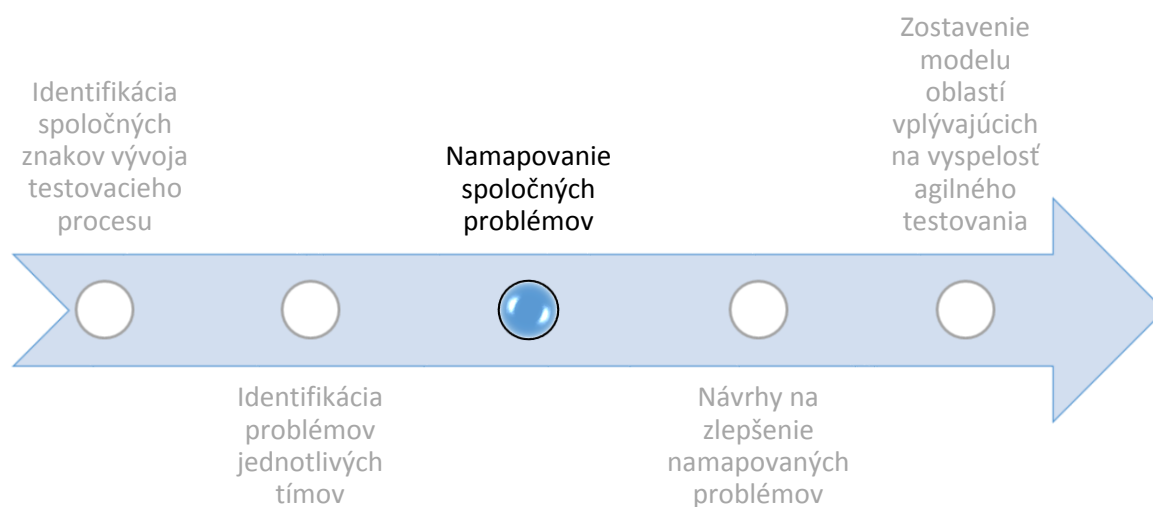
### 4.7.5 Identifikované problémy tímu 5

Nasledujúci diagram zobrazuje problémy extrahované z rozhovorov s jednotlivými členmi tímu 5. Členovia ich identifikovali sami ako dôležité oblasti, ktoré by v minulosti zlepšili. Podrobný popis problémových podoblastí obsahuje príloha C.



Obrázok 16 – Problémové oblasti tímu 5 (autor)

## 4.8 Namapovanie spoločných problémov



Obrázok 17 – Indikátor postupu výskumu krok 3 (autor)

V nasledujúcej podkapitole vytváram mapovaciu tabuľku spoločných problémových oblastí z prieniku problémových oblastí jednotlivých tímov. Za spoločné oblasti považujem problémy, ktoré sa vyskytujú aspoň v dvoch tímoch. Tieto oblasti vznikli analýzou kvalitatívnych dát za použitia konceptov zakotvenej teórie ako kategórie obsahujúce spoločné problémové podoblasti jednotlivých tímov. Namiesto pojmu kategória, ktorá sa spája so zakotvenou teóriou je použitý pojem problémová oblasť.

Tabuľka 9 obsahuje spoločné problémy skúmaných tímov. Označenie T1 až T5 identifikuje jednotlivé tímy (napríklad T1 znamená tím 1).

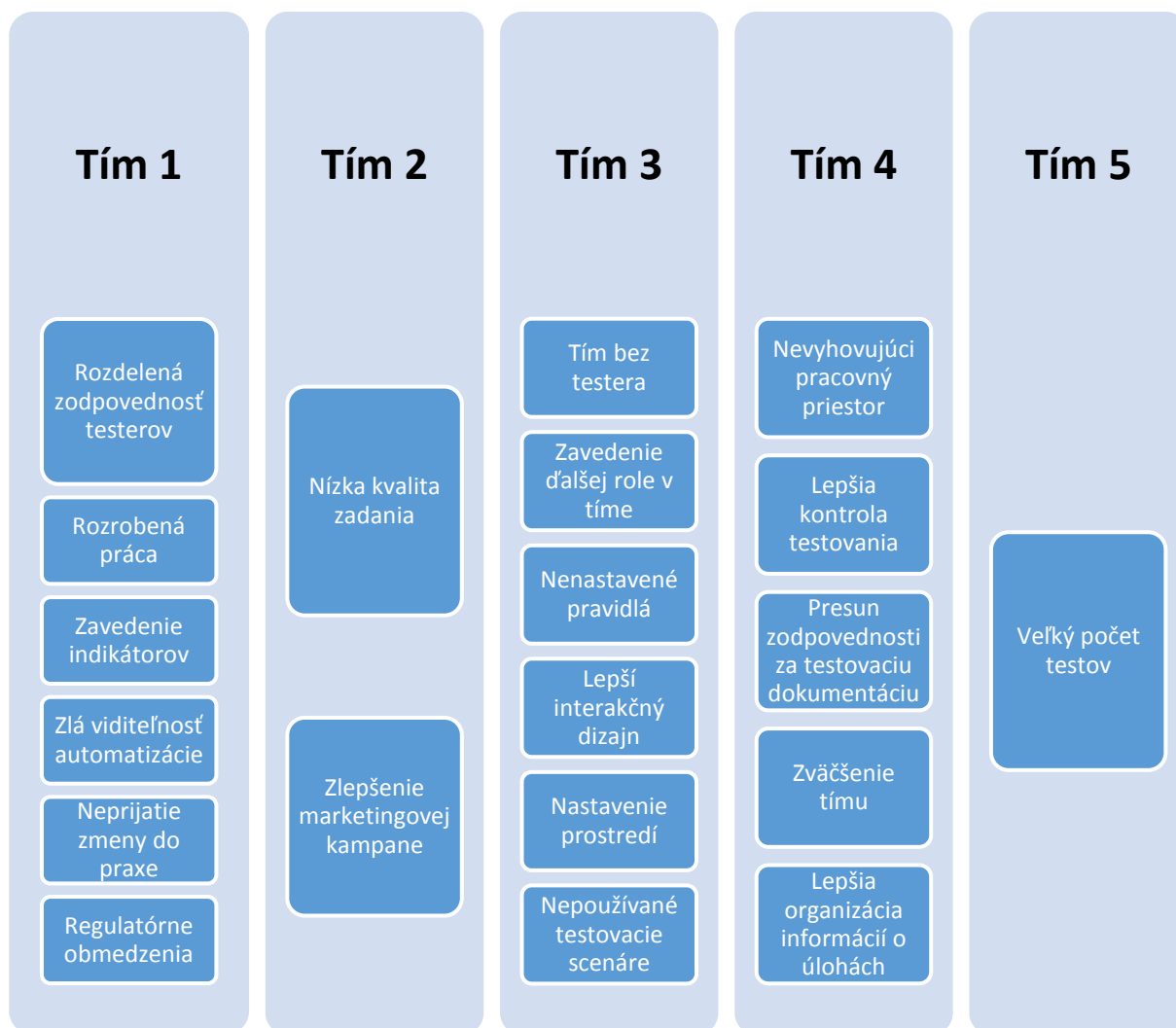
Tabuľka 9 – Mapovanie spoločných problémových oblastí (autor)

Problémová oblasť	T1	T2	T3	T4	T5
Automatizácia	X	X	X	X	
Spolupráca a komunikácia	X	X		X	
Myslenie člena tímu	X		X		X
Schopnosti člena tímu	X			X	
Zdieľanie znalostí		X		X	
Projektové obmedzenia	X	X			
Organizačné obmedzenia	X	X			X
Proces agilného vývoja	X	X	X		



Ostatné podoblasti považujem za individuality na ktoré sa zameriava práve jedna spoločnosť. Individuality sú pre spoločnosť špecifické, čo však neznamená, že iná spoločnosť o nich uvažovať nemôže.

Obrázok 18 zobrazuje individuálne problémové oblasti skúmaných tímov.



Obrázok 18 – Individuálne problémové podoblasti skúmaných tímov (autor)

## 4.9 Návrh na zlepšenie namapovaných problémov



**Obrázok 19 – Indikátor postupu výskumu krok 4 (autor)**

V tejto podkapitole sa venujem zostaveniu návrhov na zlepšenie problémových podoblastí jednotlivých tímov. Na každú problémovú podoblasť navrhujem zlepšenie z môjho pohľadu, z teoretického pohľadu podloženého odbornou literatúrou a pohľadu jednotlivých členov tímu v prípade ak návrh na zlepšenie popísali pri rozhovore.

Na riešenie individualít odporúčam použiť pracovný postup riešenia problematickej oblasti z podkapitoly 4.9.9.

### 4.9.1 Automatizácia

Tabuľka 10 zobrazuje manažérsky pohľad návrhov na zlepšenie problémových podoblastí spadajúcich do oblasti Automatizácia. Následne sú tieto návrhy podrobnejšie popísané v ďalšom texte.

Tabuľka 10 – Návrhy zlepšenia podoblastí oblasti Automatizácia (autor)

Problémová oblasť	Problémová podoblasť	Návrh zlepšenia autora	Návrh zlepšenia z odbornej literatúry	Návrh zlepšenia navrhnutý členom tímu
Automatizácia	Zodpovednosť za automatizáciu	Špecializovaná v dedikovanom tíme automatizačných testerov	Zodpovednosť za automatizáciu oddelená od zodpovednosti za testovanie (Graham a Fewster 2012)	Zdieľaná Tím 1 Špecializovaná Tím 1, 2, 3, 4
	Optimalizácia automatizácie	Experimentáciou k dosiahnutiu bodu optima spätnej väzby	Pokrytie regresných testov tak by sa tím netopil v technickom dlhu. (Gregory a Crispin 2014)	Zvýšenie pokrytia Tím 1, 2, 3, 4 Zníženie pokrytia Tím 2

Všetky tímy identifikovali, že automatizácia je dôležitou oblasťou pre zlepšenie procesu testovania a je jednou z kľúčových oblastí pre zrýchlenie spätnej väzby.

V **tíme 1** prevláda rozdielny názor na **zodpovednosť za automatizáciu** akceptačných testov.

*automation [automatizácia] je veľmi koncentrované v rukách jedného človeka, chcel by som aby sa to viacej šírilo, aby ste sa viacej učili, aby sa to posúvalo.*

*Scrum master*

*Ja by som mal svoj vlastný tím dedikovaný pre automatizáciu. [...]*

*Tester*

**Tím 2** chce presunúť zodpovednosť za akceptačné testy na vývojárov. Testerí s tým majú určité pochybnosti.

*Myslím si, že by to mohlo pomôcť, ale myslím si, že by to mohlo tiež nejak negatívne narušiť vývoj toho produktu práve napríklad akoby zo strany tých vývojárov. Budú mať o dosť menej času vyrábať ten produkt ako taký. Viem, že oni tie testy budú chcieť písať precízne, pretože sú to dobrí programátori a že im to napríklad môže zabráť veľa času, čo práve bude na úkor toho vývoja samotného a tiež si myslím, že ich to tiež nebude tak baviť a naruší to trochu tú morálku alebo náladu v tom tíme medzi vývojármi sa bojím. Takže z môjho pohľadu to nie je dobrý krok.*

*Tester*

Ako riešenie navrhujú vytvoriť dedikovaný tím zodpovedný za automatizáciu.

*Jediné, čo si ja myslím, že by sa dalo zlepšiť, a už rok sa snažíme vybudovať nejaký tím, tím čisto pre automatizáciu. Pretože, tí ľudia čo mám v tíme to samozrejme nezvládajú.*

*Test manager*

**Tím 3** chce začať s automatizáciou akceptačných testov, ktorú by mali na starosti špecializovaní testerí.

*proste zase je to i o tom, že tú automatizáciu si píšeme mi proste sami, kdežto keby to bol človek, ktorý by bol len proste len na tú automatizáciu, ako iný tester, tak už si myslím, že to má tiež inú hodnotu.*

*Vývojár*

**Tím 4** špecifikuje, že zodpovednosť za akceptačné testy by mala byť v rukách testerov.

V **tíme 5** boli automatizované integračné testy a automatizáciu akceptačných neriešili.

V rámci **optimalizácie automatizácie** chcú **tímy 1, 3 a 4** automatizovať väčšie množstvo prípadov

*Zasa čím viac regresov[regresných testov], tým lepšie viac automatizácie, lebo nemáš toľko rúk aby si robil v kuse regresiu.*

*Scrum master*

*No zlepšiť by sa to dalo reálne tak, že by sme to skúsili zautomatizovať čím viac toho, hlavne čo sa týka tej formálnej dokumentácie SDLC [dokumentácia splňajúca regulačné požiadavky] a akoby testovania releasu. Automatizácia všeobecne, proste to si myslím.*

*Vývojár*

Zatiaľ čo v **tíme 2** prevládajú rôzne názory.

*mohlo by to byť ešte viac pokryté, aby sme vedeli o tých problémoch dajme tomu skôr a viac pokryť to automatickými testami. Ale akoby som celkom spokojný.*

*Product owner*

*Automatizáciu. Za mňa by som to videl v optimalizácii počtu test casov smerom k nižšiemu počtu, pretože si myslím, že stále tam máme veci, ktoré sa testujú pomerne duplicitne, takže samozrejme by si to vyžadovalo veľa času, ale optimalizácia toho testovacieho scenára by tiež bola možnosť ako ten proces skrátiť a urýchliť.*

*Tester*

### **Návrh autora:**

Ja som toho názoru, že za automatizáciu akceptačných testov by mal byť zodpovedný dedikovaný tím testerov a ostatní testeri by sa mali sústrediť na exploratívne testovanie. Podľa môjho názoru automatizačný tester môže byť viac zastienený technickým pohľadom na aplikáciu a potom mu môžu uniknúť niektoré biznis súvislosti pri hľadaní hraničných prípadov. Na druhej strane menej technicky zameraný tester má viac biznisový pohľad a premýšľa v intenciách užívateľa, čo je žiaduce.

Čo sa týka automatizácie integračných a jednotkových testov, tam je zodpovednosť zdieľaná medzi vývojármi. Takto by to malo podľa mňa zostať, pretože vývojári majú najväčší prehľad o technickom riešení daných komponent. Akceptačné testy by podľa mňa mali byť v rukách testerov, keďže by na jednej strane zaťažili vývojárov a na druhej strane vývojári väčšinou nie sú oboznámení tak dobre s biznis kontextom ako testeri, čo vyplynulo z výskumu a pojednáva o tom podkapitola 4.11.

Čo sa týka optimalizácie automatizácie, môj názor je ten, že to záleží na veľa faktoroch a nedá sa jednoznačne určiť hranica koľko vecí máme automatizovať, aby bol spätná väzba efektívna. Tím by mal experimentovať a z vlastnej skúsenosti zistiť, kedy je spätno-väzobný cyklus optimálny. Do automatizácie by sa mal zapojiť celý tím, aby bolo pokryté čo najväčšie množstvo prípadov, ako z pohľadu akceptačných testov, tak z pohľadu jednotkových a integračných.

### Návrh inšpirovaný odbornou literatúrou:

Odborná literatúra hovorí, že zodpovednosť za automatizované testovanie by mala byť oddelená od testerskej aktivity, pretože je viac technicky zameraná.

*Udržujte role testingu a automatizácie oddelené (aj v prípade, že jedna osoba zastáva oboje role). Definujeme role dvoch disciplín, ktoré sa ukázali ako dôležité.*

*Tieto dve disciplíny znamenali, že tester by mal zostať veľmi blízko „užívateľovi“ a mal by prezentovať testovacie prípady na preskúmanie biznis manažérom. Technik automatizácie je ďaleko od biznis perspektívy a hlbšie zapojený do vytvárania vysoko technických vecí, bližšie k mentalite vývojára. Vo všeobecnosti stále udržujeme rovnaké oddelenie disciplín na dennej bázi, keďže to v praxi funguje veľmi dobre.*

*Autor experimentoval použitím čisto programátorských schopností ako spôsob prevencie od zapadnutia dobrých navrhovateľov testovacích scenárov v technikáliách automatického skriptovania, ale oddelenie vývojár verzus tester sa vždy prejaví a zaberá pomerne veľký stupeň manažovania. Zistilo sa, že táto kombinácia funguje keď sa zameriavame na dôvod využívania automatizácie – zodpovedne a pravidelne exekúovať testy požadované testerom.*

*(Graham a Fewster 2012)*

Optimalizácia automatizácie sa líši projekt od projektu ako spomínajú autorky knihy *More agile testing*, pričom nižšie vrstvy automatizačnej pyramídy z podkapitoly 2.1.4 dávajú rýchlejšiu spätnú väzbu. Množstvo regresných testov by si mal tím nastaviť tak aby sa nezačal topiť v technickom dlhu.

*Stretli sme sa s niekoľkými tímami, ktorých situácia nesesedla na originálny model automatizačnej pyramídy. Je to v poriadku. Každý tím by si mal vyvinúť model, ktorý sedí jeho potrebám. Dôležité je zapamätať si princípy skryté za pyramídou. Nižšie úrovne reprezentujú rýchlejšiu spätnú väzbu, ktorú tím potrebuje keď pridáva a aktualizuje kód. Cieľom je zahrnúť celý tím do automatizácie dostatočného množstva regresných testov tak, aby sa projekt nezačal topiť v technickom dlhu.*

*(Gregory a Crispin 2014, str. 227)*

## 4.9.2 Spolupráca a komunikácia

Tabuľka 11 zobrazuje manažérsky pohľad návrhov na zlepšenie problémových podoblastí spadajúcich do oblasti Spolupráca a komunikácia. Následne sú tieto návrhy podrobnejšie popísané v ďalšom texte.

Tabuľka 11 – Návrhy zlepšenia podoblastí oblasti Spolupráca a komunikácia (autor)

Problémová oblasť	Problémová podoblasť	Návrh zlepšenia autora	Návrh zlepšenia z odbornej literatúry	Návrh zlepšenia navrhnutý členom tímu
<b>Spolupráca a komunikácia</b>	Tímová spolupráca a komunikácia	Pracovný postup zlepšenia problémovej oblasti <i>podkapitola 4.9.9</i>	Troj krokový postup riešenia ľudských problémov <i>(Carroll 2009)</i>	Vyjadrovanie pocitov <i>Tím 1</i>
	Spolupráca a komunikácia s klientom	Vysvetlenie reálnej hodnoty a agilného vývoja klientovi	Špecifikačné workshopy <i>(Adzic 2009)</i>	Reálne ciele <i>Tím 2</i> Vysvetlenie hodnoty a agilného vývoja klientovi <i>Tím 1</i> QA komunikuje s klientom <i>Tím 4</i>

Niektorí členovia **tímu 1** považujú **tímovú spoluprácu a komunikáciu** za problematickú no niektorí sú v tomto ohľade optimistickí.

*Je to niekde v lepšej polovici. Myslím si, že ten tím je už dostatočne dlho spolu na to aby začal fungovať. Teraz to stačí už len vypilovať aby to bolo fakt super.*

*Interakčný dizajnér*

*Takže je tá komunikácia od testerov k vývojárom dôležitejšia, skôr ako nasledovať nejakú formálnu procedúru, že nájdem niečo, založím bug [...] skôr akoby komunikovať a baviť sa na normálnej úrovni a takisto z opačnej strany vývojári k testerom nebyť akože ja som vývojár, ty to otestuj a keď to nefunguje tak mi to vrátiš, keď to funguje tak fajn ja som ušetril 15 minút [...] Takže mali by sa tie role nejak prelínať.*

*Vývojár*

*Tí ľudia by mali viac so sebou komunikovať a uvedomiť si ako sa chovajú [...] Tie veci sú v sebe, nie vonku. [...]*

*Tester*

Ostatné tímy nemajú s tímovou spoluprácou a komunikáciou problém.

**Člen tímu 1** odporúča ako návrh na zlepšenie vyjadrovanie pocitov.

*Ak máš problém [...] máš vyjadriť svoj pocit, ale vyjadriť v rámci neublížovania ostatným [...] pretože to je tímová hra. [...] Že niečo robíš v tom projekte a dodáš vec, ale nemáš pocit, že ťa niekto vníma, že niekto vníma tvoju prácu a si z toho sklamaný, tak vyjadríš ten pocit.*

Tester

**Komunikácia s klientom** je problémom, ktorý riešia všetky spoločnosti.

*Nie som úplne spokojný s tým, keď sa vyvíjajú nejaké veľké tlaky odniekiaľ kde napríklad nie je úplne jasné ako to funguje a takto sú nastavené nejaké veľké požiadavky, ktoré nie sú splniteľné.*

Tester

*Ale jednoznačne proste musíme zlepšiť to, aby sme sa zamerali na potreby a nenaháňali featury.*

Product owner

**Tím 1** zlepšil komunikáciu s klientom tak, že im pomohli identifikovať ich skutočné problémy a skutočnú hodnotu. Taktiež ich zasvätili do agilného vývojového procesu a jeho princípov.

*No a vlastne tu sa s nimi začalo pracovať a nejak ich ohýbať, že vlastne takýmto štýlom nie. Začalo sa s nimi riešiť, že čo je ten problém, akoby z tej UX [užívateľský zážitok] perspektívy, že boli workshopy, kde sa snažili im vysvetliť, že musíme nájsť problém, čo je MVP [minimum viable product] a kde je skutočná hodnota a čo chceme dosiahnuť.*

*Teraz tá diskusia si myslím, že už sa nám to darí pretláčať do nejakej roviny, že skôr sa bavíme o tých potrebách než o tom návrhu a že už sú aj akoby prístupný na to, že pracujeme nejakým štýlom, že ten agile proste nejak funguje inak než boli zvyknutí, že keď tam chcú niečo pridať, tak je potrebné niečo odobrať alebo urobiť nejaké kompromisy a nemať všetko naraz ale nejak postupne a dostávať feedback [spätnú väzbu] [...] Tam sa to nejak zlepšuje, od začiatku to bolo dosť také no zlé. Myslím, že je to lepšie. [...]*

Vývojár



*Čo sa tiež zlepšilo je prístup stakeholderov [zákazníka]. Oni v podstate pochopili ako mi to chceme robiť a snažia sa nám viac pomôcť, nie sú vo svojej škatuľku a viac sa otvorili, viac sa s nami bavia, viac na nás spoliehajú, viac nám veria, už sa nehádajú kvôli každej blbine ale nechávajú spústu vecí na nás. A i to pozvedlo celý ten vývoj toho produktu. [...] Vzťah product ownera s tímom sa zlepšil za ten rok [...]*

*Interakčný dizajnér*

*Zahrňujeme biznis viac a viac do vývojového procesu, takže ich zahrňame na refinement sessions [stretnutia tímu na ktorých sa revidujú funkcie v product backlogu]. My a UX [tím interakčných dizajnérov] vplývame na nich viac a viac, takže validujeme skorý vnútorný proces, koncepty predtým než sa dostanú do sprint backlogu [sprioritizovaný zoznam funkcií požadovaných od produktu] alebo vývoja.*

*Product owner*

Mať reálne ciele od klienta bol návrh zlepšenia identifikovaný v **tíme 2**

*Mali by sa asi viac stretávať s realitou v zmysle, chceme produkt releasovať čo najčastejšie, chceme continuous integration, ale vieme, [...] že toho skôr nie sme schopní dosiahnuť, ale chceli by sme proste mať release každý týždeň v horizontu roku napríklad. Takéto hmatateľné ciele, ktoré sú možné. Takže viac tie predstavy toho manažmentu mať realnejšie nech človek má pocit, že za tým človek môže fakt ísť, pretože keď sú stanovené úplne nereálne tak ani tá snaha nie je taká.*

*Tester*

V **tíme 4** padol návrh na rozšírenie rolí komunikujúcich s klientom o QA tím.

*Ja by som určite povedal, že by sme mali viacej komunikovať my [QA tím] s klientom, pretože momentálne s nimi komunikuje len product owner. On akoby má toho strašne veľa a často sa ho opýtam, no ale toto sa mi nezdá a on povie oni to tak nechceli tak to tak neurobíme. Ale zase tam je ten problém, že čo keď oni by to tak chceli, ale nevedia o tom.*

*Tester*

**Návrh autora:**

Na vyriešenie problému s komunikáciou odporúčam celotímový prístup. Keďže výpovede jednotlivých tímov nie sú jednotné, každý tím by sa mal pokúsiť spoločne identifikovať príčinu problému a na tú zostaviť akčný plán zlepšenia ako popisuje podkapitola 4.9.9. Cieľom je nájsť skutočnú príčinu komunikačného problému a dohodnúť sa na riešení, ktoré budú akceptovať všetci členovia tímu. Bez tejto akceptácie a prijatia do praxe riešenie nemôže byť úspešné, keďže sa s ním musí stotožniť celý tím.

Čo sa týka riešenia problému komunikácie s klientom, ja sa prikláňam k riešeniu tímu 1. Myslím si, že keď tím získa náklonnosť klienta a vysvetlí mu, že všetci sú na rovnakej lodi, tak klient bude s tímom viac spolupracovať a vychádzať mu v ústrety. Na zlepšenie spolupráce by som odporúčal preto čo najčastejšiu fyzickú prítomnosť klienta s vývojovým tímom. Najlepším riešením je byť v jednej budove, no u medzinárodných spoločností je tento stav skoro nemožný, takže tu odporúčam častú organizáciu workshopov a dennú komunikáciu s klientom v prípade zistených nejasností v požiadavkách či prípadnom návrhu vylepšení.

**Návrh inšpirovaný odbornou literatúrou:**

Odporúčanie z odbornej literatúry (Carroll 2009) na riešenie komunikačných problémov hovorí o trojkrokovom procese riešenia ľudských problémov.

*Prečo použiť proces? Opakovaný proces pomôže vytvoriť rutinný zvyk a ušetrí čas kedykoľvek sa vyskytne, podľa vášho úsudku, ľudský problém. Môžete ľahko objaviť čo vás trápi, obrátiť to na pracovný problém a rýchlo sa vrhnúť do akčných krokov presne ako by ste riešili akýkoľvek iný biznis problém.*

*Tu je trojkrokový proces ktorý môžete použiť:*

- 1. Vyjadrite problém*
- 2. Definujte problém a vyjadrite pozorovateľné fakty*
- 3. Rozhodnite sa nad akčnými krokmi*

*(Carroll 2009)*

Crispin vo svojej knihe odporúča na riešenie „problémov myslenia“ (ľudských problémov) Problem solving leadership čo sú praktiky, ktoré majú pomôcť s riešením problémových situácií z psychologického hľadiska.

S problémami **spolupráce a komunikácie s klientom** by mali pomôcť špecifikačné workshopy.

*Aktivity ako sú špecifikačné workshopy fungujú skvele, keď niekto skúsený vo facilitácii pomáha viesť diskusiu. Facilitátor, ktorý nie je zahrnutý do tvorby požiadaviek by bol ideálny, ale akýkoľvek člen tímu, ktorý má kľúčové schopnosti myslenia, ktorý napríklad dokáže prinútiť rôzne role pracovať spolu a ostať zameraný na spoločný cieľ, môže byť prínosom ak je potreba. Facilitátory špecifikačných workshopov pomáhajú stakeholderom [zainteresovaným stranám] s nastavením biznis cieľov a pomáhajú vývoju a zákazníkemu tímu kolaboratívne definovať scope, ktorým sa definované ciele dosiahnu.*

(Adzic 2009)

### 4.9.3 Projektové obmedzenia

Tabuľka 12 zobrazuje manažérsky pohľad návrhov na zlepšenie problémových podoblastí spadajúcich do oblasti Projektové obmedzenia. Následne sú tieto návrhy podrobnejšie popísané v ďalšom texte.

Tabuľka 12 – Návrhy zlepšenia podoblastí oblasti Projektové obmedzenia (autor)

Problémová oblasť	Problémová podoblasť	Návrh zlepšenia autora	Návrh zlepšenia z odbornej literatúry	Návrh zlepšenia navrhnutý členom tímu
<b>Projektové obmedzenia</b>	Pevne stanovené deadliney	Deadline stanovené spoločne medzi klientom a tímom	Špecifikačné workshopy (Adzic 2009)	Zrušenie deadlineov <i>Tím 1</i>
			Ponechanie pevne stanovených deadlineov - iterácií (Cooke 2014)	Ponechanie deadlineov <i>Tím 2</i>

Táto kategória znemožňuje lepšiemu dosiahnutiu vyspelosti agilného testovania. Tím by sa mal pokúsiť odstrániť všetky obmedzenia, alebo nájsť alternatívne riešenie.

Identifikovaná bola jedna oblasť spoločná pre tímy 1 a 2

#### 1. Pevne stanovené deadliney

Člen **tímu 1** navrhuje deadliney zrušiť

*Čo by určite pomohlo kvalite je uľahčenie [...] v zmysle deadline. My sme mali v release jedna naprosto striktné daný dátum [...] Povedal som si, že v release dva už bude viac času na plánovanie a hýbanie s premennými. To sa nestalo. My máme ten deadline daný plus mínus úplne rovnako [...]*

*Interakčný dizajnér*

## Člen tímu 2 nemá problém s deadlineami

*U nás napríklad bola veľká otázka deadliney u vývojárov. [...] Tam to je také, niekto je za deadliney, niekto je za prioritizáciu feature a ja osobne vlastne ešte neviem či som pre to, tie deadliney zrušiť alebo nie. Asi na niekoho to pôsobí psychicky blbo ale na mňa nie. Myslím si, že je to občas potrebné, že to tlačí tých ľudí. [...]*

Vývojár

### Návrh autora:

Podľa môjho názoru by deadliney mali byť stanovené spoločným úsilím klienta a tímu. Tím musí pochopiť dôležitosť stanovených časových míľnikov z biznis pohľadu a v prípade nereálneho pohľadu má pomôcť klientom tento pohľad usmerniť. Bez deadlineov podľa mňa prevláda chaos a klient nemôže získať vyššiu istotu v dodávaný produkt. Zároveň zrušenie deadlineov pridáva neurčitnosť do projektu, ktorá by podľa mňa mala byť minimálna aby bol projekt úspešný a klient spokojný.

Tím by mal rozlišovať medzi biznis deadlineami a deadlineom Sprintu. V druhom prípade si tím sám stanovuje prácu a v prípade, že sa jedná o problém s týmto druhom deadlineu, mal by tím prehodnotiť svoje sily a naplánovať prácu do Sprintu reálnejšie podľa svojich schopností a nemal by byť príliš optimistický. Tím by si mal tiež uvedomiť, že v prípade nereálnych odhadov a bez poznania svojich síl sa stav dodávaných funkcií nezlepší a klient môže stratiť istotu v dodávaný produkt.

### Návrh inšpirovaný odbornou literatúrou:

Literatúra hovorí, že pevným deadlineom je jedna iterácia.

*Sila pevných deadlineov je to, prečo agilné prístupy štruktúrujú prácu aby bola garantovaná a dodaná vo fixných iteráciách s pevnými review stretnutiami [stretnutie kde tím prezentuje implementovanú funkcionality zainteresovaným stranám] na konci každej iterácie[krátky vývojový cyklus]. Tieto pevné deadliney môžu zaručiť, že aktivity nebudú odsunuté v prospech práce vnímanej pracovníkmi viac urgentne. Sú rovnomerne nadizajnované aby zaistili, že výsledky review stretnutia na konci každej iterácie nie sú pohyblivým sviatkom, ktorý môže byť odsúvaný v prospech iných priorít. V agilnom svete pevné deadliney predstavujú neustálu pripomienku pre dodávateľský tím [produkuje softvér požadovaný zákazníckym tímom] a pocit nutnosti, ktorý flexibilné (pohyblivé) deadliney [termíny] neposkytujú.*

(Cooke 2014)

#### 4.9.4 Organizačné obmedzenia

Tabuľka 13 zobrazuje manažérsky pohľad návrhov na zlepšenie problémových podoblastí spadajúcich do oblasti Organizačné obmedzenia. Následne sú tieto návrhy podrobnejšie popísané v ďalšom texte.

Tabuľka 13 – Návrhy zlepšenia podoblastí oblasti Organizačné obmedzenia (autor)

Problémová oblasť	Problémová podoblasť	Návrh zlepšenia autora	Návrh zlepšenia z odbornej literatúry	Návrh zlepšenia navrhnutý členom tímu
Organizačné obmedzenia	Pooly rolí	Pracovný postup zlepšenia problémovej oblasti <i>podkapitola 4.9.9</i>	Použitie maticovej štruktúry <i>(Crispin a Gregory 2008)</i> Vytvorenie testerskej komunity <i>(Crispin a Gregory 2008)</i>	Integrácia testerského tímu do produktového tímu <i>Tím 1</i> Zrušenie testerských rolí <i>Tím 5</i>
	Distribučované tímy	Presun zodpovednosti, ktorá obmedzuje činnosť tímu kde vzniká toto obmedzenie	Zviditeľniť alebo odstrániť závislosti medzi distribuovanými tímami <i>(Gregory a Crispin 2014)</i>	Presun zodpovednosti do tímu <i>Tím 1</i> Presun tímov na jedno miesto <i>Tím 2</i>

Táto kategória znemožňuje lepšiemu dosiahnutiu vyspelosti agilného testovania. Tím by sa mal pokúsiť odstrániť všetky obmedzenia, alebo nájsť alternatívne riešenie.

Patria sem:

##### 1. Distribuované tímy, ktoré medzi sebou spolupracujú

Navrhované riešenie **tímu 1** je presun zodpovednosti do tímu.

*Celé to, že to je mimo v nejakom inom tíme je gap [medzera], ktorý zasa, to je dobré adresovať tým, že tú kompetenciu toho celého release manažmentu, by sme dostali do tímu.*

*Scrum master*

Navrhované riešenie **tímu 2** je presun tímov na rovnaké miesto.

*Takže teraz je našou snahou mať tímy alokované na jednom mieste. [...]*

*Test manager*

## 2. Organizačné pooly rolí z ktorých sa vytvára tím

Navrhované riešenie **tímu 1** integrácia testovacieho tímu do projektu

*Tak aby ten testovací tím bol viac integrovaný v projekte. Aby nebolo, že tu máš testing, tu máš inženýring.*

*Tester*

V **tíme 5** zrušili rolu testera. Zo všetkých sa stali softvéroví inžinieri ale s touto zmenou nie sú spokojní. Aj keď oficiálne sa rola testera zrušila, neoficiálne testeri viac inklinujú k testerským činnostiam.

*No pretože nikto na to testovanie už akoby nedohliadal. Bol pocit, že sa to vyrieši samo. Ono sa to nevyrieši samo. Asi na to niekto musí dohliadať a tí čo boli testeri majú stále to v sebe, takže budú asi viac inklinovať k tomu, že budú písať testy, alebo dohliadať na to [...]*

*Vývojár*

### Návrh autora:

Čo sa týka dislokovaných tímov, ktoré medzi sebou spolupracujú a majú medzi sebou závislosti, tie by sa mali snažiť odstrániť všetky závislosti, ktoré ich blokujú v práci. Často tieto závislosti sú potrebné, a preto je podľa mňa vhodné dohodnúť sa na presune zodpovednosti medzi tímami do tímu, ktorý táto závislosť blokuje, a to v prípade, ak to blokovaný tím akceptuje. V prípade, že blokovaný tím neakceptuje presun zodpovednosti, oba tímy by sa mali dohodnúť na riešení, ktoré bude efektívne a akceptovateľné pre obe strany a zároveň nebude blokovať výkon práce ani jedného z nich.

Ja by som neodporúčal zrušenie testerských organizácií, pretože sa tým stratí platforma pre predávanie špecializovaných testerských znalostí. Testeri budú viac sústredení na technické aspekty a sústredenosť na kvalitu sa zníži. Podľa mňa organizačný presun nevyrieši príčinu problému a v tejto oblasti by sa mal tím zamerať na hlbšie skúmanie príčin, prečo túto oblasť niektorí vnímajú za problematickú. Znovu môže tím použiť pracovný postup na zlepšenie problematickej oblasti z podkapitoly 4.9.9.

**Návrh inšpirovaný odbornou literatúrou:**

Literatúra radí, že veľké organizácie by mali používať maticovú štruktúru pre agilné projekty.

*Organizácie s veľkými projektami alebo veľkým počtom projektov, ktoré sa dejú simultánne dosahujú úspech pri využívaní maticovej štruktúry. Ľudia z rôznych funkčných oblastí sa nakombinujú aby vytvorili virtuálny tím a reportujú späť do svojich individuálnych organizačných štruktúr. Vo veľkých organizáciách pool testerov sa môže presúvať z projektu na projekt.*

(Crispin a Gregory 2008, str. 64)

Taktiež je vhodné vytvoriť komunitu testerov na zdieľanie nových nápadov a špecializovaných testerských znalostí

*Pokým testerí sú súčasťou tímu, ich každodenná práca je manažovaná rovnako ako zvyšok práce projektového tímu. Tester môže nové nápady priniesť z väčšej testerskej komunity, ktorá zahŕňa testerov v iných projektových tímoch naprieč veľkej organizácie. Všetci testerí môžu zdieľať znalosti a nápady.*

(Crispin a Gregory 2008, str. 64)

Závislosti medzi distribuovanými tímami by sa mali podľa literatúry zviditeľniť alebo odstrániť.

*Buďte si vedomí závislostí medzi skupinami. Ak čakáte na výstupy ľudí v inej časovej zóne a nemajú ich pripravené, musíte čakať aspoň 24 hodín. Členovia tímu, môžu počas čakania začať pracovať na iných úlohách, čoho dôsledkom je prepínanie úloh medzi viacerými záväzkami. Diskutujte o potenciálnych závislostiach na plánovacích stretnutiach a nájdite cestu ako ich zviditeľniť alebo eliminovať. Onlinové monitorovacie nástroje môžu byť v tomto ohľade nápomocné.*

(Gregory a Crispin 2014, str. 305)

### 4.9.5 Proces agilného vývoja

Tabuľka 14 zobrazuje manažérsky pohľad návrhov na zlepšenie problémových podoblastí spadajúcich do oblasti Proces agilného vývoja. Následne sú tieto návrhy podrobnejšie popísané v ďalšom texte.

Tabuľka 14 – Návrhy zlepšenia podoblastí oblasti Proces agilného vývoja (autor)

Problémová oblasť	Problémová podoblasť	Návrh zlepšenia autora	Návrh zlepšenia z odbornej literatúry	Návrh zlepšenia navrhnutý členom tímu
<b>Proces agilného vývoja</b>	Optimalizácia procesu testovania	Zapojenie testerov počas celého životného cyklu funkcie a jej okamžité pokrytie automatickými testami	Zapojenie celého tímu do vytvárania testovacích prípadov pri plánovacích stretnutiach a ich následné pokrytie kódom a testami. (Crispin a Gregory 2008)	Testovanie počas celého životného cyklu funkcie <i>Tím 1</i> Automatické testy po vývoji funkcie <i>Tím 2</i> Automatické testy pred vývojom funkcie <i>Tím 2</i> Automatické testy počas vývoja funkcie <i>Tím 3</i>

Táto kategória reflektuje tímovú snahu zlepšovať svoj vývojový proces súčasťou ktorého je i proces testovania. Všetky tímy považujú za oblasť na zlepšenie optimalizáciu súčasného testovacieho procesu.

*nie sme v stave aby sme automatizovali to, čo vyvíjame v sprinte, je to minimálne jeden sprint pozadu, čo je menší waterfallik [vodopádový vývoj] zasa hej? Zasa raz sme v menšom waterfalliku.*

*Scrum master*

*QA ľudia sa teda využívajú na doháňanie technického dlhu, vždy ti vzniká nejaký technický dlh v produkte, u nás je to hlavne automatizácia. [...]*

*Test manager*

*Problém s tými testami je, že akoby doháňajú ten vývoj, že keď sa niečo vyvinie, tak sa k tomu integračné testy [...] vyvinú až spätne*

*Vývojár*



**Tím 1** odporúča zapojenie testera počas celého životného cyklu úlohy.

*Teraz stále vidíme to, že task je stále na testerovi. Malo by to byť tak, že podme robiť na tasku [úlohe] cez [...] celý jeho životný cyklus. "Začínam vyvíjať, môžeme sa stretnúť? Podme to prediskutovať. Ok, odchytím ťa neskôr aby som ti niečo ukázal. " Je to otestované okamžite, task [úloha] je teraz pripravený na formálny a neformálny testing [testovanie], tak ho môžeš passnúť [uskutočniť a uzatvoriť].*

*vidím ich pri pair-programmingu [párové programovanie] [...] ako tester a developer sedia vedľa seba. Tak to ja vidím. [...]*

*Product owner*

Niektorí členovia **tímu 2** odporúčajú písanie automatických testov hneď po vývoji funkcie, niektorí už na začiatku vývoja funkcie

*Teraz sa prechádza k tomu, alebo ideálny stav je keď sa otestuje napríklad tá user story a tester na to napíše nejaký test case [testovací scenár] a úplne ideálny stav je aby sa ešte v tej chvíli ten test zautomatizoval [...]*

*Tester*

*[...] Ja si myslím, že by sme mali držať nejakú konzistenciu v tých projektoch, aby boli ako nejako rovnako percentuálne pokryté minimálne. [...] Občas sa stáva, že sa na tie testy zabúda a že sa dopisujú až neskôr. Mohlo by sa to naplánovať skôr a písať to už vlastne s tým produktom.*

*Vývojár*

**Tím 3** chce vytvárať automatické testy spolu s kódom

*Chceme aby vývojári začali písať testy spolu s kódom a všetky tieto veci, ale vieme, že sa to nestane hneď, nie všetci sú na to pripravení, musíme ešte zápasit' aj s časom a produktami, ktoré máme, takže postupne.*

*Product owner*

**Návrh autora:**

Myslím si, že do vývoja novej funkcie by sa mali testerí zapájať už od začiatku aby mali aktuálne informácie a nezdržovali vývojový proces neustálym upresňovaním toho, čo bolo povedané. Všetci členovia tímu by sa mali sústrediť na písanie testov pred vývojom, kedy sa celý tím sústredí najprv na prípady, ktoré by mohli v systéme nastať z rôznych pohľadov. Testerí využívajú pohľad zákazníka a vývojári technický pohľad, kedy by mali pred vytvorením kódu navrhnuť jednotkové a integračné testy, ktoré tento kód pokryjú. Takto získajú väčšiu zameranosť na kvalitu produktu a tiež nebudú musieť tráviť veľa času pri následnej oprave chyby, ktoré nájdú až po implementácii funkcie, keďže test odchytil túto chybu už pri písaní kódu. Automatické testy by sa mali teda vytvárať súčasne s vývojom aby sa zaistila čo najrýchlejšia spätná väzba.

**Návrh inšpirovaný odbornou literatúrou:**

V literatúre sa spomína, že celý tím by mal na plánovacích stretnutiach hovoriť o testovacích scenároch a na tie následne napísať kód a testy.

*TDD [vývoj riadený testami] a SDD [vývoj riadený užívateľskými príbehmi] vedie tímy k tomu, aby mysleli najprv na testy. Počas plánovacích stretnutí hovoria o testoch a najlepšom spôsobe ako ich urobiť. Navrhujú kód, tak aby prešiel testami, takže testovateľnosť nie je nikdy problém. Automatická sada testov rastie zároveň s bázou kódu poskytujúc záchrannú sieť pre konštantný refaktoring [zmena kódu bez zmeny funkcionality]. Je dôležité aby celý tím praktizoval TDD [Test drive development] a konzistentne písal jednotkové testy, lebo záchranná sieť bude deravá.*

*(Crispin a Gregory 2008, str. 435)*

### 4.9.6 Myslenie člena tímu

Tabuľka 15 zobrazuje manažérsky pohľad návrhov na zlepšenie problémových podoblastí spadajúcich do oblasti Myslenie člena tímu. Následne sú tieto návrhy podrobnejšie popísané v ďalšom texte.

Tabuľka 15 – Návrhy zlepšenia podoblastí oblasti Myslenie člena tímu (autor)

Problémová oblasť	Problémová podoblasť	Návrh zlepšenia autora	Návrh zlepšenia z odbornej literatúry	Návrh zlepšenia navrhnutý členom tímu
<b>Myslenie člena tímu</b>	Nedostatočná zameranosť členov tímu na testovacie aktivity	Šírenie povedomia o dôležitosti testovacích aktivít	Vizualizácia testovacích aktivít na udržanie povedomia (Gregory a Crispin 2014)	Väčšia zameranosť na výkonnosť <i>Tím 1</i>  Zvýšenie povedomia o dôležitosti automatizácie <i>Tím 3</i>
	Nedostatočná zameranosť na kvalitu	Šírenie povedomia o dôležitosti kvality produktu	Párovanie testerov s kódermi (Gregory a Crispin 2014)	Zvýšenie povedomia o dôležitosti kvality produktu <i>Tím 5</i>

K zvýšeniu vyspelosti testovacích procesov prispieva i zmena myslenia jednotlivých členov tímu.

**Tím 1** identifikuje zhoršenie v oblasti výkonnostných testov kvôli nesústredenosti členov tímu na tento typ testovania

*Možno performance testing [testovanie výkonnosti] je niečo, čo odskočilo, lebo to zrazu nerobil niekto iný, kto sa na to pred tým nik nesústredil, a to zasa neni, že testerí hej, ale to bolo to, že celý tím sa zase menej sústredil na performance testy [testy výkonnosti]. A teraz to vyplávalo trošku na povrch.*

*Scrum master*

**Tím 3** zdôrazňuje za kľúčové povedomie o dôležitosti automatizácie pri jej zavádzaní.

*Plus ešte dôležité je aby bolo povedomie o tom aké to je strašne dôležité. To beriem tiež ako dôležitú vec.*

*Product owner*

**Tím 5** ukazuje, že zameranie členov tímu na kvalitu zvyšuje testovacie aktivity.

*Pokiaľ ten tím [...] alebo tí, čo ten tím vedú majú v hlave to, že potrebuje sa dodávať kvalitný produkt tak tomu testovaniu dávajú priestor.*

Vývojár

#### **Návrh autora:**

Tieto dôvody ma presvedčili, že myslenie a zameranosť jednotlivých členov tímu má veľký dopad na vyspelosť testovacích procesov i celého tímu a vývoja. Tím by mal byť oboznámený a mal by si pripomínať prečo je daná aktivita, ktorú vykonáva, dôležitá a mal by byť na ňu sústredený. Zároveň by mal mať povedomie o dôležitosti kvality produktu a pripomínať si ju.

Na podporu povedomia o činnostiach môžu slúžiť rôzne techniky uvádzané v odbornej literatúre podľa toho, čo bude tím používať. Úspešné sú vizuálne pomôcky na zobrazenie stavu testovacích aktivít umiestnené na mieste kde sa tím zdržuje väčšinu svojho pracovného času. So sústredení na kvalitu môže vývojárom pomôcť použitie TDD (Test driven development), alebo spolupráca s testerami pri tvorbe testov a iné praktiky.

#### **Návrh inšpirovaný odbornou literatúrou:**

V literatúre Crispin odporúča vizuálne pomôcky, ktoré pomáhajú udržať povedomie tímu o testovacích aktivitách.

*Zistujeme, že vizuálne pomôcky pomáhajú udržať celý tím vedomý o testovacích aktivitách, ktoré sa musia uskutočniť a sledovať, čo ešte musí byť urobené. Jednou efektívnou metódou vizualizácie je mind mapping [myšlienkové mapy].*

(Gregory a Crispin 2014, str. 98)

So zameraním na kvalitu môže kóderom pomôcť párovanie s testerami.

*Párovanie s testerami pomáha kóderom získať testovaciu uvedomelosť.*

(Gregory a Crispin 2014, str. 67)

### 4.9.7 Schopnosti člena tímu

Tabuľka 16 zobrazuje manažérsky pohľad návrhov na zlepšenie problémových podoblastí spadajúcich do oblasti Schopnosti člena tímu. Následne sú tieto návrhy podrobnejšie popísané v ďalšom texte.

Tabuľka 16 – Návrhy zlepšenia podoblastí oblasti Schopnosti člena tímu (autor)

Problémová oblasť	Problémová podoblasť	Návrh zlepšenia autora	Návrh zlepšenia z odbornej literatúry	Návrh zlepšenia navrhnutý členom tímu
<b>Schopnosti člena tímu</b>	Nerovnomerná participácia členov na tímových záležitostiach a produkte	Uvedomenie potreby participácie	Mindset člena tímu robiť veci lepšie (Crispin a Gregory 2008)	Každý sa rovnako podieľa na tímových záležitostiach a produkte <i>Tím 1</i>
	Nízke technické znalosti testerov	Uvedomenie potreby použitia znalosti	Mindset člena tímu robiť veci lepšie (Crispin a Gregory 2008)	Viac technických znalostí u testerov <i>Tím 1</i>
	Nedostatočná sústredenosť na prácu	Uvedomenie potreby použitia schopnosti	Mindset člena tímu robiť veci lepšie (Crispin a Gregory 2008)	Väčšia sústredenosť testerov na prácu <i>Tím 4</i>

Táto kategória bola vytvorená identifikáciou problémov tímov s nedostatočným využitím všetkých schopností jednotlivých členov tímu, ktoré keby sa využili tak by prispeli k vyššej vyspelosti testovania a celého vývojového procesu.

**Tím 1** identifikoval nerovnomernú participáciu členov na tímových záležitostiach a produkte. Niektorí členovia si myslia, že technické znalosti nie sú v testerskom tíme dostatočné.

**Tím 4** tvrdí, že keď je uvoľnenejšia atmosféra, testovací tím nevyužíva svoje schopnosti naplno a vznikajú tak chyby.

#### Návrh autora:

Myslím si, že toto je individuálna záležitosť každého člena tímu, ktorý by sa mal byť vedomý toho, že by mal využívať svoje schopnosti naplno a prispievať nimi v tímovom snažení na vytvorenie hodnoty pre zákazníka. Podľa mňa je vhodné sa otvorene baviť v tíme o potrebných schopnostiach. Potrebu týchto schopností si potom členovia uvedomia a v prípade, že nimi disponujú mali by byť aktívni a použiť ich. V prípade, že daná schopnosť chýba, je vhodné nájsť odborníka s touto schopnosťou, ktorý ju do tímu bude šíriť. Schopnosti testerov by mali odpovedať kvalitám testera uvedených v podkapitole 2.1.2.

### Návrh inšpirovaný odbornou literatúrou:

V literatúre sa hovorí, že v mindsete jednotlivých členov tímu by malo byť snaženie stále vykonávať lepšiu prácu.

*Hľadanie spôsobov uskutočňovať prácu lepšie je súčasťou mindsetu [myslenia]agilného testera. Samozrejme celý tím by mal myslieť týmto spôsobom, pretože hlavným základom agilu je, že tím sa vždy snaží robiť lepšiu prácu.*

*(Crispin a Gregory 2008, str. 27)*

### 4.9.8 Zdieľanie znalostí

Tabuľka 17 zobrazuje manažérsky pohľad návrhov na zlepšenie problémových podoblastí spadajúcich do oblasti Zdieľanie znalostí. Následne sú tieto návrhy podrobnejšie popísané v ďalšom texte.

Tabuľka 17 – Návrhy zlepšenia podoblastí oblasti Zdieľanie znalostí (autor)

Problémová oblasť	Problémová podoblasť	Návrh zlepšenia autora	Návrh zlepšenia z odbornej literatúry	Návrh zlepšenia navrhnutý členom tímu
<b>Zdieľanie znalostí</b>	Dokumentácia	Vytvorenie dostatočnej dokumentácie na to, aby nebola obmedzená činnosť členov tímu	Minimálne množstvo dokumentácie spĺňajúce definované formálne požiadavky organizácie <i>(Crispin a Gregory 2008)</i>	Všetky pravidlá produktu <i>Tím 2, 4</i>
	Mentoring	Priradenie mentora v prípade neskúseného tímu	Mentor pre nových zamestnancov <i>(Crispin a Gregory 2008)</i>	Skúsenejší kolega <i>Tím 4</i>

**Tímy 2 a 4** identifikovali potrebu predávania znalostí za dôležitú a to v podobe

- dokumentácie
- skúseného kolegu – mentora

*Potom ďalšia vec je písať všetky tie featury [funkcie], dokumentáciu k tomu aby všetci chápali čo to má robiť a ako sa to má tvoriť.*

*Product owner*

*a tam je ešte ten problém, že k tomu projektu nie je žiadna dokumentácia. [...] Ako nejaké náhodné Exceli a Wordy sú, ale máme problém s tým, že tam je strašne veľa pravidiel a nie je to zosumarizované do nejakej dokumentácie, takže v tomto je to také blbé.*

*Tester*

*Ale možno tak osobne ako tester by som chcel niekoho, nejakého teamleada v QA tíme, niekoho, kto by mohol nie povedať čo robiť, ale nejaký best practice [najlepšia praktika] nám zo svojej skúsenosti ponúknuť, nejaké veci, ktoré by nám pomáhali sa rozvíjať, ako pokračovať v tom svojom profesijnom raste [...]*

*Tester*

#### **Návrh autora:**

Ja by som odporučil zaznamenávanie informácií, ktoré v prípade, že by neboli zdokumentované, spomalili by, alebo zastavili činnosť člena tímu, alebo celého tímu. Dokumentácia by sa mala písať už od začiatku projektu. Tím by si mal zodpovednosť za zostavenie dokumentácie rozdeliť medzi seba a vytvoriť dokumentačné tasky v systéme na podporu vývoja. To, že je vývoj agilný totiž neznamená, že je bez dokumentácie.

Predávať znalosti by si mali všetci členovia tímu navzájom s tým, že ak tím nie je veľmi skúsení mal by si do tímu prizvať mentora, ktorí sa so skúsenosťami rád podelí. Jednotlivé organizácie, či už formálne (ukotvené v organizácii) alebo neformálne (komunity) odborníkov na špecifické oblasti (testerské, vývojárske, interakčný dizajn, atď.) by mali tiež zdieľať medzi svojimi členmi odborné znalosti a mali by disponovať skúsenými pracovníkmi ochotnými v prípade potreby pomôcť.

### **Návrh inšpirovaný odbornou literatúrou:**

Literatúra v prípade potreby dokumentácie odporúča zaradiť dokumentačný proces do testovacej stratégie a určiť minimum dokumentácie uspokojujúcej formálne požiadavky na dokumentáciu.

*Ak jedinou vecou, ktorá v tíme chýba je dokumentácia, potom uvažuj o zahrnutí tvojho procesu do testovacej stratégie dokumentácie. Spýtaj sa aké je minimálne množstvo dokumentácie, ktoré máš mať aby si uspokojil CMMI [capability maturity model integration] požiadavky.*

*(Crispin a Gregory 2008, str. 90)*

Čo sa týka mentoringu, je dobré mať mentora, ktorý by pomohol novým zamestnancom.

*Je nápomocné pridelit' učiteľa alebo mentora ako prvú líniu podpory pre nových zamestnancov; dvaja môžu fungovať ako predvolený pár.*

*(Crispin a Gregory 2008, str. 40)*

#### **4.9.9 Pracovný postup zlepšenia problémovej oblasti**

Tento pracovný postup som zostavil s prihliadnutím na tímovú zodpovednosť v agilnom vývoji, kedy by si mal tím sám nájsť cestu, ktorá je pre neho najvhodnejšia, tú prijať a začať aplikovať v praxi.

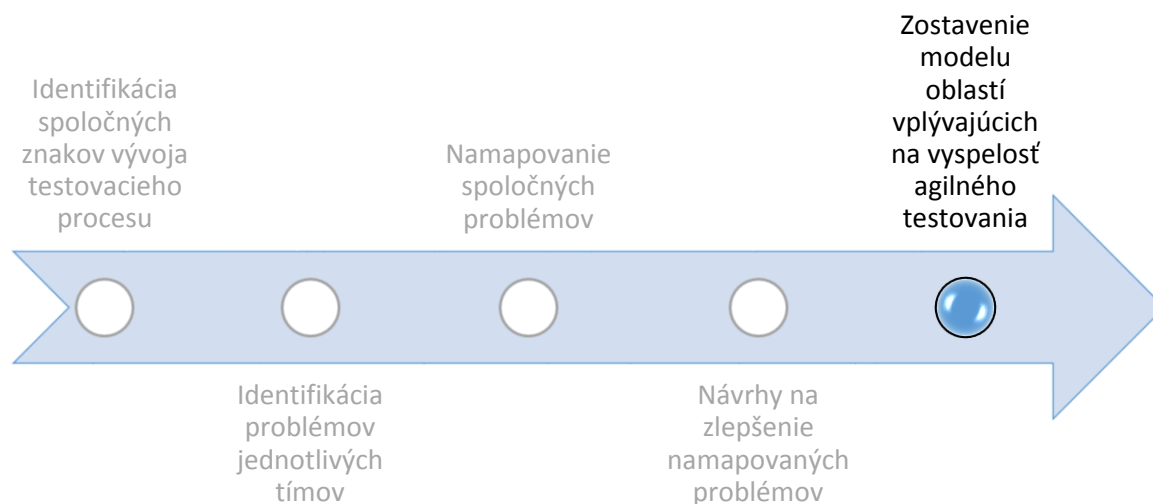
Postup predpokladá, že na základe názorov každého člena tímu sa zostaví akčný plán na zlepšenie problémovej oblasti, ktorý sa bude tímovým učením optimalizovať.

Nasleduje zoznam krokov pracovného postupu:

- 1. Validácia navrhnutých oblastí zlepšenia celým tímom**
- 2. Vypracovanie alternatív na riešenie problémov**
- 3. Ohodnotenie alternatív každým členom tímu**
- 4. Výber víťaznej možnosti**
- 5. Akčný plán implementovania vybranej možnosti**
- 6. Zavedenie do praxe**
- 7. Overenie úspešnosti zavedenej možnosti**
- 8. Optimalizácia alebo zmena možnosti**



## 4.11 Model oblastí ovplyvňujúcich vyspelosť agilného testovania



Obrázok 20 – Indikátor postupu výskumu krok 5 (autor)

Na základe analýzy dát z výskumu usudzujem, že vyspelosť agilného testovania sa podľa praktikov (jednotlivých členov skúmaných agilných tímov) nedá definovať ako postupnosť určitých krokov, ktoré by sa mali vykonať, aby sme dosiahli zefektívnenie testovacích procesov, ale vyspelosť agilného testovania je o neustálom učení sa tímu a riešení individuálnych problémov, pričom si tím rozvíja vlastnú cestu a vlastné procesy. Teoreticky by sa dali nájsť určité obecné kroky, no otázkou je, či by boli praktikmi akceptované. V tomto „tímovom“ vývoji som našiel niekoľko spoločných znakov, ktoré ale nie sú finálne a tímy by mali naďalej hľadať oblasti pre zlepšenie, ako vyplýva zo spoločného znaku kontinuálneho učenia a zlepšovania.

Tabuľka 18 zobrazuje mapovanie spoločných oblastí z podkapitoly 4.8 a spoločných znakov vývojového procesu z podkapitoly 4.6.6 na prvky zostaveného modelu z obrázku 21.

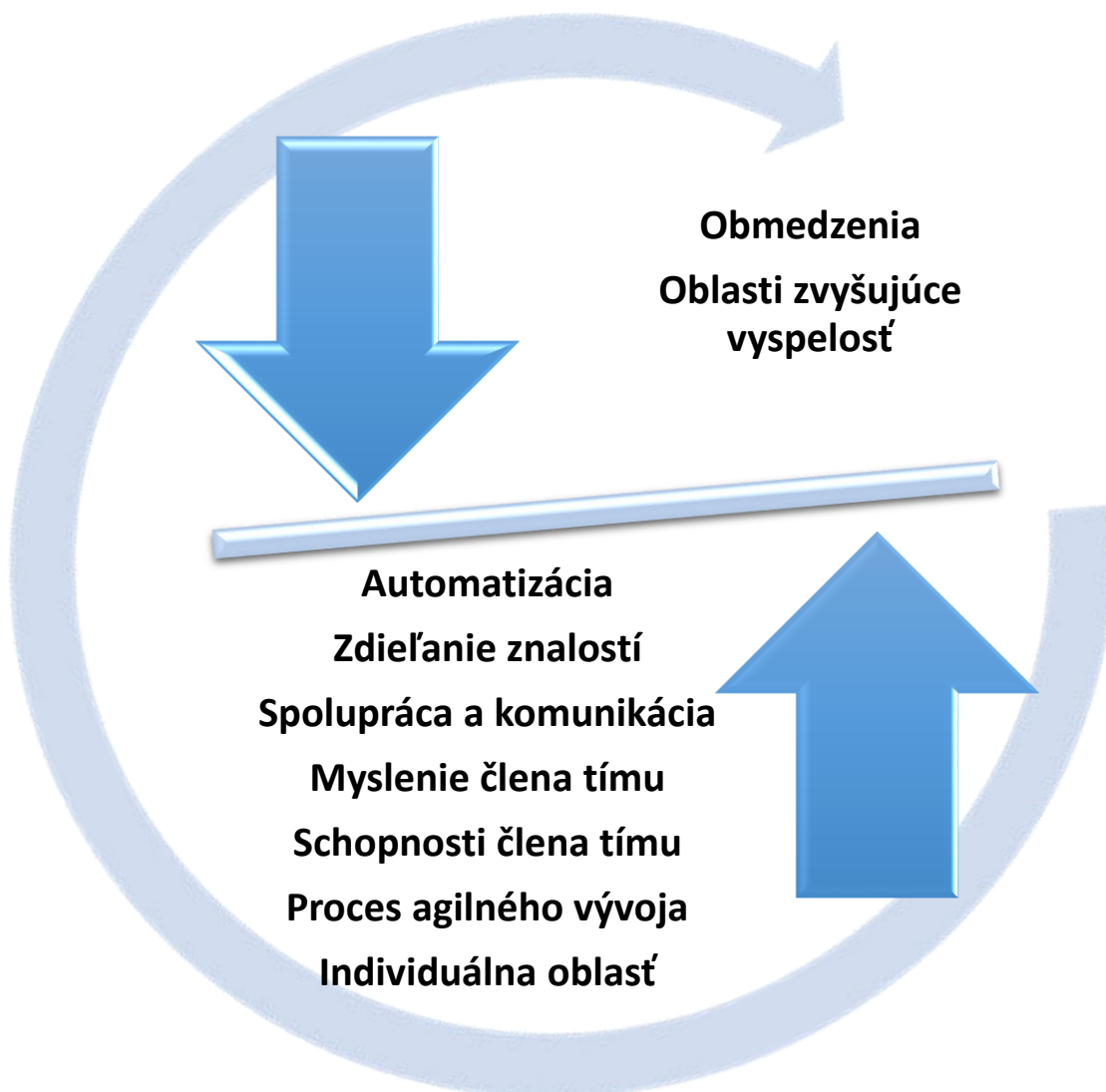
Tabuľka 18 – Mapovanie spoločných oblastí a znakov na prvky modelu (autor)

Spoločná oblasť / Spoločný znak	Prvok v modeli oblastí ovplyvňujúcich vyspelosť agilného testovania	Primárny účinok prvku*
Automatizácia	Automatizácia	+
Spolupráca a komunikácia	Spolupráca a komunikácia	+
Myslenie člena tímu	Myslenie člena tímu	+
Schopnosti člena tímu	Schopnosti člena tímu	+
Zdieľanie znalostí	Zdieľanie znalostí	+
Projektové obmedzenia	Obmedzenia	-
Organizačné obmedzenia	Obmedzenia	-
Proces agilného vývoja	Proces agilného vývoja	+
Kontinuálne učenie	Šípka ohraničujúca model	+

\* Prvky primárne zvyšujúce agilnú vyspelosť testovania sú označené znamienkom (+) a prvky primárne znižujúce agilnú vyspelosť testovania sú označené znamienkom (-)

Model bol zostavený zo spoločných problémových oblastí skúmaných tímov a spoločných znakov vývoja procesov testovania. Šípka ohraničujúca model reprezentuje kontinuálne učenie, ktoré prebieha naprieč všetkými oblasťami, ktoré sa jej pôsobením postupne menia. Táto zmena môže byť pozitívna, alebo negatívna, no v oboch prípadoch je pre vyspelosť prospešná, pretože tím získava nové skúsenosti a je viac pripravený na nové zmeny.

Agilná vyspelosť testovania je podľa výsledkov analýzy kvalitatívnych dát o hľadani zlepšení v oblastiach ako je automatizácia testov, spolupráca a komunikácia, zdieľanie znalostí, myslenie a schopnosti členov tímu, vývojový proces a ďalších individuálnych oblastiach. Tieto oblasti by mali zvýšiť vyspelosť testovania, no je nutné ich vyvažovať aby sa nedostali do oblastí blokujúcich zvýšenie vyspelosti. Tieto oblasti tiež tvoria základy pilierov testovania z podkapitoly 2.1.5 a prispievajú k zvýšeniu istoty v kvalitný softvér. Zároveň však tím musí znižovať pôsobenie oblastí blokujúcich vo zvýšení vyspelosti, ktorými sú projektové, organizačné obmedzenia a v určitom štádiu i samotné oblasti, ktoré pomáhajú vyspelosť zlepšiť.



Obrázok 21 - Model oblastí ovplyvňujúcich vyspelosť agilného testovania (autor)

Model ukazuje, že agilná vyspelosť testovania zahŕňa najmä „mäkké prvky“ súvisiace s osobou člena tímu, akými sú zdieľanie znalostí, spolupráca a komunikácia, myslenie člena tímu a schopnosti člena tímu. Tieto oblasti ukazujú, že členovia jednotlivých tímov považujú tieto prvky za dôležité pri pôsobení na kvalitu produktu. Podrobný popis prvkov modelu je uvedený v prílohe B, kde sú popísané problémové oblasti, ktoré tieto prvky reprezentujú.

## 4.12 Spoločné znaky prínosu testera v agilnom tíme

V tejto podkapitole sa venujem zodpovedaniu otázky na doplnkový cieľ, ktorá sa zaoberá dôležitosťou testera v agilnom tíme. Z analýzy kvalitatívnych dát vyplynulo, že všetci respondenti okrem jedného považujú testera za neoddeliteľnú súčasť tímu a považujú jeho prítomnosť v tíme za dôležitú. Jeden respondent nepovažuje dôležitú testerskú rolu, ale mindset testera, ktorým by mal člen tímu disponovať. Tímy, ktoré experimentovali s odstránením testerov z tímov a tímy bez testerov tvrdia, že tím si uvedomil prínosy testerov až po ich odstránení čo indikuje spoločný znak kontinuálneho učenia a tiež oblasť myslenie a schopnosti členov tímu, ktoré vplyvajú na výslednú kvalitu produktu.

Všetci respondenti sa zhodujú, že mindset testera a vývojára je odlišný a jeho neprítomnosť vplyva na kvalitu a rýchlosť dodávky produktu. Tieto odlišnosti považujem za spoločné znaky prínosu testera v prípade, že o nich hovorili aspoň dvaja členovia tímu a uvádzam ich v tabuľke 19. Niektoré znaky, ktoré z analýzy dát vyplynuli videli respondenti odlišne, preto ich za prínosy nepovažujem.

**Tabuľka 19 – Spoločné znaky prínosu testera v agilnom tíme (autor)**

Spoločný znak prínosu testera	Popis
Má iný pohľad	Poskytuje druhý pár očí a pozerá sa na aplikáciu z inej perspektívy
Má viac času na testovanie	Sústredí sa na rôzne prípady, ktoré môžu nastať a má viac času na dôkladné testovanie ako ostatní členovia tímu
Zvyšuje čas na vývoj	Vývoj bez testera by bol pomalší, zodpovednosť za testovanie sa rozprestrie do tímu
Tester ako kouč	Zastávajú rolu konzultantov v oblasti testovania a pomáhajú ostatným členom tímu sústrediť sa na kvalitu
Má pohľad užívateľa	Pozerá sa na aplikáciu z pohľadu užívateľa a overuje prípady, ktoré pre užívateľa prinášajú hodnotu
Má technický pohľad	Okrem užívateľského či biznis pohľadu sa vyzná sa i v technických veciach aplikácie
Je záchytnou sieťou	Zachytáva chyby aby sa nedostali do produkcie
Poskytuje hĺbkové testovanie	Tester pri svojej práci využíva skúsenosti a testovacie praktiky čím odhaľuje kritické testovacie scenáre

## 5 DISKUSIA

Podľa Buchalcevej (2005) je v tradičnom vývoji najznámejším príkladom hodnotenia softvérových procesov Model zrelosti softvéru (CMM). Spojením a rozšírením tohoto modelu je Integrovaný model zrelosti (CMMI), ktorý podporuje spôsob zlepšenia softvérových procesov priebežne alebo kontinuálne. Zrelosť procesu je definovaná ako úroveň procesu, pri ktorej je proces explicitne definovaný, riadený, meraný, kontrolovaný, efektívny a v prípade, že ju organizácia dosiahne zvyšuje sa produktivita a kvalita i výkon procesu. Podstatný význam na zlepšení procesu má kľúčová oblasť procesu. Tieto oblasti určujú procesy na ktoré by sa mala organizácia zamerať aby zvýšila kvalitu softvéru. Softvérový proces je chápaný ako sada činností, metód, praktík a transformácií, ktoré ľudia používajú na vývoj a údržbu softvéru a ďalších produktov. (Buchalcevová 2015) Z predchádzajúceho vyplýva, že tradičný vývoj predpokladá, že na zlepšenie kvality je nutné zlepšiť procesy vývoja a údržby softvéru.

Z analýzy kvalitatívnych dát usudzujem, že zlepšenie procesov vývoja je len jednou oblasťou, ktorá prispieva k výslednej kvalite produktu a predstavuje ju oblasť proces agilného vývoja. Agilný vývoj sa sústreďuje viac na ľudí v tíme a zlepšenie oblastí ako spolupráca a komunikácia, myslenie a schopnosti členov tímu, zdieľanie znalostí, ktoré reprezentujú skôr „mäkké znalosti“, ktoré podľa výsledkov výskumu majú podstatný vplyv na zvýšenie kvality výsledného produktu, a preto by im mala byť venovaná zvýšená pozornosť. Taktiež sa tu namiesto obecného postupu riešenia problémov používa individuálny postup, vytvorený kontinuálnym učením členov tímu. V agilnom vývoji je dôležitá čo najrýchlejšia spätná väzba, ktorá sa tiež podieľa na zvýšení kvality produktu a reprezentuje ju oblasť automatizácia.

Oblasti, ktoré vznikli z analýzy dát sú prienikom *základov* v modeli pilierov testovania, praktík budovania istoty a kľúčových faktorov úspešnosti. Vytvorený model oblastí ovplyvňujúcich vyspelosť agilného testovania zároveň potvrdzuje a prepája teoretické poznatky rôznych modelov, čím vytvára ich rozšírenie integráciou ich prvkov do spoločného modelu. Zároveň z analýzy dát vyplýva, že na vyspelosť agilného testovania sa nemôžeme pozeráť len zo zameraním na testovacie aktivity a testovacie problémy, ale uvedomiť si, že testovanie je časťou celku za ktorý je zodpovedný každý člen tímu. Vyspelosť agilného testovania je teda viazaná na celý produktový ale i zákaznícky tím a ľudský faktor, namiesto sústredenia sa len na zlepšenie procesu vývoja softvéru.

V tabuľke 20 zobrazujem prepojenie výsledkov výskumu s teoretickými poznatkami.

Tabuľka 20 – Prepojenie výsledkov výskumu vyspelosti agilného testovania s teóriou (autor)

Oblasti ovplyvňujúce vyspelosť agilného testovania	Oblasť v modeli pilierov testovania	Praktiky budovania istoty	Kľúčové faktory úspešnosti
<b>Automatizácia</b>	Efektívna automatizácia		Automatizácia regresných testov
<b>Spolupráca a komunikácia</b>	Spolupráca a komunikácia		Spolupráca so zákazníkmi
<b>Myslenie a schopnosti členov tímu</b>	Zručnosti a skúsenosti		Myslenie agilného testera
<b>Zdieľanie znalostí</b>	Spolupráca a komunikácia		Celotímový prístup
<b>Obmedzenia</b>		Citlivosť na kontext	
<b>Proces agilného vývoja</b>		Používanie príkladov	
<b>Kontinuálne učenie</b>	Testovacie techniky	Kontinuálne učenie	
<b>Exploratívne testovanie</b>	Testovacie techniky	Exploratívne testovanie	
<b>Individuálna oblasť</b>		Vytvorenie zdieľanej vízie	

Ak sa pozrieme na spoločné znaky prínosu testera, ktoré boli identifikované a popísané v podkapitole 4.11, ich zhodu nájdeme so zručnosťami agilného testera uvedenými a popísanými v podkapitole 2.1.2. Tabuľka 21 mapuje zručnosti z teórie na spoločné znaky prínosu testera, ktoré vyplynuli z výskumu. Keďže oblasť schopnosti člena tímu bola identifikovaná ako oblasť podieľajúca sa na zvyšovaní kvality, nasledujúce spoločné znaky predstavujú zoznam znalostí, ktorými by mali testerí disponovať aby boli pre tím prínosom. Uvedené znalosti z praktickej skúsenosti autoriek Gregory a Crispin (2008) (2014) boli teda výskumom verifikované.

Z výskumných dát je zrejmé, že existuje rozdiel medzi myslením vývojára a testera. Tester má iný pohľad, príliš technicky nezastienený a premýšľa skôr opačne ako vývojár. Snaží sa hľadať i prípady, ktoré sú netypické a nezaručujú pozitívny priebeh testu. Oproti vývojárovi má viac času na využitie rôznych testovacích techník a hľadanie chýb. Jeho prítomnosť umožňuje vývojárovi sústrediť sa na svoju prácu. Je užívateľsky orientovaný a chce spolu s tímom dodať kvalitný produkt, s ktorým bude užívateľ spokojný. Preto pri práci aplikuje užívateľský pohľad a objaví ním i chyby, ktoré nie sú definované v akceptačných kritériách.

Keďže disponuje hlbšími znalosťami zo svojej profesie, je schopný v testovaní aplikovať techniky, ktoré ho zefektívnia a pomôžu tak odchytiť chyby ešte pred nasadením aplikácie do produkčného prostredia. Môže tiež viesť ostatných členov tímu k efektívnemu používaniu týchto schopností a zdieľať s nimi svoje skúsenosti. Disponuje i širšími technickými znalosťami čím vie prispieť do tímu aj v oblastiach ako je automatizácia, continuous integration a iné.

**Tabuľka 21 – Spoločné znaky prínosu testera v agilnom tíme (autor)**

Spoločný znak prínosu testera	Znalosť testera
Má iný pohľad	odlišné myslenie
Má viac času na testovanie	organizujúci
Zvyšuje čas na vývoj	organizujúci
Tester ako kouč	koučovacie a načúvacie schopnosti
Má pohľad užívateľa	znalosť biznis domény
Má technický pohľad	automatizačné a kódovacie znalosti obecné technické znalosti vývojové prostredia testovacie prostredia kontinuálna integrácia a systém na správu kódu
Je záchytnou sieťou	odlišné myslenie
Poskytuje hĺbkové testovanie	techniky testovacieho návrhu

## 6 Záver

Ústrednou témou diplomovej práce bolo definovať agilnú vyspelosť testovania k čomu som najprv musel pochopiť agilnú vyspelosť v podkapitole 2.2 a čo pojem agilné testovanie obnáša. Tento pojem je do hĺbky priblížený v podkapitole 2.1. Z kvalitatívneho výskumu som získal spoločné problémové oblasti súvisiace s vyspelosťou agilného testovania a spoločné znaky vývoja procesu testovania, ktoré pomáhajú vyspelosť testovania zvýšiť. Z týchto poznatkov som priblížil moje vnímanie agilnej vyspelosti testovania v podkapitole 4.10.

### 6.1 Zodpovedanie výskumných otázok

V tejto podkapitole hodnotím zodpovedanie výskumných otázok definovaných v metódach výskumu a uvádzam dôvody ich hodnotenia.

- **Čo je vyspelosť v agilnom testovaní?**

Túto otázku som zodpovedal zostavením modelu obecných oblastí, ktoré ovplyvňujú vyspelosť agilného testovania v podkapitole 4.10. Model zobrazuje, že agilná vyspelosť nemôže byť definovaná ako postupnosť krokov, ktoré vedú k zlepšeniu testovacieho procesu ale v agilnom prostredí vyspelosť je nutné vnímať ako proces neustáleho učenia tímu a hľadanie vlastnej cesty optimalizácie procesov.

- **Ako vieme zlepšiť vyspelosť agilného testovania?**

Za zodpovedanie tejto otázky považujem návrh zlepšení spoločných problémových podoblastí zostavený v podkapitole 4.9 pre každý tím. Návrhy boli zostavené zo skúseností autora, odbornej literatúry a návrhov členov skúmaných tímov. Tieto návrhy môžu pomôcť zvýšiť vyspelosť testovacích procesov v skúmaných tímoch.

- **Aký stupeň vyspelosti dosahujú skúmané tímy?**

Z výstupov prvej otázky vyplýva, že pojem stupeň by bol v agilnom vývoji nepoužiteľný a tím by sa nemal porovnávať s ostatnými ale aktívne hľadať zlepšenia, ktoré budú najviac vyhovovať jednotlivým členom tímu. Túto otázku považujem teda tiež za zodpovedanú.

- **Sú testerí potrební v agilných tímoch?**

Túto otázku sa mi podarilo zodpovedať v podkapitole 4.11, ktorá sa venuje vnímaniu testerov v agilných tímoch a zdôvodneniu ich prínosov. Všetky tímy považujú testerov za dôležitých a nevedia si predstaviť, že kvalita produktu by sa zlepšila keby v tíme neboli. Odpoveďou na túto otázku je, že testerí sú potrební v skúmaných agilných tímoch.



## 6.2 Prínosy práce a naplnenie cieľov

Prínosmi práce sú nasledujúce výstupy, ktoré vznikli z analýzy kvalitatívnych dát zobrazené v tabuľke 22. Tím sa nemusí zdržiavať identifikáciou problémových oblastí, pretože tie sú explicitne definované vo výstupe identifikácia problémových oblastí a podoblastí jednotlivých tímov. Zároveň pomáhajú s uvedomením si problémov a potreby reagovať na ne. Návrhy na zlepšenie problémov môžu následne tímom poskytnúť nezainteresovaný pohľad, návod, ako postupovať pri ich riešení a tiež ako jednu z variant riešení. Teoretickým prínosom v podobe výstupu zakotvenej teórie je model oblastí ovplyvňujúcich vyspelosť agilného testovania, ktorí ukazuje, že vyspelosť v agilnom vývoji chápeme odlišne od vyspelosti definovanej v tradičnom vývojovom procese. Tímto poznatkom sa tiež rozšírili teoretické poznatky o agilnom vývoji a práca tiež sumarizuje a prepája poznatky z predchádzajúcich prác a štúdií zameriavajúcich sa na oblasť agilného testovania. Je taktiež zdrojom pre zdieľanie znalostí a skúseností skúmaných agilných tímov záujemcom o agilný vývoj. Práca tiež podáva pohľad na myslenie testera a jeho prínos kde dokazuje, že sa jeho myslenie sa odlišuje od myslenia vývojára. Podáva teda dátami podložený dôkaz o dôležitosti testerov v agilných tímoch.

**Tabuľka 22 – Prínosy práce - výstupy z kvalitatívnych dát (autor)**

Výstup práce	Odkaz
Identifikácia problémových oblastí a podoblastí jednotlivých tímov	4.7.1 Identifikované problémy tímu 1 4.7.2 Identifikované problémy tímu 2 4.7.3 Identifikované problémy tímu 3 4.7.4 Identifikované problémy tímu 4 4.7.5 Identifikované problémy tímu 5
Zostavenie návrhov na zlepšenie spoločných problémových podoblastí jednotlivých tímov z troch pohľadov <ol style="list-style-type: none"> <li>1. pohľad autora</li> <li>2. pohľad odbornej literatúry</li> <li>3. pohľad členov skúmaných tímov</li> </ol>	4.9 Návrh na zlepšenie namapovaných problémov
Model oblastí ovplyvňujúcich vyspelosť agilného testovania	4.11 Model oblastí ovplyvňujúcich vyspelosť agilného testovania
Rozšírenie znalosti o vyspelosti agilného testovania	2 TEÓRIA 4 POZNATKY VÝSKUMU
Spoločné znaky prínosu testera v agilnom tíme	4.12 Spoločné znaky prínosu testera v agilnom tíme
Náhľad do vývoja procesov jednotlivých tímov a zdieľanie týchto tímových skúseností	4 POZNATKY VÝSKUMU

Tabuľka 23 zobrazuje metodické postupy, ktoré som v práci vytvoril. Keďže som aplikoval zakotvenú teóriu, ktorá je pôvodne určená na sociologické problémy a použil som ju na analýzu informatického problému, musel som metodiku trochu prispôbiť. Výstupom sú dva metodické postupy. Prvý zobrazuje obecný pohľad na priebeh výskumu a druhý proces analýzy dát, ktorá je obecným krokom priebehu výskumu. Podľa dostupných zdrojov sa jedná o jednu z prvých informatických prác, ktoré používajú zakotvenú teóriu. Dokonca aj na výskumnom poli je podľa Stola (2016) táto metóda nová a pre výskumníkov zaujímavá.

**Tabuľka 23 – Prínosy - metodické postupy (autor)**

Prínos	Odkaz
Vizuálny indikátor postupu výskumu znázorňujúci proces kvalitatívnej analýzy	Obrázok 9 – Indikátor postupu výskumu (autor)
Postup výskumníka znázorňujúci proces priebehu výskumu	Obrázok 6 – Postup výskumníka (autor)

Z uvedených prínosov vyplýva naplnenie všetkých stanovených cieľov na začiatku práce. Podarilo sa mi identifikovať silné a slabé oblasti v piatich projektoch zaoberajúcich sa agilným vývojom. Navrhol som zlepšenia procesov a tiež vytvoril model oblastí, ktorými agilný tím vie ovplyvniť vyspelosť testovacieho procesu. Hlavný cieľ teda považujem za naplnený, pretože som našiel spôsob, akým sa dajú zlepšiť procesy testovania.

### 6.3 Varianty rozšírenia práce

Prácu je možné prevziať a aplikovať získane poznatky o oblastiach ovplyvňujúcich vyspelosť agilného testovania do praxe a potvrdiť alebo vyvrátiť úspešnosť modelu. Kvalitatívne dáta môžu byť tiež analyzované znovu a pridané k ďalšiemu rozširujúcemu výskumu.

V tejto diplomovej práci boli skúmané tímy pôsobiace vo veľkých medzinárodných i národných spoločnostiach. Ďalší výskum by sa mohol zamerať na start-upy, malé spoločnosti a stredné spoločnosti. Výsledky tohto výskumu môžu byť porovnané s výstupmi tejto diplomovej práce čím sa rozšíria znalosti o vyspelosti agilného testovania.

V analýze kvalitatívnych dát boli identifikované kategórie, ktoré neboli predmetom výskumných otázok tejto diplomovej práce, ale môžu slúžiť ako podklad pre ďalšie práce zaoberajúce sa porovnaním aplikácie metodiky scrum, vnímaním kvality tímu alebo vnímaním QA (quality assurance) v rôznych tímoch.

Dôležitosť testerov v agilných tímoch je tiež vhodné rozšíriť o poznatky zo start-upov, malých a stredných spoločností a do výskumu tiež zahrnúť tímy, ktoré sú úspešné bez testerov ak nejaké existujú.



# Príloha A: Otázky na skúmanie vyspelosti agilného testovania

## Identifikačné otázky

- 1) Aká je Vaša rola v projektovom tíme?
- 2) Aké máte skúsenosti s prácou na vývoji softvéru?
- 3) Ako dlho pôsobíte v spoločnosti?
- 4) Ako dlho pôsobíte na projekte?
- 5) Ako dlho sa pohybujete v oblasti agilného vývoja softvéru?
- 6) Akú metodológiu vývoja softvéru používa Vaša spoločnosť?

## Exploratívne otázky

- 1) **Viete mi povedať príbeh Vášho testovacieho procesu od vzniku vášho tímu až do súčasnosti?**

Doplňujúce otázky pre 1 a 2

- a) Ktoré míľniky viedli k zlepšeniu vášho testovacieho procesu?
- b) Kto bol zodpovedný za aktivity?
- c) Čo a ako sa dialo?
- d) Aký bol dôvod testovania?
- e) Kde sa testovanie uskutočňovalo?
- f) Ako dlho trval proces?
- g) Výhody a nevýhody?
- h) Ako ste sa dozvedeli, že ste zlepšili tento proces?
- i) Ďalšie v závislosti na odpovedi

- 2) **Ako spokojný ste so súčasným stavom testovania?**

Doplňujúce otázky pre 3

- a) Prečo?
- b) Ďalšie v závislosti na odpovedi

- 3) **Aké oblasti pre zlepšenie testovacieho procesu vidíte?**

Doplňujúce otázky pre 4

- a) Čo konkrétne sa podľa Vás má urobiť pre zlepšenie procesu testovania v týchto oblastiach?
- b) Ďalšie v závislosti na odpovedi

**4) Akú istotu máte v rozhodnutiach pre release softvérových zmien?**

Doplňujúce otázky pre 5

- a) Ako to bolo v minulosti?
- b) Čo by podľa Vás pomohlo zvýšiť túto istotu?
- c) Ďalšie v závislosti na odpovedi

**5) Ako vnímate rolu testera v tíme?**

Doplňujúce otázky pre 6

- a) Prínosy?
- b) Prečo sú potrební?
- c) Čo by mali podľa Vás robiť?
- d) Ďalšie v závislosti na odpovedi

**6) Je ešte niečo s čím by ste sa chceli podeliť a myslíte si, že je pre výskum dôležité?**

## Príloha B: Popis problémových oblastí a podoblastí

Tabuľka 24 zobrazuje popis problémových oblastí, ktoré boli vytvorené spojením problémových podoblastí. Oblasti obsahujú spoločné problémy aspoň dvoch tímov a predstavujú vyššiu úroveň abstrakcie problémov.

Tabuľka 24 – Popis problémových oblastí skúmaných tímov (autor)

Problémová oblasť	Popis
<b>Automatizácia</b>	V tejto oblasti sú zahrnuté všetky aspekty týkajúce sa automatizácie testov od definovania aktivít, možnosti optimalizácie automatizácie až po určenie zodpovedností za automatizáciu v produktovom tíme. Má sa vylepšovať.
<b>Spolupráca a komunikácia</b>	Táto oblasť obsahuje aspekty komunikácie a spolupráce vo vnútri tímu, medzi členmi tímu, ale i tímovú spoluprácu a komunikáciu navonok, napríklad k zákazníkemu tímu. Má sa podporovať.
<b>Projektové obmedzenia</b>	Sem patria obmedzenia, ktoré majú negatívny vplyv na tím a vyspelosť testovania, ako napríklad pevne stanovené termíny. Sú spojené s projektovým plánom. Majú sa odstraňovať.
<b>Organizačné obmedzenia</b>	Tieto obmedzenia prináša organizačné prostredie v ktorom tím pôsobí. Môžu vyplývať z organizačnej štruktúry, kultúry a iných aspektov organizačného prostredia. Majú sa odstraňovať.
<b>Proces agilného vývoja</b>	Tu zahŕňam procesné aspekty, ktoré tím uskutočňuje. Skladá sa z dielčích procesov ako napríklad proces testovania, proces vývoja produktu, proces návrhu dizajnu a iné, ktoré spolu tvoria proces agilného vývoja. Patrí sem napríklad optimalizácia procesov, vykonávané aktivity až po zodpovednosti za aktivity. Má sa vylepšovať.
<b>Myslenie člena tímu</b>	Táto oblasť obsahuje aspekty myšlienkových pochodov jednotlivých členov tímu a ich zmeny. Patrí sem napríklad sústredenie člena tímu, spôsob akým premýšľa o svojej práci, o tíme a ďalších aspektoch. Má sa podporovať.
<b>Schopnosti člena tímu</b>	Oblasť zahŕňa nedostatok vedomostí, znalostí alebo skúsenosti člena tímu, ktorými by mal disponovať. Ide napríklad o nízke technické znalosti, rozdielnu participáciu členov vo výkone tímu a ďalšie. Majú sa podporovať.
<b>Zdieľanie znalostí</b>	Do tejto oblasti zaradujem problémy s predávaním znalostí iným členom tímu písomne, ústne alebo elektronicky. Môže to byť napríklad problém nedostatočnej dokumentácie, chýbajúceho človeka so skúsenosťami. Majú sa podporovať.

Tabuľka 25 zobrazuje popis problémových podoblastí, ktoré predstavujú konkrétne problémy skúmaných tímov. Niektoré problémy sú spoločné a niektoré individuálne, o čom pojednáva podkapitola 4.8.

**Tabuľka 25 – Popis problémových podoblastí skúmaných tímov (autor)**

<b>Problémová podoblasť</b>	<b>Popis</b>
<b>Pevne stanovené deadlines</b>	Nespokojnosť s pevne stanovenými projektovými termínmi.
<b>Pooly rolí</b>	Nespokojnosť s organizačným členením jednotlivých pracovných rolí.
<b>Distribuované tímy</b>	Nespokojnosť s rozmiestnením jednotlivých členov tímu.
<b>Regulatórne obmedzenia</b>	Obmedzenia vyplývajúce z požiadaviek vďaka ktorým je kladený väčší dôraz na dokumentáciu a definované procesy.
<b>Optimalizácia procesu testovania</b>	Nespokojnosť s nedostatočne optimalizovaným procesom testovania, ktorý je vhodné zlepšiť.
<b>Zavedenie indikátorov</b>	Nespokojnosť s nedostatočnými metrikami na sledovanie vývoja testovacieho procesu.
<b>Rozrobená práca</b>	Nespokojnosť s dlhým časom neuzatvorených úloh v product backlogu.
<b>Rozdelená zodpovednosť testerov</b>	Nespokojnosť so zdieľanou zodpovednosťou testerov za jednu úlohu.
<b>Zodpovednosť za automatizáciu</b>	Nespokojnosť s rozdelením tímovej zodpovednosti za automatizáciu.
<b>Optimalizácia automatizácie</b>	Nespokojnosť s nedostatočne optimalizovaným procesom automatizácie, ktorý je vhodné zlepšiť.
<b>Nerovnomerná participácia členov na tímových záležitostiach</b>	Nespokojnosť s rozdielnym zapojením jednotlivých členov tímu do tímovej výkonnosti.
<b>Nízke technické znalosti testerov</b>	Nespokojnosť s nedostatočnými technickými znalosťami testerov v tíme.
<b>Tímová spolupráca a komunikácia</b>	Nespokojnosť so spolupracou a komunikáciou vo vnútri produktového tímu.
<b>Zlá viditeľnosť automatizácie</b>	Nespokojnosť s viditeľnosťou prínosov automatizácie.
<b>Spolupráca a komunikácia s klientom</b>	Nespokojnosť so spolupracou a komunikáciou produktového tímu navonok ku klientom.

Problémová podoblasť	Popis
<b>Nedostatočná zameranosť členov na testovacie aktivity/kvalitu</b>	Problém nesústredenia členov tímu na kvalitu produktu.
<b>Neprijatie zmeny do praxe</b>	Problém neprijatia tímom odsúhlasených zmien do praxe.
<b>Dokumentácia</b>	Nespokojnosť s rozsahom projektovej dokumentácie.
<b>Zlepšenie marketingovej kampane</b>	Návrh na zlepšenie marketingovej kampane.
<b>Nízka kvalita zadania</b>	Nespokojnosť s kvalitou zadania, ktoré tím dostáva od zákazníckeho tímu.
<b>Tím bez testera</b>	Nespokojnosť členov tímu s chýbajúcou rolou testera.
<b>Lepší interakčný dizajn</b>	Nespokojnosť s aktuálnym dizajnom aplikácie.
<b>Nastavenie prostredí</b>	Nespokojnosť nastavením aktuálne používaných testovacích prostredí.
<b>Nepoužívané testovacie scenáre</b>	Problém s nepoužívanou vytvorenou testovacou dokumentáciou.
<b>Nenastavené pravidlá</b>	Nespokojnosť s nedostatočne nastavenými pravidlami predávok práce.
<b>Lepšia kontrola testovania</b>	Nespokojnosť s chýbajúcim človekom zodpovedným za kontrolu testovania.
<b>Presun zodpovednosti za testovaciu dokumentáciu</b>	Nespokojnosť s aktuálne pridelenou zodpovednosťou za prípravu testovacej dokumentácie.
<b>Lepšia organizácia informácií o úlohách</b>	Problém s neprehľadnými informáciami o úlohách užívateľských príbehov.
<b>Nevyhovujúci pracovný priestor</b>	Nespokojnosť s aktuálne prideleným pracovným priestorom v ktorom tím pôsobí.
<b>Zväčšenie tímu</b>	Nespokojnosť so súčasným počtom členov tímu.
<b>Mentoring</b>	Chýbajúci učiteľ, mentor, ktorý disponuje potrebnými skúsenosťami.
<b>Nedostatočná sústredenosť na prácu</b>	Nespokojnosť s poľavením v sústredení na svoju prácu. Dočasná zmena.
<b>Veľký počet integračných testov</b>	Nespokojnosť s rozsahom integračných automatizovaných testov.



## Príloha C: Formulár informovaného súhlasu

Byl/a jste zkontaktován/a k rozhovoru za účelem získání dat a informací pro kvalitativní výzkum k diplomové práci Bc. Martina Homoly. Výzkum probíhá v rámci magisterského studia na Fakultě informatiky a statistiky Vysoké školy ekonomické v Praze (obor: Informační systémy a technologie). Tématem jsou procesy testování v agilních projektech a výstupem bude diplomová práce, příp. volitelně doplněná konferenčním/časopiseckým příspěvkem v angličtině na některém mezinárodním akademickém fóru.

Diplomová práce se specificky zaměřuje na empirický výzkum týkající se testovacích aktivit (procesů a praktik) a posun v chápání těchto aktivit v agilních projektech oproti vodopádově řízeným projektům. Teoretická část bude shrnovat dosavadní literaturu na toto téma, empirická část bude vedena kvalitativními metodami zaměřenými na porozumění a perspektivu participantů v souladu s etickými zásadami kvalitativního výzkumu. Ambicí výzkumu je rovněž přinést souhrn praktických návrhů pro zlepšení testovacích aktivit v agilních projektech.

Vaše spolupráce na projektu je dobrovolná. Účast na projektu spočívá v nahrávaném rozhovoru, který potrvá zhruba 45 min. Vaše identita zůstane v plné anonymitě a Vaše odpovědi jsou důvěrné, určené pouze pro potřeby tohoto výzkumného projektu. Vaše názory mohou být v projektu použity dle Vaší individuální pozice (developer, tester, scrum master, ...), ale žádná individuální označení (jména osob, názvy firem) nebudou v projektu zmíněna. Rozhovor bude nahráván na digitální diktafon. Nahrávka a poznámky z rozhovoru budou přístupné pouze autorovi a vedoucímu diplomové práce, příp. oponentovi (pokud si ji vyžádá k nahlédnutí). Do jednoho roku od ukončení projektu budou zničeny. Na kteroukoliv otázku můžete odmítnout odpovědět a kdykoliv v průběhu rozhovoru můžete interview ukončit. Ukončení rozhovoru a vystoupení z projektu pro Vás nebude mít žádné negativní důsledky. Vaše účast na projektu pro Vás nenese žádná rizika.

V případě dotazů týkajících se tohoto projektu se můžete obrátit na jeho hlavního řešitele, Bc. Martina Homolu (tel. 773 081 417, [xhomm06@vse.cz](mailto:xhomm06@vse.cz)). V případě otázek týkajících se Vašich práv jako respondenta/ky v projektu se můžete obrátit na vedoucího této diplomové práce, Ing. et Ing. Michala Doležela ([michal.dolezel@vse.cz](mailto:michal.dolezel@vse.cz) – lze dohodnout i telefonickou konzultaci).

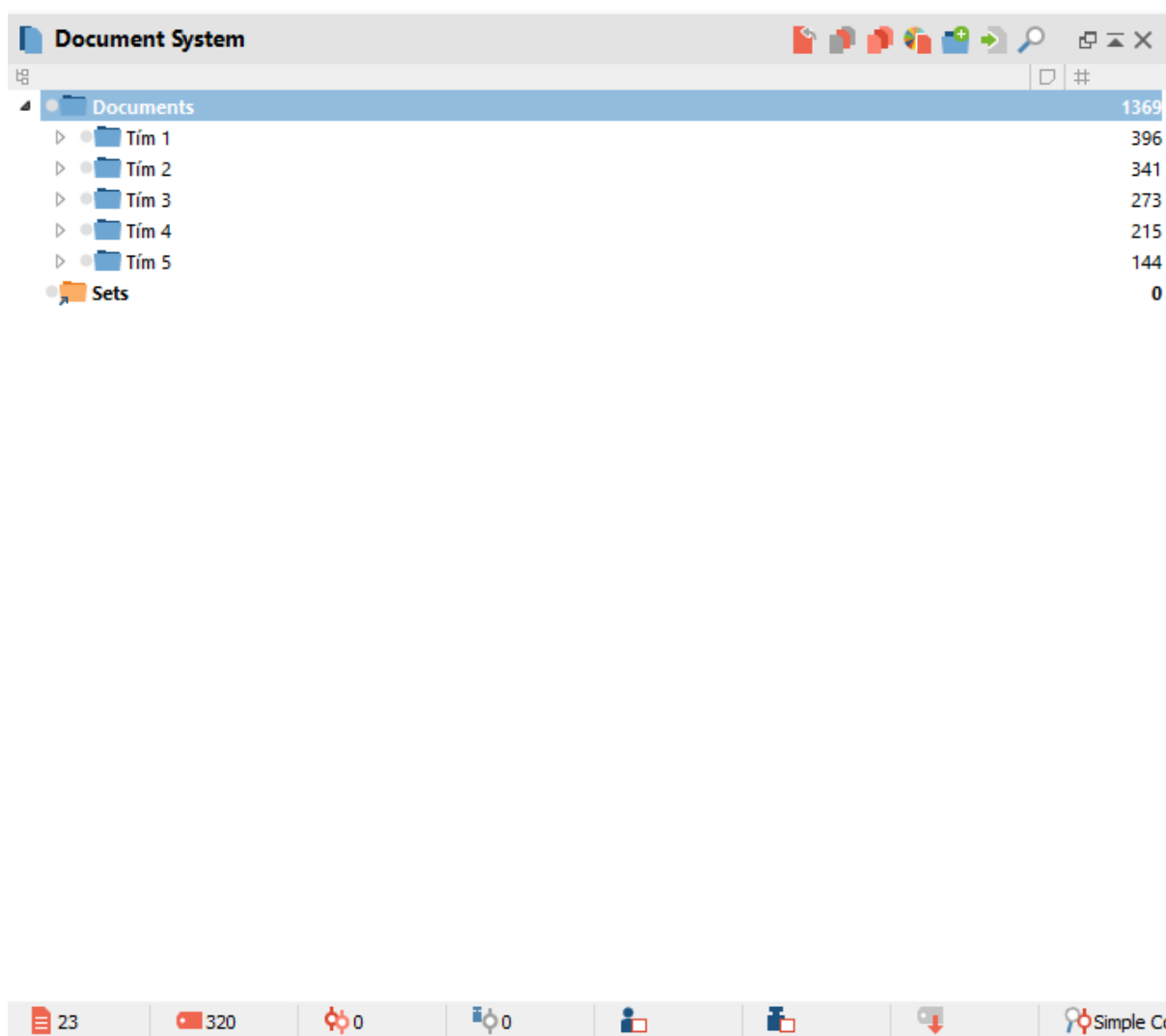
S těmito pravidly byli účastníci výzkumu seznámeni předem a uskutečněním rozhovoru s nimi vyjadřují svůj souhlas.

*Text souhlasu v souladu s etickými zásadami pro kvalitativní výzkum revidoval a výzkum odborně garantuje: Ing. et Ing. Michal Doležel, Katedra informačních technologií, FIS VŠE*

*Praha, leden 2017*

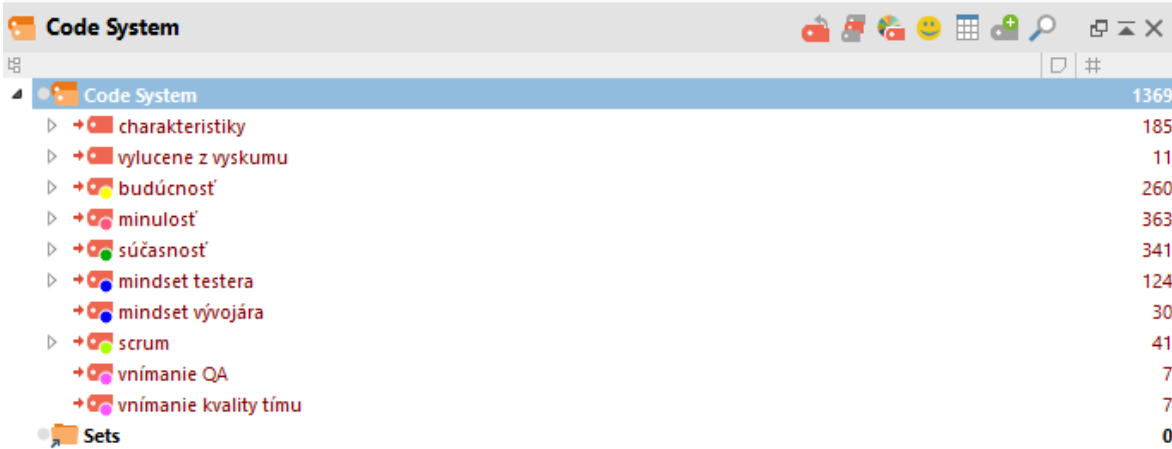
## Príloha D: Výstup z analýzy dát v MAXQDA Analytics Pro

Príloha obsahuje informácie o počte kódovaných segmentov výpovedí respondentov a prehľade kategórií vytvorených z týchto segmentov za použitia konceptu axiálneho kódovania. Otvorené kódovanie tu neuvádzam z dôvodu veľkého počtu pridelených kódov v tejto fázi analýzy dát. Okódovaných bolo 23 dokumentov v ktorých bolo použitých 320 kódov ako je zrejme z dolnej časti obrázku 22. Tento obrázok tiež znázorňuje počet použitých kódov podľa skúmaných tímov.



Obrázok 22 – Počet okódovaných segmentov podľa skúmaných tímov (autor)

Obrázok 23 zobrazuje kategórie prvej úrovne, ktoré obsahujú i okódované segmenty nesúvisiace s definovanými výskumnými otázkami môjho výskumu a môžu predstavovať zdroj informácií pre budúce výskumné aktivity v oblasti porovnania aplikácie metodiky scrum, kvality tímu a vnímania QA (quality assurance) v rôznych tímoch. Týmito témami sa zaoberajú kategórie scrum, vnímanie QA a vnímanie kvality tímu. Kategória charakteristiky obsahuje základné charakteristiky skúmaných tímov. Kategória vylúčené z výskumu obsahuje segmenty, ktoré som považoval za nerelevantné. Kategórie budúcnosť, súčasnosť a minulosť sa zaoberá hlavnými výskumnými otázkami ohľadom agilnej vyspelosti testovania a kategórie mindset testera a mindset vývojára sú zamerané na skúmanie prínosu testera v agilnom tíme.



Code System	Count
Code System	1369
↳ charakteristiky	185
↳ vylucene z vyskumu	11
↳ budúcnosť	260
↳ minulosť	363
↳ súčasnosť	341
↳ mindset testera	124
↳ mindset vývojára	30
↳ scrum	41
↳ vnímanie QA	7
↳ vnímanie kvality tímu	7
Sets	0

Filtering Query (OR combination of codes)

Obrázok 23 – Prehľad kategórií prvej úrovne (autor)

Kategória budúcnosť obsahuje problémové oblasti identifikované z výpovedí členov tímu ako kategórie druhej úrovne. Tieto oblasti vznikli zoskupením problémových podoblastí, ktoré na obrázku 24 zobrazujú kategórie tretej úrovne. Nižšie úrovne boli vytvorené v prvej fázi analýzy dát, otvorenom kódovaní a sú k nahliadnutiu u autora práce.



Obrázok 24 – Prehľad kategórie budúcnosť (autor)

Kategórie súčasnosť a minulosť obsahujú znaky vývoja identifikované z výpovedí členov tímu a predstavujú kategórie druhej úrovne. Z kategórií uvedenie potreby zmeny, aktívne hľadanie zlepšení a zavedenia zmeny do praxe bola v axiálnom kódovaní vytvorená kategória kontinuálne učenie, ktorá predstavuje spoločný znak vývoja testovacieho procesu. Kategórie spokojnosť s testovacím procesom, istota v kvalitu dodávaného softvéru a organizačné obmedzenia obsahujú informácie, ktoré nie sú spoločné pre skúmané tímy a sú uvedené pri popise vývoja testovacieho procesu v podkapitole 4.6. Nižšie úrovne boli vytvorené v prvej fázi analýzy dát, otvorenom kódovaní a sú k nahliadnutiu u autora práce.

- ▶ ➔ ➡ minulosť
  - ▶ ➔ ➡ uvedenie potreby zmeny
  - ▶ ➔ ➡ aktívne hľadanie zlepšení
  - ▶ ➔ ➡ zavedenie zmeny do praxe
  - ▶ ➔ ➡ spokojnosť s testovacím procesom
  - ▶ ➔ ➡ istota v kvalitu dodávaného softvéru
  - ▶ ➔ ➡ organizačné obmedzenia
- ▶ ➔ ➡ súčasnosť
  - ▶ ➔ ➡ uvedenie potreby zmeny
  - ▶ ➔ ➡ aktívne hľadanie zlepšení
  - ▶ ➔ ➡ zavedenie zmeny do praxe
  - ▶ ➔ ➡ spokojnosť s testovacím procesom
  - ▶ ➔ ➡ istota v kvalitu dodávaného softvéru

Obrázok 25 – Prehľad kategórií súčasnosť a minulosť (autor)

# Terminologický slovník

## Agilita

Schopnosť vytvárať a reagovať na zmenu za účelom zisku v turbulentnom biznis prostredí. Agilita je schopnosť vyvažovať flexibilitu a stabilitu. (Highsmith 2012)

## Akceptačné testovanie

Formálne testovanie rešpektujúce užívateľské potreby, požiadavky a biznis procesy vykonávané k určení, či systém uspokojuje akceptačné kritériá a umožňuje užívateľovi, zákazníkovi alebo iným autorizovaným entitám vymedziť či systém akceptovať alebo nie. (Van Veenendaal 2010)

## Aplikačné rozhranie

Rozhranie, ktoré umožňuje vývojárom a iným aplikáciám komunikáciu s aplikáciou, prístup k niektorým jej dátam a funkciám na týchto dátach (ako je vytvorenie, editácia alebo odstránenie). (Peacock 2016)

## Atribúty kvality testovania

Nefunkčné požiadavky slúžiace k ohodnoteniu činnosti systému. Voláme ich niekedy podľa anglickej prípony „ility“ atribúty. Obyčajne sú to architektonicky významné požiadavky, ktoré vyžadujú architektovu pozornosť. (Chen et al. 2013)

## Beta verzia aplikácie

Riadená distribúcia produktu na cieľový trh za účelom získania spätnej väzby z trhu, jej vyhodnotenia a integrácie získaných dát. Na získanie spätnej väzby využíva užívateľov v dobe pred releasom produktu. (Fine 2002)

## Build

Ide o konštrukciu softvéru do spustiteľnej podoby. Označuje buď proces konvertovania zdrojového kódu do samostatných softvérových artefaktov (tzv. kompiláciu) a alebo samotný výsledok tohto procesu. (Lee 1996)

## Capability maturity model integration (CMMI – DEV)

Je referenčný model procesov, ktoré pokrývajú celý životný cyklus softvérového produktu (od návrhu, cez vývoj a následnú údržbu). (Buchalcegová 2009)

## Distribuovaný tím (geograficky rozptýlený tím)

Skupina ľudí, ktorí spolupracujú naprieč časom, priestorom a cez hranice ako jednotlivých štátov, tak i jednotlivých organizácií prostredníctvom počítačových sietí s využitím informačných a komunikačných technológií. (Bedrnová et al. 2012)

**Dodávateľský tím/ Produktový tím**

Je technický tím, ktorý produkuje softvér požadovaný zákazníckym tímom. Patria sem všetci, ktorí sa podieľajú na dodaní softvéru zákazníkovi ako vývojári, tester, analytici a ďalší. (Crispin a Gregory 2008)

**Empirické poznanie**

Je poznanie alebo zdroj znalostí získaných pomocou zmyslov, hlavne pozorovaním a experimentovaním. (Pickett 2011)

**Exploratívne testovanie**

Štýl testovania softvéru, ktorý zdôrazňuje osobnú slobodu a zodpovednosť testera kontinuálne optimalizovať kvalitu svojej práce vzdelávaním sa v oblasti testovania, testovacím návrhu, vykonávaní testov a interpretácii výsledkov testovania ako podporujúce sa aktivity, ktoré bežia paralelne počas projektu. (Kaner 2008)

**Funkcia (feature/epic/theme)**

Kus funkcionality popísanej užívateľom a je položkou v product backlogu. Rozpadá sa na užívateľské príbehy. (Gregory a Crispin 2014)

**Iterácia**

Krátky vývojový cyklus, obecné trvajúci dva až štyri týždne, na konci ktorého môže byť potenciálne dodaný produkčne použiteľný kód. (Crispin a Gregory 2008)

**Jednotkové testy (unit testy)**

Verifikujú chovanie malej časti celého systému. Časť môže byť malá ako jeden objekt alebo metóda, ktorá je výsledkom jedného alebo viacerých návrhových rozhodnutí. (Crispin a Gregory 2008)

**Kontinuálna integrácia**

Umožňuje tímu často kontrolovať zmeny v kóde. Každá kontrola spustí automatický proces tvorby buildu v ktorom automatická regresia testov poskytuje rýchlu spätnú väzbu na overenie, či zmeny kódu spôsobili regresné chyby alebo nie. Tieto buildy produkujú artefakty, ktoré môžu byť manuálne alebo automaticky nasadené na rôzne prostredia. (Crispin a Gregory 2008)

**Metóda**

Postup alebo návod, ako získať správne poznatky, prostriedok poznania. (Fajkus 2005)

**Metodika**

Metodika predstavuje v obecnom zmysle súhrn metód a postupov pre realizáciu určitej úlohy. Metodika budovania IS/ICT definuje princípy, procesy, praktiky, role, techniky, nástroje a produkty používané pri vývoji, údržbe a prevádzke informačného systému a to ako z pohľadu. (Buchalcevová 2005)

### **Minimum viable product**

Má práve potrebné množstvo funkcií, ktoré dovoľujú nasadiť produkt malému množstvu potenciálnych zákazníkov. Reprezentuje najmenej zmien, ktoré môžu byť doručené zákazníkovi a prinášajú biznis hodnotu. (Gregory a Crispin 2014)

### **Návratnosť investícií**

Miera efektívnosti investície. V testovaní je to prospech získaný z testovacej aktivity ako automatizovaný test, vážené oproti nákladom na tvorbu a údržbu testu alebo aktivity. (Crispin a Gregory 2008)

### **Pracovná rola**

Očakávaný štandard jednania zamestnanca, ktorý vyplýva z pracovnej pozície, ktorú zastáva. Pracovná rola je určitý „ideál“ odvodený z príslušnej pozície bez ohľadu na konkrétneho pracovníka, ktorý ju bude vykonávať. (Bedrnová et al. 2012)

### **Quality assurance tím (QA tím)**

Vo vývoji softvéru pojem QA tím je často používaný pre tím, ktorý vykonáva testovanie softvéru. Testovacie tímy poskytujú zainteresovaným stranám informácie spojené s kvalitou produktu. Uskutočňujú aktivity aby sa naučili ako by sa mal systém správať a overujú či sa správa ako to zákazník požaduje. V agilnom vývoji, tieto aktivity sú plne integrované s vývojárskymi aktivitami. Testeria sú často súčasťou vývojárskeho tímu spolu s ostatnými zainteresovanými do vývoja softvéru. (Crispin a Gregory 2008)

### **Refaktoring**

Zmena kódu, bez zmeny jeho funkcionality tak, aby bol lepšie udržiavateľný, ľahšie čitateľný a testovateľný, alebo ľahšie rozšíriteľný. (Crispin a Gregory 2008)

### **Regresné testy**

Verifikuje, že chovanie testovaného systému sa nezmenilo. Sú väčšinou písané ako jednotkové testy, ktoré riadia vývoj alebo akceptačné testy definujúce požadované chovanie systému. (Crispin a Gregory 2008)

### **Release**

Môže pozostávať z nového produktu alebo nových funkcií pre existujúci produkt. V agilnom vývoji funkcia môže pozostávať z jedného alebo viacerých dokončených užívateľských príbehov. (Gregory a Crispin 2014)

### **Sprint backlog**

Je Scrum pojem označujúci sprioritizovaný zoznam celej funkcionality požadovanej od produktu. Backlog časom rastie, keď sa zákaznícky tím rozhodne, že bude potrebovať nové funkcie. (Crispin a Gregory 2008)



## **System na správu kódu / Manažment verzii**

Dovoľuje viacerým členom tímu pracovať nad rovnakým základom kódu, dokumentov, alebo iných typoch súborov bez narušenia zmien ostatných členov tímu. Medzi bežné funkcie patrí spájanie zmien do rovnakého súboru, návrat vykonaných zmien, porovnávanie verzii, zobrazenie histórie zmien a získanie špecifických verzii. (Gregory a Crispin 2014)

## **Technický dlh**

V prípade, že tím produkuje softvér bez použitia najlepších praktík akými sú vývoj riadený testami, kontinuálna integrácia a refaktoring, môže sa objaviť technický dlh. Zvyčajne technický dlh narastá a spomaľuje tímovú rýchlosť. Každú iteráciu produkuje stále menej hodnoty pre biznis, pretože kódu chýba záchranná sieť automatizovaných regresných testov alebo sa automatizácia stala zložitou na porozumenie a údržbu. (Crispin a Gregory 2008)

## **Úloha (task)**

Kúsky práce potrebné na dokončenie užívateľského príbehu. Môže byť akciou potrebnou k implementácii malého kusu užívateľského príbehu, alebo to môže byť vytvorenie časti infraštruktúry, alebo testovanie, ktoré zahŕňa viac ako jeden užívateľský príbeh. Obecne reprezentuje prácu na maximálne jeden deň. (Crispin a Gregory 2008)

## **Užívateľský príbeh (user story)**

Krátky popis funkcionality, povedaný z perspektívy užívateľa, ktorý je hodnotný pre užívateľa alebo zákaznícky tím. Typicky obsahuje jednoriadkový popis funkcie. (Crispin a Gregory 2008)

## **Vývoj riadený akceptačnými testami (acceptance test driven development)**

Špecifikujú sa spustiteľné akceptačné testy za použitia príkladov zistených z konverzácií so zákazníkmi. Programátorov tieto testy navádzajú na vývoj funkcie. (Gregory a Crispin 2014)

## **Vývoj riadený testami (test driven development)**

Programátor píše a automatizuje malé množstvo jednotkových testov predtým, než napíše malý kus kódu, ktorý umožní ich úspešný prechod. Produkčný kód sa vytvára postupne test za testom. (Crispin a Gregory 2008)

## **Zákazník (zákaznícky tím)**

Identifikuje a prioritizuje potrebné funkcie požadované biznisom. Zahŕňajú všetky zainteresované strany mimo produktový tím ako biznis expertov i koncových užívateľov. (Crispin a Gregory 2008)

## Bibliografia

ADALI, Onat Ege, Özden ÖZCAN-TOP a Onur DEMİRÖRS, 2016. *Evaluation of Agility Assessment Tools: A Multiple Case Study* [online]. s. 135 [cit. 2017-04-24]. DOI: 10.1007/978-3-319-38980-6\_11. Dostupné z: [http://link.springer.com/10.1007/978-3-319-38980-6\\_11](http://link.springer.com/10.1007/978-3-319-38980-6_11)

ADZIC, Gojko, 2009. *Bridging the communication Gap: specification by example and agile acceptance testing*. London: Neuri Limited. ISBN 978-0-9556836-1-9.

AMBLER, Scott, 2012. *Results from the November 2012 Agile Testing Survey* [online] [vid. 2017-04-09]. Dostupné z: <http://www.ambysoft.com/surveys/agileTesting201211.html>

BECK, et al., 2001. *Manifesto for Agile Software Development* [online] [vid. 2017-04-15]. Dostupné z: <http://agilemanifesto.org/>

BEDRNOVÁ, Eva, Ivan NOVÝ a Eva JAROŠOVÁ, 2012. *Manažerská psychologie a sociologie*. Praha: Management Press, ISBN 978-80-7261-239-0.

BIRO, M., et al., 2012. Agile Maturity Model, Go back to the Start of the Cycle. In: *Industrial Proceedings of the 2012 EuroSPI Conference*. p. 5.9-5.30.

BLAŽKOVÁ, Tereza, 2013. *Testování softwaru v agilních metodikách* [online]. Praha: Vysoká škola ekonomická v Praze. Dostupné z: <http://theses.cz/id/kz63w8/>

BUCHALCEVOVÁ, Alena, 2005. *Metodiky vývoje a údržby informačních systémů: kategorizace, agilní metodiky, vzory pro návrh metodiky*. Praha: Grada. ISBN 978-80-247-1075-4.

BUCHALCEVOVÁ, Alena, 2009. *Metodiky budování informačních systémů*. Praha: Oeconomica: ISBN 978-80-245-1540-3.

BUREŠ, Miroslav, Miroslav RENDA, Michal DOLEŽEL a kolektiv A, 2016. *Efektivní testování softwaru: Klíčové otázky pro efektivitu testovacího procesu*. Praha: Grada Publishing a.s. ISBN 978-80-271-9389-9.

CARROLL, N.R., 2009. *The Communication Problem Solver: Simple Tools and Techniques for Busy Managers* [online]: AMACOM. ISBN 978-0-8144-1309-8.

COOKE, J.L., 2014. *Agile Productivity Unleashed: Proven approaches for achieving real productivity gains in any organization* [online]: IT Governance Limited. ISBN 978-1-84928-564-3.

COTTMEYER, Mike, 2015. *What Is An Agile Team and How Do You Form Them?* - [online] [vid. 2017-04-09]. Dostupné z: <https://www.leadingagile.com/2015/02/what-is-an-agile-team-and-how-do-you-form-them/>

- CRESWELL, John W, c1998. *Qualitative inquiry and research design: choosing among five traditions*. Thousand Oaks, Calif.: Sage Publications. ISBN 978-0-7619-0144-0.
- CRISPIN, Lisa a Janet GREGORY, 2008. *Agile Testing: A Practical Guide for Testers and Agile Teams*. Pearson Education. ISBN 978-0-321-61693-7.
- DEEKONDA, Rahul a Prithvi Raj SIRIGUDI, 2016. *Assessment of Agile Maturity Models : A Survey* [online] [vid. 2017-04-09]. Dostupné z: <http://www.diva-portal.org/smash/record.jsf?pid=diva2:1010343>
- DOŠEK, Tomáš, 2012. *Modely zlepšování procesů testování softwaru a zajištění kvality* [online] Praha. Diplomová práce. Vysoká škola ekonomická v Praze. Dostupné z: <http://theses.cz/id/dlqnoc/>
- DOLEŽEL, Michal, 2017. Images of Enterprise Test Organizations: Factory, Center of Excellence, or Community?. In: *International Conference on Software Quality*. Springer, Cham. p. 105-116.
- ELSSAMADISY, Amr, 2007. *Does the Agile Community Need a Maturity Model?* [online] [vid. 2017-04-15]. Dostupné z: [https://www.infoq.com/news/2007/10/agile\\_maturity\\_model](https://www.infoq.com/news/2007/10/agile_maturity_model)
- FAJKUS, Břetislav. *Filosofie a metodologie vědy: vývoj, současnost a perspektivy*. Praha: Academia, 2005. ISBN 978-80-200-1304-0.
- FINE, M.R., 2002. *Beta Testing for Better Software*. Wiley. ISBN 978-0-471-42786-5.
- FONTANA, Rafaela Mantovani, Isabela Mantovani FONTANA, Paula Andrea da Rosa GARBUIO, Sheila REINEHR a Andreia MALUCELLI, 2014b. Processes versus people: How should agile software development maturity be defined? *The Journal of Systems and Software; New York* [online]. 11., 97 [vid. 2017-04-14]. ISSN 01641212. Dostupné z: <http://search.proquest.com.zdroje.vse.cz/docview/1564770857/6344AF61134B47A8PQ/1>
- GIUDICE, Diego Lo, 2013. *Are You Really Ready To Test Agile?* [online] [vid. 2017-04-09]. Dostupné z: [http://blogs.forrester.com/diego\\_lo\\_giudice/13-02-19-are\\_you\\_really\\_ready\\_to\\_test\\_agile](http://blogs.forrester.com/diego_lo_giudice/13-02-19-are_you_really_ready_to_test_agile)
- GMBH, VERBI, 2017. *MAXQDA Analytics Pro - Qualitative and statistical analysis features in one* [online] [vid. 2017-04-15]. Dostupné z: <http://www.maxqda.com/products/maxqda-analytics-pro>
- GRAHAM, D. a M. FEWSTER, 2012. *Experiences of Test Automation: Case Studies of Software Test Automation*. Pearson Education. ISBN 978-0-13-277655-4.
- GREGORY, Janet a Lisa CRISPIN, 2014. *More Agile Testing: Learning Journeys for the Whole Team*. Addison-Wesley Professional. ISBN 978-0-13-374956-4.

- GUEST, David, 1991. The hunt is on for the Renaissance Man of computing. *The Independent (London)*. 17.
- HENDL, Jan, 2016. *Kvalitativní výzkum: základní teorie, metody a aplikace*. Čtvrté, přepracované a rozšířené vydání. Praha: Portál. ISBN 978-80-262-0982-9.
- HIGHSMITH, 2012. *What is Agility? | Jim Highsmith* [online]. [vid. 2017-04-23]. Dostupné z: <http://jimhighsmith.com/what-is-agility/>
- CHEN, Lianping, Muhammad Ali BABAR a Bashar NUSEIBEH, 2013. Characterizing architecturally significant requirements. *IEEE software*. 30(2), 38–45. ISSN 0740-7459.
- CHODURA, Ondřej, 2012. *Lean software development v testovacím týmu* [online]. Praha. Diplomová práce. Vysoká škola ekonomická v Praze. Dostupné z: <http://theses.cz/id/vc3h4i/>
- KANER, Cem, 2008. *A tutorial in exploratory testing*. Tutorial presented at QUEST2008. [online]. [vid. 2017-04-04]. Dostupné z: <http://www.kaner.com/pdfs/QAIEExploring.pdf>.
- KRÁLOVÁ, Iveta, 2012. *Metodika testování podle mezinárodních praktik a standardů* [online]. Praha. Diplomová práce. Vysoká škola ekonomická v Praze. Dostupné z: <http://theses.cz/id/fw24ne/>
- KUČERA, Martin, 2007. *Zvyšování zralosti testovacích procesů v IT společnosti* [online]. Praha. Diplomová práce. Vysoká škola ekonomická v Praze. Dostupné z: <http://theses.cz/id/kpyj9e/>
- KULHÁNKOVÁ, Barbora, 2015. *Behaviour Driven Development a Scrum v korporátním prostředí* [online]. Praha. [vid. 2017-04-04]. Diplomová práce. Vysoká škola ekonomická v Praze. Dostupné z: [http://theses.cz/id/evyqhz/53994\\_xkulb00.pdf?info=1;isshlret=Agile%3B;zpet=%2Fvyhl edavani%2F%3Fsearch%3D%22agile%20testing%22%26start%3D2](http://theses.cz/id/evyqhz/53994_xkulb00.pdf?info=1;isshlret=Agile%3B;zpet=%2Fvyhl edavani%2F%3Fsearch%3D%22agile%20testing%22%26start%3D2)
- LEE, Kevin A., 1996. *The Buildmeister's Guide: Achieving Agile Software Delivery*. 1st edition. Lulu: Lulu.com. ISBN 978-1-84728-373-3.
- LINZ, Tilo, 2014. *Testing in Scrum: A Guide for Software Quality Assurance in the Agile World*. Rocky Nook, Inc. ISBN 978-1-4920-0153-9.
- MRÁZEK, Štěpán, 2014. *Analýza a optimalizace procesů testování SW na vybraném IT oddělení mezinárodní finanční instituce* [online]. Praha. Diplomová práce. Vysoká škola ekonomická v Praze. Dostupné z: <http://theses.cz/id/uctfa8/>
- PEACOCK, M., 2016. *Programujeme vlastní sociální síť*. ISBN 978-80-251-4090-1.
- PICKETT, Joseph P, 2011. Empirical. *The American Heritage Dictionary of the English Language*.
- ROJO, Robert, 2014. *Testování softwaru v agilních projektech* [online] [online]. B.m.: Vysoká škola ekonomická v Praze, Praha. Dostupné z: <http://theses.cz/id/bpdl0i/>

- ROTHMAN, Johanna, 2011. Where is agile going with culture. In: [online]. Business. Vancouver. [vid. 2017-04-24]. Dostupné z: <https://www.slideshare.net/johannarothman/where-is-agile-going-withculture>
- SHIRLY, Ronen, 2011. *Shirly Ronen - A practical view on Agile Testing Maturity Levels* [online] [vid. 2017-04-14]. Dostupné z: <https://www.slideshare.net/AgileSparks/atmm-practical-view>
- SCHWEIGERT, Tomas, Mohsen EKSSIR-MONFARED a Magda OFNER, 2013. *An Agile Management Process Group for TestSPICE®* [online]. s. 228 [cit. 2017-04-24]. DOI: 10.1007/978-3-642-39179-8\_20. Dostupné z: [http://link.springer.com/10.1007/978-3-642-39179-8\\_20](http://link.springer.com/10.1007/978-3-642-39179-8_20)
- SCHWEIGERT, Tomas, Detlef VOHWINKEL, Monique BLASCHKE a Mohsen EKSSIR-MONFARED, 2013. *TestSPICE and Agile Testing – Synergy or Confusion* [online]. s. 154 [cit. 2017-04-24]. DOI: 10.1007/978-3-642-38833-0\_14. Dostupné z: [http://link.springer.com/10.1007/978-3-642-38833-0\\_14](http://link.springer.com/10.1007/978-3-642-38833-0_14)
- SCHWEIGERT, Tomas, Detlef VOHWINKEL, Morten KORSAA, Risto NEVALAINEN a Miklos BIRO, 2013. *Agile Maturity Model: A Synopsis as a First Step to Synthesis* [online]. s. 214 [cit. 2017-04-24]. DOI: 10.1007/978-3-642-39179-8\_19. Dostupné z: [http://link.springer.com/10.1007/978-3-642-39179-8\\_19](http://link.springer.com/10.1007/978-3-642-39179-8_19)
- SOFTWARE, N. C. H., 2017. *Express Scribe Free Transcription Software for Typists* [online] [vid. 2017-04-15]. Dostupné z: <http://www.nch.com.au/scribe/>
- STOL, Klaas-Jan, Paul RALPH a Brian FITZGERALD, 2016. Grounded theory in software engineering research. In: *Proceedings of the 38th International Conference on Software Engineering - ICSE '16* [online]. New York, New York, USA: ACM Press, s. 120-131 [cit. 2017-04-24]. DOI: 10.1145/2884781.2884833. ISBN 9781450339001. Dostupné z: <http://dl.acm.org/citation.cfm?doid=2884781.2884833>
- TECHOPEDIA INC., 2017. What is Application Log? - Definition from Techopedia. *Techopedia.com* [online] [vid. 2017-04-23]. Dostupné z: <https://www.techopedia.com/definition/1819/application-log>
- TUCKMAN, B. W. a M. A. C. JENSEN. 1977, Stages of Small-Group Development Revisited. *Group & Organization Management* [online]. 2(4), 419-427 [cit. 2017-04-24]. DOI: 10.1177/105960117700200404. ISSN 1059-6011. Dostupné z: <http://gom.sagepub.com/cgi/doi/10.1177/105960117700200404>
- VACHALEC, Vladan, 2013. *Testování a kvalita softwaru v metodikách vývoje softwaru* [online]. Praha. Diplomová práce. Vysoká škola ekonomická v Praze. Dostupné z: <http://theses.cz/id/w4j4zw/>
- VAN VEENENDAAL, Erik, 2010. Standard glossary of terms used in Software Testing. *International Software Testing Qualifications Board*. 1–51.
- WELLS, Don, 2009. *Extreme Programming: A Gentle Introduction*. [online] [vid. 2017-04-15]. Dostupné z: <http://www.extremeprogramming.org/>

---

ZHÁNĚL, Michal, 2014. *Analýza a optimalizace procesu vedení vývoje softwaru metodou SCRUM* [online]. Brno. Diplomová práce. Masarykova univerzita, Ekonomicko-správní fakulta. Dostupné z: [http://is.muni.cz/th/251450/esf\\_m/](http://is.muni.cz/th/251450/esf_m/)