

Vysoká škola ekonomická v Praze

Fakulta informatiky a statistiky

Katedra informačních technologií

Studijní program: Podniková informatika

Rozšíření metodiky MMSP v oblasti analýzy a návrhu testování

DIPLOMOVÁ PRÁCE

Student : Bc. Roman Novotný

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

Oponent : Mgr. Zdeněk Grössl

2015

Prohlášení

Prohlašuji, že jsem diplomovou práci na téma „Rozšíření metodiky MMSP v oblasti analýzy a návrhu testování“ zpracoval samostatně pod vedením vedoucího diplomové práce. Dále prohlašuji, že jsem uvedl všechny použité prameny a literaturu, ze kterých jsem čerpal.

V Praze dne 8. prosince 2015

.....

Bc. Roman Novotný

Poděkování

Rád bych na tomto místě srdečně poděkoval vedoucí své diplomové práce doc. Ing. Aleně Buchalcekové, Ph.D. a to nejen za podmětne připomínky, rady a čas, který mi věnovala, ale i za zprostředkování přístupu k české lokalizaci řady norem ČSN ISO/IEC/IEEE 29119.

Abstrakt

Diplomová práce „*Rozšíření metodiky MMSP v oblasti analýzy a návrhu testování*“ charakterizuje disciplínu Analýza a návrh testování a provádí její zapracování do Metodiky pro malé softwarové projekty (MMSP). Teoretická část vymezuje pojem disciplíny Analýza a návrh testování a charakterizuje její úlohu v kontextu testování softwaru a použití při vývoji softwarových produktů. Dále analyzuje, jak se k analytickým činnostem v oblasti testování staví obecně známé modely a metodiky vývoje softwaru, certifikace, standardy a normy v oblasti testování včetně rozšiřované metodiky MMSP.

Praktická část popisuje rozšíření metodiky MMSP o disciplínu Analýza a návrh testování, která umožňuje využít tuto disciplínu i na malých projektech vyvíjející nekritická IS/ICT řešení, což doposud nebylo možné. Disciplína analýza a návrh testování rozšiřuje metodiku MMSP o značné množství analytických činností v oblasti testování. Rozšíření metodiky MMSP zahrnuje zavedení nové role Analytika testování, úpravu stávajících rolí, zavedení nových pracovních produktů a popis doporučených technik testování a návrhu testování.

Klíčová slova

Analýza a návrh testování, Analytik testování, analytické činnosti v oblasti testování, Metodika pro malé softwarové projekty (MMSP), Metodika vývoje softwaru

Abstract

The thesis "*Extending the methodology MMSP in test analysis and design*" characterizes the discipline of test analysis and design carried out by its inclusion in the Methodology for small software projects (MMSP). The theoretical part defines the concept of discipline Test analysis and design and characterizes its role in the context of software testing and use in developing software products. It also analyzes attitude of commonly known models and methodologies of software development, certification, standards and norms in testing methodology including MMSP to the analytical activities in the field of testing.

The practical part describes the methodology MMSP extension by discipline Test analysis and design, which allows the use of this discipline even on small projects developing uncritical IS/ICT solutions, which previously was not possible. Discipline Test analysis and design expands MMSP at considerable amount of analytical activities in the area of testing. Extending the methodology MMSP includes the introduction of new role Test analyst, modification of existing roles, the introduction of new products and descriptions of the recommended testing techniques and test design.

Keywords

Test analysis and design, Test analyst, Analytical activities in testing, Methodology for small software projects (MMSP), Methodology for Software Development

Obsah

1	Úvod	1
1.1	Cíle práce	2
1.2	Přínos práce	2
1.3	Předpoklady a omezení práce	3
2	Rešerše zdrojů zabývajících se analýzou a návrhem testování	4
2.1	Odborná tištěná literatura	4
2.2	Certifikace a mezinárodní normy.....	5
2.3	Elektronické informační zdroje	6
2.4	Akademické práce	7
3	Charakteristika disciplíny Analýza a návrh testování a vymezení základních pojmů	9
3.1	Vymezení pojmu disciplína Analýza a návrh testování.....	9
3.2	Přínos disciplíny Analýza a návrh testování	10
3.3	Úvod do problematiky testování softwaru	11
3.4	Shrnutí	14
4	Modely a metodiky vývoje softwaru dle postoje k testování	15
4.1	Obecně o modelech a metodikách vývoje softwaru.....	15
4.1.1	Dělení metodik vývoje softwaru	15
4.2	Modely životního cyklu softwaru a jejich vztah k testování.....	17
4.3	Přístup rigorózních a agilních metodik k testování.....	19
4.4	Rozšiřovaná metodika MMSP	23
4.5	Shrnutí	26
5	Analýza a návrh testování z pohledu certifikací ISTQB	27
5.1	Obecně o certifikacích ISTQB.....	27
5.2	Role Test Analytik dle ISTQB.....	29
5.3	Role Technický test analytik dle ISTQB	34
5.4	Shrnutí	35
6	Analýza a návrh testování z pohledu normy ČSN ISO/IEC/IEEE 29119	37
6.1	Norma ČSN ISO/IEC/IEEE 29119.....	37
6.2	Definice pojmů v oblasti analýzy a návrhu testování podle ČSN ISO/IEC/IEEE 29119-1	38

6.2.1	Definice pojmů rolí	38
6.2.2	Definice pojmů s ohledem na disciplíny metodiky MMSP.....	39
6.2.3	Definice pojmů technik	41
6.2.4	Definice pojmů pracovních produktů.....	42
6.3	Analýza a návrh testování v testovacím procesu podle normy ČSN ISO/IEC/IEEE 29119-2..	45
6.3.1	Úloha analytických činností v procesu testování organizace	46
6.3.2	Proces plánování testování	47
6.4	Shrnutí	48
7	Návrh na rozšíření metodiky MMSP	50
7.1	Úprava rolí metodiky MMSP	51
7.1.1	Manažer testování	53
7.1.2	Analytik testování	54
7.1.3	Tester	56
7.1.4	Architekt	57
7.1.5	Projektový manažer	57
7.1.6	Analytik	57
7.1.7	Vývojář	58
7.1.8	Návrhář uživatelského rozhraní (Návrhář UI)	58
7.1.9	Grafik.....	58
7.1.10	Nedefinovaná role.....	58
7.1.11	Zainteresaná strana.....	58
7.1.12	Rozvržení testovacího týmu.....	59
7.2	Zavedení disciplíny Analýza a návrh testování do metodiky MMSP	59
7.2.1	Úloha Analýza testovatelnosti.....	60
7.2.2	Úloha Analýza rizik testování	61
7.2.3	Úloha Analýza Báze testování	62
7.2.4	Úloha Analýza a identifikace testovacích podmínek	63
7.2.5	Úloha Definování Strategie testování	63
7.2.6	Úloha Plánování testování	64
7.2.7	Úloha Návrh testování	66
7.2.8	Úloha Příprava a tvorba testů	67
7.2.9	Úloha Provedení testování	68

7.2.10	Úloha Vyhodnocení a ukončení testování.....	68
7.3	Vybrané techniky návrhu testování používané při analýze a návrhu testování	69
7.3.1	Testování založené na rizicích	70
7.3.2	Testování tříd ekvivalence	70
7.3.3	Analýza okrajových hodnot	71
7.3.4	Kombinatorická analýza	72
7.4	Pracovní produkty.....	73
7.4.1	Dokument Analýza testovatelnosti	74
7.4.2	Dokument Analýza rizik testování	74
7.4.3	Báze testování	75
7.4.4	Testovací podmínky	76
7.4.5	Strategie testování	76
7.4.6	Plán testování	77
7.4.7	Návrh testování	78
7.4.8	Testovací scénář.....	78
7.4.9	Testovací sada.....	78
7.4.10	Testovací data	79
7.4.11	Záznam výsledků testů.....	79
7.4.12	Report hodnocení kvality	79
7.5	Dopad rozšíření metodiky MMSP na životní cyklus	80
7.5.1	Fáze Zahájení	80
7.5.2	Fáze Rozpracování	81
7.5.3	Fáze Konstrukce.....	81
7.5.4	Fáze Zavedení	82
7.6	Shrnutí	82
8	Závěr.....	84
	Seznam literatury a zdrojů.....	86
	Seznam obrázků	90
	Seznam tabulek	91
	Terminologický slovník	92

1 Úvod

V dnešní informační a komunikační společnosti, nebo dokonce by se dalo říci i znalostní společnosti, je celá naše kultura založena na využívání informačních a komunikačních technologií (IS/ICT). IS/ICT technologie se staly běžnou součástí našeho života a nedovedeme si bez nich běžný den už ani představit. Jedná se o různé druhy informačních systémů (IS), programů a aplikací, které se vyvíjejí na míru pro daný přístroj nebo technologii formou projektu. Projekty vývoje softwaru probíhají z pravidla za použití metodiky vývoje softwaru a skládají se z mnoha specializovaných činností, mezi něž patří v neposlední řadě i testování výstupního produktu.

Tato diplomová práce na téma „Rozšíření metodiky MMSP v oblasti analýzy a návrhu testování“ přibližuje oblast poměrně novou oblast analytických činností v testování. Zabývá se dnes již plně samostatnou disciplínou v rámci testovacího procesu, tj. Analýzou a návrhem testování. Tato disciplína se oddělila (obdobně jako problematika např.: automatizace testů) díky narůstající specializaci jednotlivých činností na projektech zabývajících se vývojem informačních systémů.

Disciplína Analýza a návrh testování v sobě zahrnuje převážně analytické činnosti, které se týkají potřeb testování. Řadím mezi ně především poznávání funkčních požadavků, analýzu architektonického modelu, sběr podkladů, návrh a tvorbu testovacích scénářů za použití různorodých analytických technik. Hlavním výstupem jsou kromě strategie testování a plánu testování i testovací scénáře, které poté testeré vykonávají podle potřeby.

Tato disciplína je v současnosti na vzestupu, stejně jako celá oblast testování softwaru. Je na ni kladen důraz ale pouze na velkých projektech, kde jsou dostatečné kapacity na vytvoření plnohodnotného testovacího týmu. Soudím tak podle toho, že disciplína Analýzy a návrhu testování je zahrnuta pouze v metodikách pro velké projekty, jako je například RUP (Kruchten, 2000). Metodiky jako například MMSP (Buchalcegová a Stanovská, 2013) vhodné pro použití na menších projektech tuto disciplínu zatím nezahrnují. Zahrnutí této disciplíny právě do metodik pro menší projekty v sobě v současnosti skrývá velký potenciál.

Toto téma jsem si vybral proto, že se jedná o zajímavou dynamicky rozvíjející se oblast v rámci vývoje informačních systémů, tato oblast úzce souvisí se studovaným

oborem Podniková informatika, zároveň se jedná o oblast, jejíž význam v dnešní moderní znalostní společnosti stále sílí a také proto, že již přes dva roky pracuji na pozici test analytika na různých projektech vývoje informačních systémů pro oblast bankovníctví a pojišťovnictví.

1.1 Cíle práce

Hlavním cílem diplomové práce je charakterizovat disciplínu Analýza a návrh testování a provést její zpracování do metodiky MMSP tak, aby bylo možné tuto disciplínu reálně použít i na menších projektech vývoje softwaru.

Pro splnění hlavního cíle práce jsem definoval dílčí cíle. Prvním dílčím cílem je podrobná charakteristika a vymezení disciplíny Analýza a návrh testování. Dalším dílčím cílem je přiblížit, jak se k problematice testování a jeho analytických činností staví obecně známé modely a metodiky vývoje softwaru. Třetím dílčím cílem je analyzovat, jak se k problematice analytických činností v oblasti testování staví certifikace, standardy a normy v oblasti testování. Posledním dílčím cílem je vypracovat vlastní návrh na rozšíření vybrané metodiky o disciplínu Analýza a návrh testování.

1.2 Přínos práce

Hlavním výstupem diplomové práce je rozšíření stávající metodiky MMSP o relativně novou disciplínu Analýza a návrh testování. Tuto vysoce specializovanou disciplínu v oblasti vývoje informačních systémů dosud popisují metodiky, které jsou určeny především pro velké komerční projekty. Metodika MMSP vznikla lokalizací metodiky OpenUP s rozšířením o best-practice z řady dalších agilních metodik. Jejím rozšířením o Analýzu a návrh testování se značně zvýší její efektivita a účinnost při testování.

Hlavním přínosem je samotná charakteristika a vymezení této specializované disciplíny. Na projektech vývoje softwarových produktů, se kterými jsem se během své praxe zatím setkal, byla tato disciplína týkající se testování vždy rozdělena částečně a nerovnoměrně mezi role byznys analytika, projektového manažera, vedoucího testovacího oddělení a manuálního testera. V současné době je tato disciplína mnohem

rozvinutější. Sjednocení činností do samostatné disciplíny, kterou plní samostatná role test analytika, bude proto značným přínosem.

Tato práce má řešeršní charakter, z čehož vyplývají i použité metody pro vypracování diplomové práce. V rámci práce provádím analýzu současného stavu metodiky MMSP a požadavků na testování a přínosu analytických činností v oblasti testování na malých projektech. Dále se v diplomové práci věnuji analýze veřejně dostupných zdrojů, kde analyzuji, jak se k dané problematice staví ostatní autoři. V práci kladu velký důraz na analýzu nejnovějších technických norem pro oblast testování. Získané poznatky používám pro rozšíření a aktualizaci metodiky MMSP v oblasti analytických činností testování.

1.3 Předpoklady a omezení práce

V diplomové práci přibližuji úzce specializovanou problematiku v oblasti testování na projektech vývoje softwarových aplikací a produktů. Laik, který se v této oblasti nepohybuje, by mohl mít potíže s pochopením některých tematických okruhů. Během zpracování diplomové práce je třeba volit srozumitelné formulace a vysvětlovat i některé základní pojmy.

Disciplína Analýza a návrh testování je dynamicky rozvíjející se obor. Z tohoto faktu usuzuji, že práce by neměla být ohrožena nedostatkem informačních zdrojů.

Při vlastním rozšiřování metodiky MMSP o disciplínu Analýza a návrh testování je zapotřebí zachovat celistvost rozšiřované metodiky. Po jejím rozšíření musí být zachována posloupnost procesů a celková logika.

2 Rešerše zdrojů zabývajících se analýzou a návrhem testování

V této kapitole se zaměřím na vyhledání zdrojů, které se zabývají popisem analýzy a návrhu testování. Cílem kapitoly je nejen nalézt kvalitní a důvěryhodné informační zdroje, ale také přiblížit problematiku specializace testování tak, jak na ní pohlíží jiní autoři.

2.1 Odborná tištěná literatura

Hlavním zdrojem pro diplomovou práci je tištěná odborná literatura. Jedná se především o knihy týkající se testování, metodik vývoje softwaru a tvorby informačních systémů. Mezi těmito knihami je pro mou diplomovou práci velice přínosná kniha *„Tvorba informačních systémů: principy, metodiky, architektury“* vydaná uznávanými pedagogy Vysoké školy ekonomické, v Praze (Bruckner, Voříšek, Buchalcegová a kol., 2012), která komplexně popisuje nejen analýzu a tvorbu informačních systémů, ale přináší i popis jednotlivých disciplín souvisejících s jejich budováním.

Za další bezesporu významnou knihu považuji *„Jak testuje software Microsoft“* od manažerů testování v Microsoftu (Page, Johnston a Rollison, 2009), která se zabývá velmi podrobně testováním z praktického hlediska. Popisuje podrobně techniky úvodní analýzy, mezi něž patří například analýza testovatelnosti, nebo rizik. Dále rovněž popisuje vlastní techniky analýzy testování, jimiž jsou například testování tříd ekvivalence, nebo okrajových hodnot.

Další důležitou knihou je *„Příklady modelů analýzy a návrhu aplikace v UML“* (Buchalcegová a Stanovská, 2013). Kniha podrobně popisuje Metodiku pro Malé Softwarové Projekty (MMSP), kterou budu v rámci této diplomové práce rozšiřovat o výše uvedenou disciplínu Analýza a návrh testování.

Další důležitou knihou je v neposlední řadě kniha *„Řízení kvality softwaru“* (Roudenský a Havlíčková, 2013). Tato kniha popisuje podrobně procesy testování a řízení kvality. Je určena primárně IT specialistům pracujících na pozicích tester, analytik a manažer testování, ale poskytuje užitečné informace pro všechny, kdo se

zajímají o kvalitu softwaru. Autoři se zaměřují mimo jiné na rozlišování jednotlivých úrovní a typů testování, nebo i navrhováním testovacích případů.

Rozsah tištěné odborné literatury je široký, neboť v dnešní době již existuje obrovské množství informačních zdrojů týkajících se tvorby softwaru a jeho testování. Mezi významná vydavatelství této odborné technické literatury patří například Computer Press nebo Oeconomica. To je způsobeno především dynamickým růstem celého odvětví informačních technologií, které se během uplynulých dvou desetiletí rozvíjelo exponenciálně.

2.2 Certifikace a mezinárodní normy

Dalším důležitým informačním zdrojem jsou podklady pro certifikace Mezinárodního sdružení testerů ISTQB (International Software Testing Qualifications Board) (ISTQB, 2015). Tato mezinárodně uznávaná organizace byla založena v roce 2002 ve Skotsku a klade si za cíl zajistit uznání IT specialistům pracujícím v oblasti testování. Organizace zajišťuje certifikace celkem na třech úrovních, jsou to základní, pokročilá a expertní. Pro potřeby diplomové práce je zásadní pokročilá úroveň, protože poskytuje certifikace nejen pro roli Test manažera, ale hlavně také pro role Test analytika a Technického test analytika. Podklady pro studium na tyto certifikace jsou veřejně dostupné na webu ISTQB (ISTQB, 2015).

Mezi další významné informační zdroje patří mezinárodní normy organizací ISO (International Organization for Standardization – Mezinárodní organizace zabývající se tvorbou norem) a IEC (International Electrotechnical Commission - Mezinárodní elektrotechnická komise), které společně vydávají normy tvořící systém normalizace s celosvětovou působností. Pro potřeby mé diplomové práce je relevantní především česká lokalizace normy ISO 29119 (ČSN ISO/IEC/IEEE 29119-1, 2014) týkající se testování softwaru. Tato norma spadá do oblasti norem Softwarového a systémového inženýrství. Skládá se celkem ze čtyř částí, česká lokalizace však zatím obsahuje první dvě. První část popisuje koncepty a definice v oboru testování. Obsahuje velmi obsáhlý rejstřík definic termínů. Druhá část popisuje obecně testovací proces. Tato část se zaměřuje na zásady univerzálního užití testování v jakýchkoli podmínkách. Třetí částí je normalizace dokumentace testování. Poslední čtvrtou částí je popis technik testování. Tato norma zahrnuje normalizaci základních technik a metodik používaných

při testování. V blízké budoucnosti má být uvolněna ještě pátá část normy, která se bude zabývat testováním pomocí klíčových slov (Keyword-Driven Testing). Pro mou diplomovou práci budou nejvíce uspokojovat informační potřebu první a čtvrtá část.

2.3 Elektronické informační zdroje

Dalším informačním zdrojem jsou internetové zdroje. Jsou to například specializované webové portály, komunitní fóra nebo zveřejněné dokumenty zabývající se informačními technologiemi, v tomto případě testováním. Pro vyhledávání veřejně dostupných informačních zdrojů používám vyhledávač Seznam (Seznam.cz, 2015), který je nejpoužívanějším internetovým vyhledávačem v České Republice. Vyhledání elektronických zdrojů zahrnuje i vyhledávání akademických prací, pro tento účel používám portál na vyhledávání zdrojů Vysoké školy ekonomické, v Praze (VŠE, 2015).

Prvním vybraným českým webovým portálem o testování je „*Testování softwaru*“ (Hlava, 2015), který pravidelně vydává odborné články o testování a zároveň podporuje českou testerskou komunitu propagací testerských konferencí. Portál je přehledně rozdělen do čtyř oblastí. Jsou jimi manuální testování, dokumentace, automatizované testování a nástroje. Portál se zaměřuje na vše, co s testováním souvisí a obsahuje i hodnotné informace pro disciplínu Analýzy a návrhu testování.

Dalším vybraným portálem je „*Clever and Smart*“ (Čermák, 2008). Tento webový portál se specializuje především na softwarovou bezpečnost a kvalitu. Autor tohoto webu dosud publikoval 18 zajímavých článků o testování. Patří mezi ně například články o testování SW, testovacím plánu, typy a způsoby testování.

Třetím vybraným webovým zdrojem je internetový blog paní Anny Havlíčkové (Havlíčková, 2014), který se zaměřuje podrobně na disciplínu testování. Jeho autorka je rovněž i autorkou zmíněné knihy „*Řízení kvality softwaru*“ (Roudenský a Havlíčková, 2013) a zároveň i aktivním členem otevřené testerské komunity na webu „*(pro)Test!*“ (pro-Test!, 2013). Tato komunita pravidelně pořádá setkávání a přednášky profesionálů z oboru testování s cílem vytvářet prostor pro sdílení zkušeností mezi testery. Komunita rovněž aktivně publikuje příspěvky na sociální síti Google+ (Google+, 2015).

Z rešerše českých webových stránek vyplynulo, že takových stránek, které se zabývají přímo testováním softwaru, není příliš mnoho a na disciplínu Analýza a návrh

testování se nesespecializuje žádná. Většina webových stránek se zabývá procesem testování pouze okrajově v rámci celé tvorby informačních systémů. Velké množství zobrazených zdrojů se týkalo prezentace firem, poskytování služeb a nabídek práce.

Rešerši anglických internetových zdrojů provádím pomocí vyhledávače Google.com (Google, 2015). Internetových informačních zdrojů v oblasti testování a disciplíny Analýza a návrh testování existuje v anglickém jazyce podstatně více. Ve své rešerši však uvedu jen některé zdroje týkající se Analýzy a návrhu testování. Mezi nejpresnější zdroje patří opět webový portál ISTQB (ISTQB, 2015), kde se kromě již zmíněných podkladů pro certifikace nacházejí i řady zajímavých článků. Mezi velmi přínosné patří článek „*What is fundamental test process in software testing?*“ (ISTQB, 2015b), který diskutuje postavení disciplíny Analýzy a návrhu testování v rámci testovacího procesu a článek „*What is Test analysis? or How to identify the test conditions?*“ (ISTQB, 2015c), který popisuje Analýzu a návrh testování jako samostatnou disciplínu.

2.4 Akademické práce

Mezi další informační zdroje patří i rešerše ostatních diplomových prací. Pro vlastní rešerši se jedná o dva primární tematické okruhy. První okruh se týká obecně testování a Analýzy a návrhu testování. Druhý okruh se týká metodik vývoje softwaru a jejich rozšiřování. V druhém okruhu se zaměřuji především na diplomové práce související s metodikami OpenUp a MMSP.

Prohledal jsem katalogy vyhotovených závěrečných prací předních českých universit (Vysoká škola ekonomická v Praze, Universita Karlova v Praze, České vysoké učení technické a Vysoké učení technické v Brně) a našel jsem několik prací zabývajících se problematikou testování, z nich se jedna zabývá rozšířením metodiky MMSP v oblasti testování. Je to diplomová práce „*Testování a kvalita softwaru v metodikách vývoje softwaru*“ (Vachalec, 2014), která popisuje testovací proces v agilních metodikách a ve své praktické části popisuje obecný návrh rozšíření testování metodiky MMSP, avšak disciplínu Analýza a návrh testování neobsahuje.

Diplomová práce „*Zlepšování softwarových procesů v oblasti testování*“ (Libík, 2014) popisuje zlepšování testovacího procesu. Využívá k tomu část normy ISO/IEC 29110 po jejímž použití na malém projektu navrhuje úpravy a její zlepšení.

Výchozí diplomovou prací však je „*Lokalizace a přizpůsobení metodiky OpenUP*“ (Rejnková, 2011). Tato diplomová práce popisuje vytvoření nové metodiky MMSP lokalizací a přizpůsobením metodiky vyvinutou společností IBM. Metodika MMSP byla vytvořena pro potřeby výuky na Vysoké škole ekonomické v Praze. Jedná se o kompletní českou metodiku, kterou je možné převzít a dále upravovat. Právě tuto metodiku budu v rámci své diplomové práce rozšiřovat.

Diplomová práce „*Metodika testování podle mezinárodních praktik a standardů*“ (Králová, 2013) popisuje metodiky a standardy z pohledu testování. Získané poznatky autorka používá k vypracování metodiky testování. Autorka se ve své diplomové práci zaměřuje na řízení testování podle mezinárodních norem ISO 9000, který se zabývá obecně řízením kvality, ISO/IEC 12207, který se zaměřuje na životní cyklus.

Diplomová práce „*Prototypování při vývoji softwaru*“ (Nývlt, 2012) rozšiřuje metodiku MMSP o roli vývojáře uživatelského rozhraní a grafika. Toto rozšíření rovněž obsahuje úpravu rolí, úloh, pracovních produktů a životního cyklu.

3 Charakteristika disciplíny Analýza a návrh testování a vymezení základních pojmů

V této kapitole se zaměřuji na charakteristiku a vymezení disciplíny Analýza a návrh testování, včetně jejích základních pojmů a kontextu v životním cyklu projektu a testování. Zaměřuji se nejen na vysvětlení základních pojmů a uvedení do problematiky, ale i na to, co je její náplní a jaký má význam z hlediska tvorby informačních systémů a softwarových aplikací, aby bylo možno lépe pochopit, proč jsem se tuto relativně novou disciplínu rozhodl zahrnout do metodiky MMSP.

3.1 Vymezení pojmu disciplína Analýza a návrh testování

Analýza a návrh testování je disciplína v oblasti vývoje IS/ICT technologií. Disciplíny metodiky MMSP a metodiky RUP, ze které MMSP vychází, jsou seskupení aktivit vykonávaných v různých fázích projektu. Skládá se z úloh, které tvoří dílčí kroky vedoucí k dosažení požadovaného výsledku. Disciplína je souhrn, kolekcí činností, které se týkají konkrétní oblasti zájmu v průběhu IS/ICT projektu (Kruchten, 2000).

V angličtině existuje pro disciplínu Analýza a návrh testování původní označení „*test analysis*“, kterou vykonává IT specialista zastávající roli „*test analyst*“. Překlad role „*test analyst*“ odpovídá dobře výrazu test analytik. Přeložili bychom však pojem „*test analysis*“ do českého jazyka doslovně jako „test analýza“, byl by tento překlad velmi hrubý a nepřesný. Pro správné pojmenování disciplíny po konzultaci s doc. Ing. Alenou Buchalceovou, Ph.D. je nejvhodnější používat termín tak, jak jej uvádějí autoři skript pro certifikace ISTQB (Müller a kol., 2011, str. 15) a to:

„Analýza a návrh testování“

Podle slovníku cizích slov (Mojzisek, 2015) pojem analýza znamená:

„Vědecká metoda založená na dekompozici celku na elementární části. Cílem analýzy je identifikovat podstatné a nutné vlastnosti elementárních částí celku, poznat jejich podstatu a zákonitosti.“

Dovolují si tedy shrnout poznatky do tvrzení, že Analýza s návrh testování je disciplína, zabývající se poznáním, rozbořem a identifikací zadání, potřeb a požadavků na oblast testování.

3.2 Přínos disciplíny Analýza a návrh testování

Hlavní přínos disciplíny Analýza a návrh testování je zefektivnění celé oblasti testování, protože začleňuje do procesu testování analytické činnosti, které umožňují výrazně detailnější kontrolu vstupů a porozumění testovaného objektu. Díky této disciplíně se testování značně rozšířila identifikace podmínek a potřeb testování, které pomáhají lépe stanovit priority testování a zlepšit pokrytí testy (McKay a kol., 2012).

Testování je velice nevědný obor. Má malou šanci dostat se do ideálního bodu, protože i když dochází v každé iteraci k odstranění problémů, tak v další iteraci se objeví další problémy, které nebyly dosud odhaleny, nebo vznikly nově. Analýza a návrh testování se snaží maximalizovat přiblížení ideálního stavu, i když ho testovací týmy nemohou nikdy dosáhnout. Disciplína napomáhá (McKay a kol., 2012):

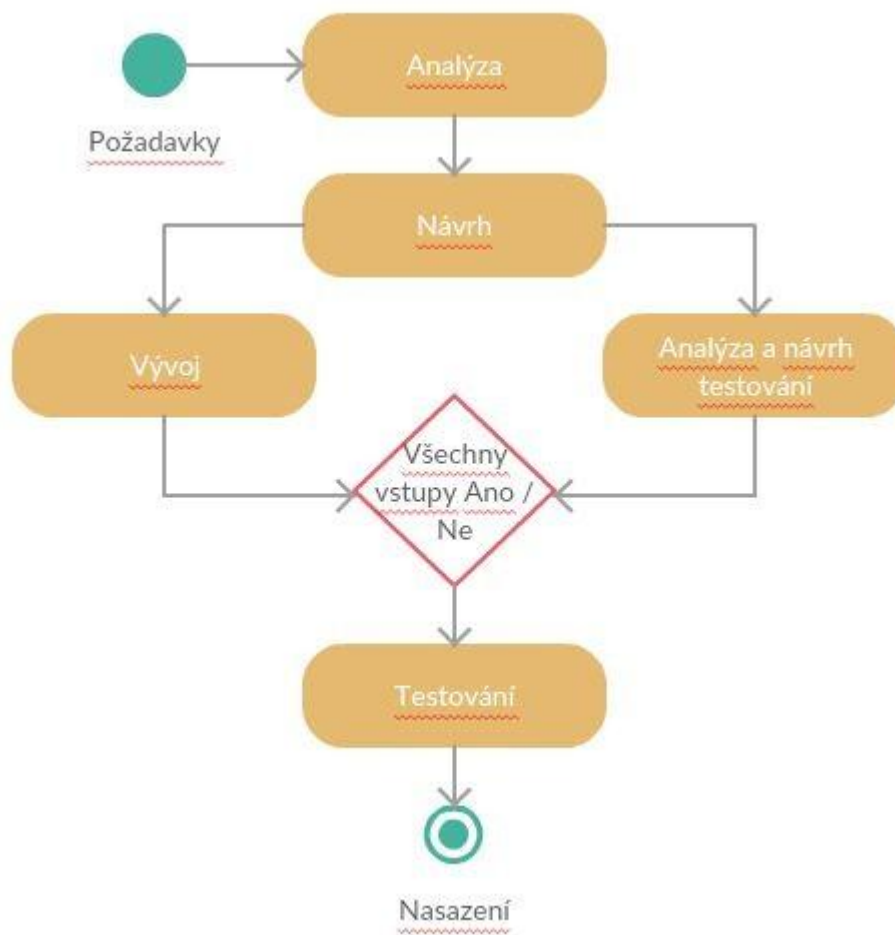
- stanovit priority testování,
- analyzovat rizika testování,
- vyhodnotit testovatelnost předmětu testování,
- zvýšit přesnost a důslednost testovacích scénářů,
- identifikovat potřebná testovací data,
- identifikace požadavků na testovací prostředí,
- dodržovat normy testování,
- zpracovávat výstupy testování
- neustále zlepšovat proces testování s každou iterací,
- zvyšovat spokojenost klienta,
- celkově zlepšit hodnocení kvality vyvíjeného produktu.

Zahrnutí disciplíny Analýza a návrh testování do testovacího procesu metodiky MMSP zavádí značné množství analytických činností. Rozšířenou metodiku MMSP je možné použít i pro větší testovací týmy, nebo pro náročnější a složitější projekty. Rozšíření metodiky MMSP zajišťuje efektivní stanovení strategie testování, komplexnější vytvoření plánu testování a včasnější odhalení základních chyb.

3.3 Úvod do problematiky testování softwaru

Na úvod je třeba vysvětlit, kde se disciplína Analýza a návrh testování využívá. Většina softwarových produktů a softwarových aplikací se v současné době již tvoří výhradně formou projektu. Milton Rosenau se ve své knize „*Řízení projektů*“ popisuje projekt (Rosenau, 2000) jako jedinečný, časově ohraničený, realizovaný za pomoci zdrojů a mající za cíl vytvoření unikátního výstupu, v tomto případě se jedná o unikátní softwarový produkt. Vývoj softwarového produktu se provádí buď formou interního vývoje, nebo formou outsourcingu na zakázku. Každý projekt by měl být podpořen adekvátní metodikou vývoje softwaru, která v sobě zahrnuje i životní cyklus vývoje softwaru. Postavení oblasti testování a disciplíny Analýza a návrh testování v rámci životního cyklu projektu se odvíjí právě od druhu zvolené Metodiky vývoje softwaru, o tom ale více v kapitole 4.

Pro správné pochopení významu disciplíny Analýza a návrh testování zatím stačí zjednodušené schéma (Obrázek 1). Na schématu jsou znázorněny obecné základní kroky životního cyklu většiny metodik vývoje softwaru. Projekt začíná zpravidla formulací požadavků (Kruchten, 2000). Zadavatel (nebo ho též můžeme nazývat klient) a dodavatel (systémový integrátor) se společně dohodnou na co pokud možno nejpresnější zadání. Součástí definice zadání je i soupis akceptačních kritérií. Jsou to kritéria, která musí výsledný produkt splňovat, aby bylo možné dojít k předání výsledného softwarového produktu. V praxi se však zadání projektu velmi často mění. Z toho důvodu se doporučuje využívání agilních metodik vývoje softwaru, které pomáhají na náhlé změny v zadání projektu lépe reagovat. Dalším krokem je analýza. Ta má na starosti prozkoumání zadání a vytvoření co nejefektivnějšího návrhu realizace. Vytvoření návrhu obsahuje nejčastěji architektonický model, datový model a dokument specifikace požadavků. Na základě toho návrhu vývojáři implementují softwarové řešení, které pak testéři testují. Celý proces vývoje pak končí dodáním softwaru zadavateli.



Obrázek 1: Obecné schéma životního cyklu vývoje softwaru (Zdroj: autor)

Analýza a návrh testování vznikla oddělením analytických činností v oblasti testování do samostatné role, kterou je ve většině případů už nezastává tester, ale test analytik. Disciplína Analýza a návrh testování není závislá na výstupech procesu vývoje, ale její vstupy jsou tvořeny již návrhem realizace a jejím výstupem jsou komplexní podklady pro vlastní testování (Obrázek 1).

V této disciplíně je hlavním úkolem určení celé strategie testování, definování milníků za testování a vytvoření plánu testování ve spolupráci s vedením testovacího týmu. Analýza a návrh testování určuje rozsah testování v rámci vývoje, definuje cíle projektu za oblast testování a zajišťuje efektivní podporu vedení testování. Disciplína zkoumá relevanci požadavků na systém, zabývá se realizovatelností akceptačních kritérií. To znamená, že v rámci této disciplíny se provádí i analýza testovatelnosti. Úlohou disciplíny je také starat se o prevenci, aby nedošlo k provádění zbytečných, nebo neefektivních úkonů během aktivity provádění testů.

Analýza a návrh testování se snaží co nejvíce omezit rizika nesplnění akceptačních kritérií. Čím lépe jsou zvoleny priority testování, typy testů a vypracované testovací scénáře, tím je menší šance, že dojde k neodhalením nepřesností od zadání během testování.

Za rozvržení rolí v testovacím týmu bývá odpovědné vedení projektu. Jsou jimi buď projektový manažer, manažer testování, nebo vedoucí testerů. Rozvržení testovacího týmu spadá do disciplíny řízení projektu. Analýza a návrh testování však zkoumá potřebné zdroje pro testování. Jedná se o zjištění potřeby lidských nebo materiálních zdrojů potřebných k zajištění potřebné kvality vyvíjeného softwarového produktu.

Analýza a návrh testování má na starosti výběr typů testů. Typů testů existuje velké množství a jejich výběr se promítá do plánu testování. Například česká lokalizace technické normy pro testování uvádí (ČSN ISO/IEC/IEEE 29119-1, 2014) testy akceptační, regresní, penetrační, výkonu, udržitelnosti, fail-over testy atd. Pro správnou volbu typů testů se musí odpovědná osoba uvažovat rozsah produktu, velikost testovacího týmu, schopnosti testerů a časové kapacity na testování. Zjednodušeně řečeno Analýza a návrh testování určuje, co a jakým způsobem se bude testovat. V konečném důsledku disciplína velmi ovlivní kvalitu celého testování. Na druhou stranu Analýza a návrh testování dokáže díky správnému návrhu testování zásadním způsobem usnadnit vlastní tvorbu testovacích scénářů.

Disciplína Analýza a návrh testování nedisponuje rozhodovacími pravomocemi. Podílí se značnou měrou na výstupní kontrole, ale nemá řídicí funkce. Disciplína vesměs pouze hodnotí výslednou kvalitu softwarového produktu. Anna Havlíčková, česká odbornice v oblasti testování, velmi výstižně tvrdí (Havlíčková, 2014) o testování toto:

„Já nezajišťuji kvalitu, já ji hodnotím. Hledám informace skrz kladení otázek softwaru a zpracovávám je. Poté třídím, zhodnocuji a poskytuji tyto informace ostatním.“

Celkové zhodnocení zda produkt splnil, či nesplnil očekávanou kvalitu a může být nasazen ke klientovi, má v praxi pouze doporučující charakter.

3.4 Shrnutí

Analýza a návrh testování je relativně nová disciplína v oblasti testování softwarových produktů, která se zabývá poznáním, rozbořením a identifikací zadání, potřeb a požadavků na oblast testování. Disciplínu Analýza a návrh testování zastává na projektu role test analytika. Ten má na starosti v rámci disciplíny definici strategie testování a plánu testování. Disciplína zkoumá testovatelnost softwarového produktu a analyzuje možná rizika, která by mohla mít vážný dopad na výsledky testování. Zavedením této disciplíny do metodiky MMSP se celkově zvýší efektivita procesu testování.

4 Modely a metodiky vývoje softwaru dle postoje k testování

V této kapitole analyzuji historický průřez vybraných modelů a metodik vývoje softwaru s ohledem na jejich vývoj postoje k testování a k analytickým činnostem v oblasti testování, které se na projektech vývoje softwaru používají. Úkolem kapitoly je nastínit, jak se testování v metodikách vyvíjelo a popsat v kapitole jak se jednotlivé metodiky staví k otázce analytických činností v procesu testování. Vytvoření přehledu vybraných metodik je zároveň průzkumem metodik, jak přistupují k otázce testování. V rámci kapitoly také vysvětluji, proč jsem si pro rozšíření vybral metodiku MMSP a vysvětluji její zařazení mezi metodikami.

4.1 Obecně o modelech a metodikách vývoje softwaru

Model vývoje softwaru je zjednodušené schéma doporučených postupů k dosažení požadovaného cíle. Model je sekvenční proces po sobě jdoucích činnostech (Wikipedia, 2014). Je důležité neplést si model s metodikou vývoje softwaru. Mezi modely vývoje softwaru patří například Vodopád, nebo Spirála.

Metodika je na rozdíl od modelu komplexnější. Metodika vývoje softwaru definuje kromě postupů a činností i procesy, praktiky, role techniky a produkty používané při vývoji (Buchalceová, 2004). Zavedená metodika má vliv nejen na celý vývoj, ale i na celý proces testování včetně případné Analýzy a návrhu testování. Metodika tvoří obecný rámec, definuje obecné postupy a pravidla za účelem vývoje softwaru. V praxi se metodika musí přizpůsobit a upravit podle potřeb vybraného projektu, protože jednotlivé organizace se od sebe velmi liší podnikovou kulturou, schopnostmi svých zaměstnanců i používanými technologiemi. Existuje i možnost stanovit si postupy vlastní, ale jen na nižších úrovních, obecný rámec však musí být zachován.

4.1.1 Dělení metodik vývoje softwaru

Metodiky vývoje softwaru se dělí na dvě skupiny. Jsou to rigorózní a agilní metodiky. Pro rigorózní metodiky je typické, že jsou velmi obsáhlé a podrobné. Tyto metodiky vycházejí z předpokladu, že lze definovat všechny požadavky, procesy

činnosti definovat předem, neboť se budou opakovat. Jedná se o tzv. těžké metodiky. Obsahují popis procesů propracovaný do nejmenších detailů. Výjimkou není ani proces testování včetně všech jeho analytických činností. Jak uvádí Buchalcegová v článku „*Stav používání agilních metodik v ČR*“ (Buchalcegová, 2006), v současné době je použití rigorózních metodik na ústupu, neboť nepodporují snadnou a flexibilní změnu v závislosti na měnící se požadavky zadavatelů. Z tohoto důvodu jsem si nevybral možnost, rozšiřovat některou z rigorózních metodik o disciplínu Analýza a návrh testování.

Oproti tomu agilní metodiky podporují flexibilní změny v procesu vývoje a na změny požadavků zadavatele. Buchalcegová uvádí (Buchalcegová, 2006), že agilní metodiky umožňují rychlé zavedení IS/ICT. V agilních metodikách je část odpovědnosti přesunutá na zákazníka tak, že on rozhoduje o prioritách dohodnutých požadavků a podrobuje je důkladnému testování. Smyslem agilních metodik je vyvinout softwarový produkt (nebo jeho část), dát ho zákazníkovi a upravit ho na základě zpětné vazby. Tento způsob umožňuje výrazným způsobem zrychlit celý proces vývoje, který probíhá iterativním a inkrementálním způsobem. Iterace v tomto případě znamená cyklus vývoje od analýzy až po nasazení jednoho, nebo více požadavků k zadavateli. Inkrementální způsob, znamená vývoj po částech. Buchalcegová uvádí (Buchalcegová, 2006) seznam následujících agilních metodik a přístupů:

- Dynamic Systems Development Method (DSDM),
- Adaptive Software Development (ASD),
- Feature–Driven Development (FDD),
- Extrémní programování (Extreme Programming, XP),
- Lean Development,
- Scrum ,
- Crystal metodiky,
- Agilní modelování (Agile Modeling),

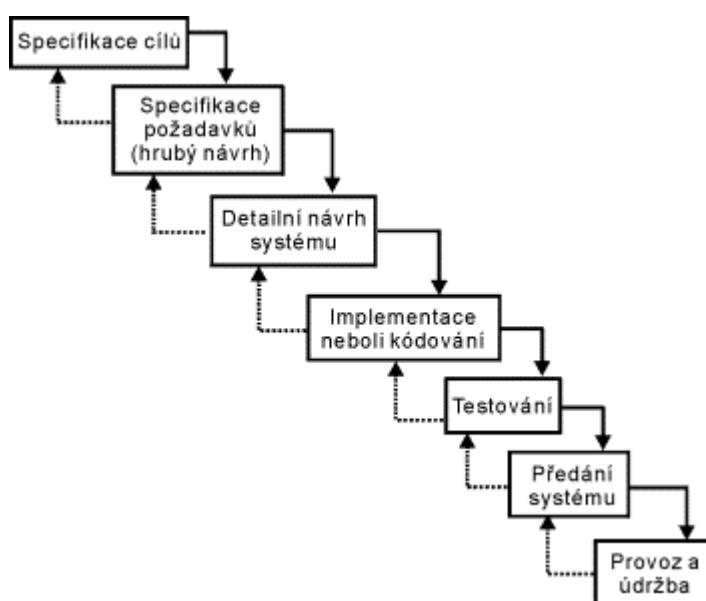
Z těchto metodik jsou dnes aktuální Lean Development, Scrum, Crystal metodiky a nově k nim uvedu například Kanban, nebo Metodiku pro malé softwarové projekty.

Existují však metodiky, které je obtížné zařadit. Takovým příkladem je například RUP (Kruchten, 2000). Tato metodika vykazuje znaky těžkých metodik, ale zároveň vykazuje iterativní přístup a agilní prvky k vývoji softwaru.

4.2 Modely životního cyklu softwaru a jejich vztah k testování

Tradiční modely vývoje softwaru nejsou plnohodnotnými metodikami, jedná se spíše o shrnutí principů a doporučení životního cyklu projektu. Nelze v nich ani hledat sofistikované řešení pro oblast testování softwaru. Jejich přínos však není zanedbatelný, protože se z historického hlediska jednalo o první pokusy zavedení řádu do problematiky vývoje software.

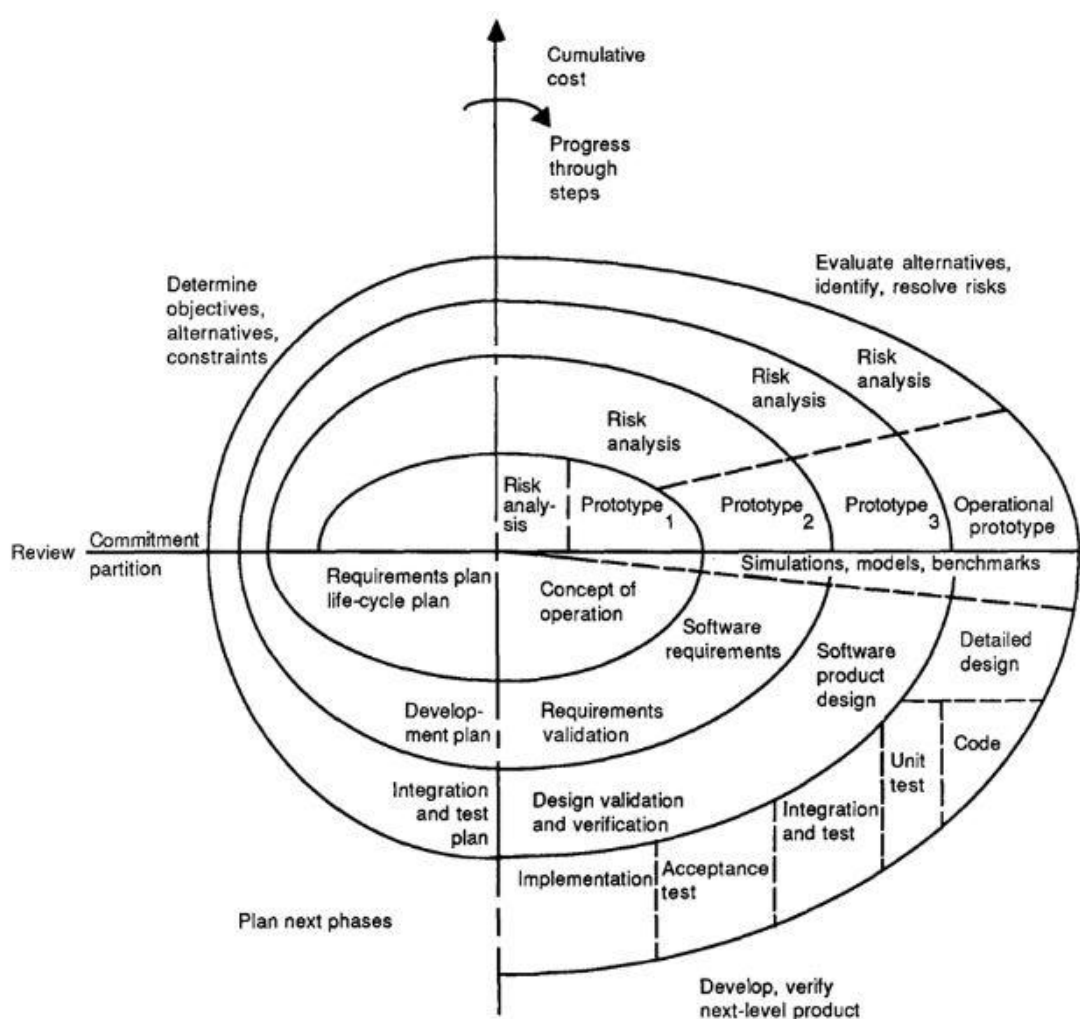
Vodopád (Waterfall) byl jedním z prvních modelů životního cyklu vývoje softwaru. Poprvé byl definován Dr. Winstonem Roycem v knize „*Managing the development of Large Software Systems*“ (Royce, 1970). Princip spočívá v postupném provádění jednotlivých činností v rámci vývoje softwarového produktu. Předpokladem pro start činnosti je vždy dokončení činnosti předešlé. Jak je vidět na schématu vodopádového modelu (Obrázek 2), tak jednotlivé činnosti na sebe navazují a vzájemně se nepřekrývají. Dodání produktu probíhá formou „velkého třesku“.



Obrázek 2: Vodopádový model (Royce, 1970)

Vodopádový model se testováním zabývá jen minimálně. Autor klade velký důraz na dokumentaci, podle které programátor provádí experimentální testování. Technika počítá s testováním vždy hotového softwarového produktu. Autor vychází z předpokladu, že naprostá většina chyb se odhalí vizuální kontrolou během kódování. Případné rozšíření této metodiky o disciplínu Analýza a návrh testování postrádá smysl. Technika je sice jednoduchá a snadno dostupná, ale analytické činnosti v oblasti

Spirála (Spiral model) je technika kterou publikoval Barry Boehm koncem osmdesátých let ve své knize „*A Spiral Model of Software Development*“ (Boehm, 1989). Principem této techniky je inkrementální vývoj neustálým opakováním kroků vývoje, které vycházejí z vodopádového modelu. Postupně se na produkt nabalují další části systému na vyšší úrovni. Na schématu spirálového modelu (Obrázek 3) je vidět, že se cykly řídí vypracováváním dílčích analýz rizik z předešlého cyklu. Jedná se o techniku vývoje řízeného rizikem (Risk-Driven Development).



Spirálový model už počítá s oblastí testování. Technika zahrnuje testování jednotek, integrace a akceptace. Vlastní testování probíhá formou validace vždy na konci každé iterace. Výsledky testování se analyzují a získané poznatky se promítají do analýzy rizik, ze které vychází formulace zadání pro další kolo vývoje. Technika

nezapře, že vychází z vodopádového modelu. Akceptační testování probíhá na závěr vývoje nad celým již hotovým produktem.

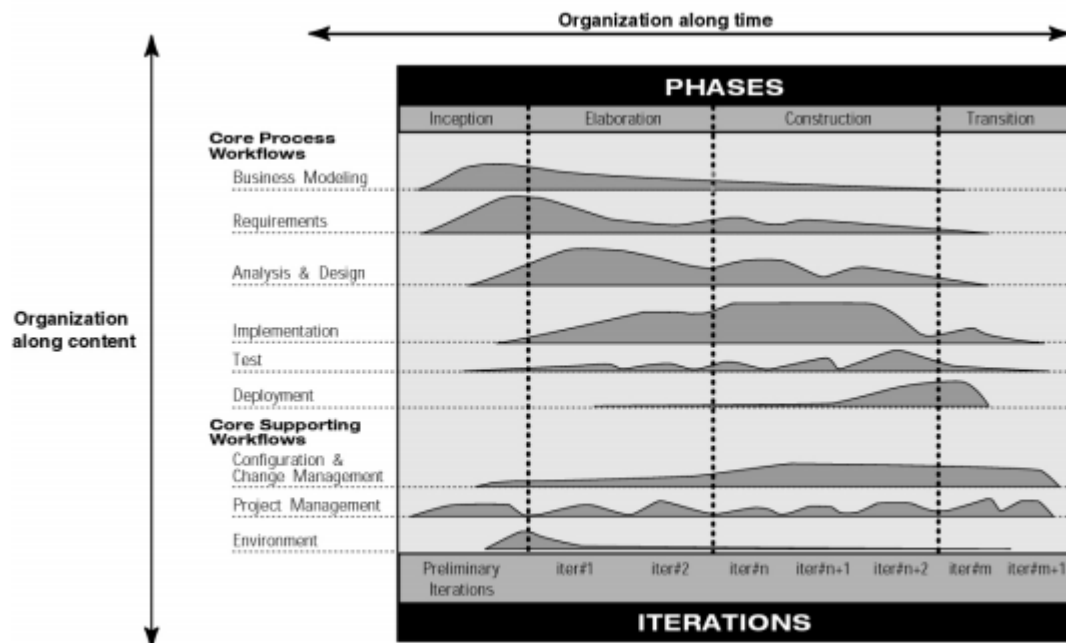
4.3 Přístup rigorózních a agilních metodik k testování

V kapitole 4.1 jsem popsal, co to jsou metodiky vývoje softwaru a v kapitole 4.1.1 jaké jsou jejich kategorie. Všechny metodiky v současné době přistupují k problematice testování a hodnocení kvality softwarového produktu jako k plnohodnotné disciplíně. Některé metodiky jsou propracovány velmi podrobně a zahrnují i využití analytických činností v oblasti testování. Pro rozbor metodik vyberu pouze některé z nich.

Rational Unified Proces (RUP) je velmi rozsáhlá metodika (Kruchten, 2000) vyvinutá společností Rational Unified Software, kterou v roce 2003 pohltila IBM. Jde původně o rigorózní těžkou metodiku, která se v průběhu času vyvíjela (stejně jako celé odvětví informačních technologií) a v současné době nese mnoho znaků agilních metodik. Metodika RUP je velmi propracovaná a lze ji použít téměř na jakýkoli typ projektu. Hodí se však především na větší projekty. Jedná se objektivě orientovanou metodiku, která je založena na iteracích probíhajících v cyklech. Každý projekt je definován čtyřmi fázemi (Obrázek 4), které jsou na sobě posloupně závislé a připomínají tak vzdáleně vodopádový model. Jsou to fáze (Kruchten, 2000):

- Zahájení
- Příprava
- Konstrukce
- Předávání

Každá fáze může být rozdělena do více iterací (cyklů). To umožňuje flexibilní zapracovávání změnových požadavků během již probíhajícího vývoje.



Obrázek 4: Schéma iterativního modelu metodiky RUP (Kruchten, 2000)

Metodika obsahuje řadu podrobně definovaných disciplín. Tyto disciplíny jsou všechny činné během celého průběhu projektu (Obrázek 4). Mezi disciplíny metodiky RUP patří (Kruchten, 2000):

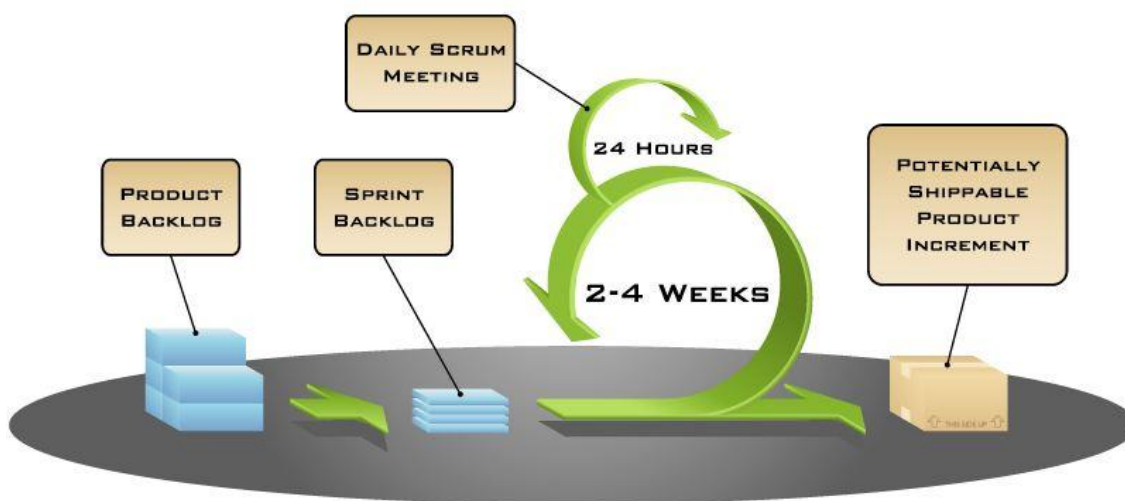
- Tvorba modelu
- Správa požadavků
- Analýza a návrh
- Implementace
- Testování
- Nasazení
- Konfigurace a změny
- Řízení projektu
- Správa prostředí

U metodiky RUP je velmi důležité, že klade velký důraz na oblast testování. Obrázek 4 ukazuje, že disciplína testování je činná od samého začátku projektu. Hodnocení kvality však neprobíhá pouze v rámci testování, ale i v jiných činnostech, jako je kontrola správnosti při správě požadavků, nebo kontrola správnosti analýz. Metodika RUP je na principu ověřování postavena. Samotná disciplína testování se dále dělí na řadu dalších podřízených činností. Metodika RUP rovněž podrobně definuje role

pro testování. V metodice jsou charakterizovány specializace oblasti testování, což značí vysokou vyspělost procesu testování v této metodice. Jsou to role test manažer, test analytik, test designer a tester.

Právě role test analytika a test designera mají na starosti analytické činnosti v oblasti testování. Dle metodiky RUP (Kruchten, 2000) je test analytik odpovědný za stanovení způsobu testování. Defínuje, jaké jsou cíle testování, jaké testy budou prováděny, jaká je požadovaná konfigurace prostředí, jaké nástroje budou použity, nebo jaká jsou potřeba data pro testování. Test analytik dle RUP je odpovědný za testovací případy. Metodika rovněž klade velký důraz na analýzu případů užití. Role test designera se dle RUP zabývá vytvářením a přípravou testů. Má na starosti ověření, zda je možné danou verzi testovat, zajišťuje přístupy a konfiguraci testovacího prostředí, prvky a mechanismy testovatelnosti. Z ohledu testování je metodika velmi propracovaná

Scrum je agilní metodika (Mountain Goat Software, 2005) pracující na principu inkrementálních a iterativních cyklů. Tato metodika se však vymyká svým přístupem k řízení projektu. Ze schématu cyklu metodiky Scrum níže (Obrázek 5) je vidět, že každý cyklus má dvě fáze. Prvním je klidové období, kdy probíhá vyhodnocení předešlého cyklu, analýzy a příprava a plánování na druhou fázi. Druhou fází je tzv. sprint, kdy probíhá vlastní iterace vývoje, ta je časově omezena, trvá většinou 2 až 4 týdny. V této fázi je kladen velký důraz na testování. Během sprintu dochází k předávání zpětné vazby každý den na pravidelných meetingách.



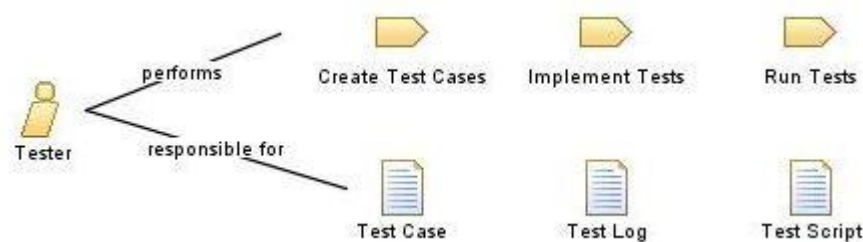
Obrázek 5: Schéma cyklu metodiky Scrum (Mountain Goat Software, 2005)

Scrum nedefinuje role ani jednotlivé disciplíny pro správný vývoj softwaru. Definuje odpovědnou osobu (vlastníka produktu), správce metodiky a scrumový tým, který vystupuje jako celek. Členové scrum týmu nejsou rozřazeni do konkrétních rolí, ale každý pracuje na tom, co má v daný moment nejvyšší prioritu. Tím by měla být dosažena maximální zastupitelnost. Tento přístup má však jedno úskalí. Je velmi obtížné nalézt IT specialisty, kteří by dokázali zastat disciplínu analýzy, vývoje a testování dohromady. V praxi dochází však pouze k mírnému prolínání rolí. Obecně platí, že každý člen scrum týmu plní ty úkol, které jsou mu nejbližší (Mountain Goat Software, 2005).

Ve Scrumu probíhá testování tak, že tester provádí většinu analytických činností v období plánování. Ve sprintu se tester soustředí primárně na provádění testovacích scénářů. Velkou výhodou pro testera je, že je v bezprostředním kontaktu s ostatními členy týmu. Může tak s nimi konzultovat případné problémy, nebo nejasnosti při vytváření strategie testování a plánu testování. Pro Scrum je typické, že se testéři aktivně podílejí na hledání řešení a jsou podrobně zasvěceni do analýzy a realizace požadavků. Testéři jsou přítomni na projektu od samého počátku vývoje. To usnadňuje pochopení všech případů užití vyvíjené aplikace. Komplikací při návrhu testování ve Scrumu je, že vzniká ve stejný okamžik jako vlastní analýza požadavků. Testéři pak mají nedostatek oficiálních podkladů pro tvorbu testů. V rámci scrumu se proto testéři hodně soustředí na automatizaci testování (Mountain Goat Software, 2005).

OpenUp je agilní metodika vytvořená senior software inženýrem Ricardem Balduinem (Balduino, 2007). Autor metodiky se aktivně podílel na vývoji metodiky RUP ve společnosti IBM. Ricard Balduin je rovněž autorem nástroje Eclipse Proces Framework Composer. Jde o webový nástroj pro vytváření, editaci a prohlížení metodik. Při práci s RUP však došel k poznatku, že tato metodika je příliš komplexní pro použití na menších projektech. Proto vytvořil metodiku OpenUp.

Jedná se o minimalistickou metodiku, která je odvozená od metodiky RUP a má sloužit pro použití na malých projektech mající za účel vývoj nekritických informačních systémů. OpenUp obsahuje pouze základní principy agilního vývoje. Z metodiky RUP přebírá 4 základní fáze projektu (viz strana 18) a některé základní disciplíny, které jsou nutné pro chod projektu. Testování je v metodice popsáno pouze obecně.



Obrázek 6: Úloha role testera v metodice OpenUP (IBM, 2011)

Z obrázku znázorňujícím úlohu Testera v metodice OpenUp (Obrázek) vyplývá, že metodika uvažuje prakticky pouze minimální manuální testování. Tester vytváří a provádí testy. Tester kromě analýzy výsledků testů prakticky žádné analytické činnosti nevykonává.

4.4 Rozšiřovaná metodika MMSP

Metodika pro malé softwarové projekty (MMSP) byla vytvořena českou lokalizací metodiky OpenUp s rozšířením o best-practice z dalších metodik. MMSP je agilní metodika, která původně vznikla jako didaktická metodika na Vysoké škole ekonomické v Praze v rámci diplomové práce „*Lokalizace a přizpůsobení metodiky OpenUP*“ dnes již Ing. Petrou Rejnkovou (Rejnková, 2011). MMSP může být použita ale i na reálných projektech v praxi. MMSP je detailně popsána v knize „*Příklady modelů analýzy a návrhu aplikace v UML*“ (Buchalceová a Stanovská, 2013) a je veřejně dostupná na portálu kitsem.vse.cz. Metodika MMSP je metodikou vývoje softwaru, kterou budu v rámci diplomové práce rozšiřovat o novou disciplínu Analýza a návrh testování (viz kapitola 3).

Životní cyklus projektu modle metodiky MMSP má 4 fáze stejné jako metodika RUP a OpenUp. Jsou to fáze Zahájení, Rozpracování, Konstrukce a Zavedení. Kromě životního cyklu je metodika definována oblastmi Role, Disciplíny a Pracovní produkty. Role je obecně chápána úloha v projektovém týmu, která vykonává určitou disciplínu, nebo disciplíny za účelem vytvoření pracovního produktu. Současná verze metodiky MMSP definuje následující role:

- Architekt
- Projektový manažer
- Analytik
- Tester

- Vývojář
- Zainteresoovaná strana
- Nedefinovaná role

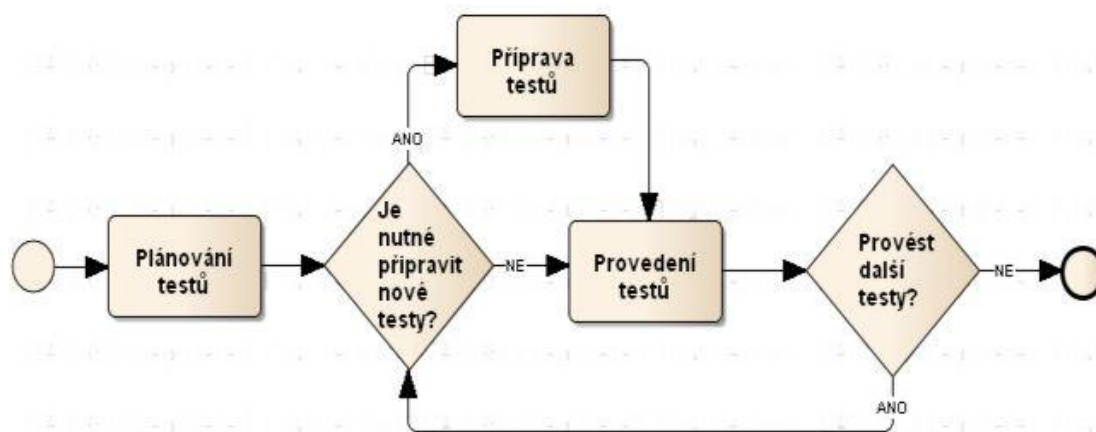
Disciplíny na projektu mají za cíl plnění úloh, za účelem dosažení stanovených cílů, kterými jsou například pracovní produkty. Jsou tvořeny zpravidla sekvencí neurčitého počtu kroků. Disciplíny jsou vykonávány rolemi. Disciplíny definované v metodice MMSP jsou:

- Řízení projektu
- Řízení změn a konfigurací
- Požadavky
- Architektura
- Vývoj
- Testování.

Pracovní produkty jsou objekty, které jsou vytvářeny, nebo upravovány během provádění úloh vývoje softwaru.

Testování v metodice MMSP provádí IT specialista plnící roli Tester (Rejnoková, 2011). Metodika doporučuje splňovat jistou míru zastupitelnosti, proto se na testování může podílet i více rolí. Tester, který má však nejvíce zkušeností s testováním, má hlavní odpovědnost za hodnocení kvality produktu. Testování provádí tester za pomoci jiných členů. Nejčastěji testerovi pomáhá Analytik, nebo Vývojář (Unit testy).

Na schéma testovacího cyklu (Obrázek) je zobrazeno jak testování v metodice MMSP probíhá. Tester má na starosti naplánování, tvorbu a provedení testů. Po dokončení testování provede tester analýzu výsledku testování. Ta má za cíl mimo jiné identifikovat, zda je třeba provést další testy. Metodika MMSP na rozdíl od metodiky OpenUP obsahuje základní prvky analytických činností v oblasti testování. Tester musí vytvořit plán testů, provést naplánování a sestavit seznam testovacích scénářů, vytváří testovací sady a analýzu výsledků testování. Analytické činnosti v oblasti testování v metodice MMSP nezahrnují většinu aktivit z disciplíny Analýza a návrh testování (definování strategie testování, analýza rizik testování, analýza testovatelnosti, milníky v rámci plánu testů, časové odhady pracnosti, atd.), o které by bylo vhodné tuto metodiku rozšířit.



Obrázek 7: Testovací cyklus metodiky MMSP (Rejnková, 2011)

Poslední aktuální verze metodiky MMSP, ze které budu vycházet je 1.5 uvedená na webových stránkách <http://mmsp.czweb.org/> (Rejnková, 2011b), která byla vytvořena pomocí Eclipse Process Framework Composer jako plug-in metodiky OpenUp. Od doby vydání byla metodika MMSP dvakrát rozšířena. Poprvé v diplomové práci „*Prototypování při vývoji softwaru*“ (Nývlt, 2012), kde byla metodika rozšířena o úlohy prototypování v oblasti vývoje. Podruhé v diplomové práci „*Testování a kvalita softwaru v metodikách vývoje softwaru*“ (Vachalec, 2014) byla autorem rozšířena o agilní přístupy. Faktory, které mě vedly k rozšíření metodiky MMSP o disciplínu Analýza a návrh testování:

- Česká lokalizace metodiky – jedná se o metodiku dostupnou v českém jazyce
- Doporučení vedoucí diplomové práce – doporučení, abych téma nové rozvíjející se disciplíny Analýza a návrh testování popsal v rámci rozšíření metodiky MMSP
- Dostupnost zdrojů – metodika je dostupná z veřejných zdrojů, ve formě manuálu v podobě diplomové práce (Rejnková, 2011), tak i ve formě webových stránek (Rejnková, 2011b).
- Didaktická povaha metodiky – metodiku je možné dále rozšiřovat v rámci výuky na VŠE v Praze

4.5 Shrnutí

Disciplína testování bývala dlouho zanedbávanou oblastí. Základní techniky vývoje softwaru obsahovaly testování pouze jako podpůrnou činnost. Teprve až s rozvojem těžkých metodik se testování stalo důležité. Těžké metodiky jsou sice detailně propracované a jako například metodika RUP zahrnují testování s detailně specifikovanými úlohami včetně analytických činností a návrhu v oblasti testování. Tyto metodiky se těžko přizpůsobují změnám v průběhu projektu a jsou určeny primárně pro velké projekty. Agilní metodiky, které dokáží pružně reagovat na změny požadavků vyvíjeného softwarového produktu a obsahují popis procesu testování, ale bez specifikace analýzy a návrhu v této oblasti.

V současné době je testování plnohodnotnou disciplínou, která se stává stále specifitější a je na ni kladen čím dál větší důraz. Tato disciplína pomáhá odhalovat chyby dříve, ještě před samotným dovyvinutím softwarového produktu, zvyšuje efektivitu testování, šetří peníze a čas. Disciplína testování může zahrnovat plno analytických činností, které nejsou zahrnuty v metodice, kterou by bylo možné použít pro potřeby vývoje menších nekritických projektů. Metodika MMSP je agilní metodika určená primárně pro malé projekty a je možné ji rozšířit v oblasti testování o disciplínu Analýza a návrh testování.

5 Analýza a návrh testování z pohledu certifikací ISTQB

V této kapitole se zaměřím na přístup k analytickým činnostem v oblasti testování z pohledu Mezinárodního sdružení testerů ISTQB. Cílem kapitoly je zjistit jak ISTQB přistupuje k současnému testování a disciplíně Analýza a návrh testování. Tento rozbor je důležitý, protože ISTQB je v současné době nejuznávanější na světě. ISTQB ve svém know-how zahrnuje disciplínu Analýza a návrh testování v rámci certifikace na roli test analytika. Získané poznatky zahrnu společně ostatními do rozšíření metodiky MMSP v oblasti analýzy a návrhu testování.

5.1 Obecně o certifikacích ISTQB

Jak jsem již zmínil v úvodu kapitoly ISTQB (International Software Testing Qualifications Board) je mezinárodní sdružení testerů. Toto sdružení vzniklo v roce 2002 ve Skotsku, v Edinburgu (ISTQB, 2015) za účelem sdružovat mezinárodní skupinu IT specialistů v oblasti testování, která by měla váhu a světové uznání. ISTQB se postupně rozrostla po celém světě a má svou pobočku u nás, která reprezentuje Českou a Slovenskou republiku (CaSTB, 2013).

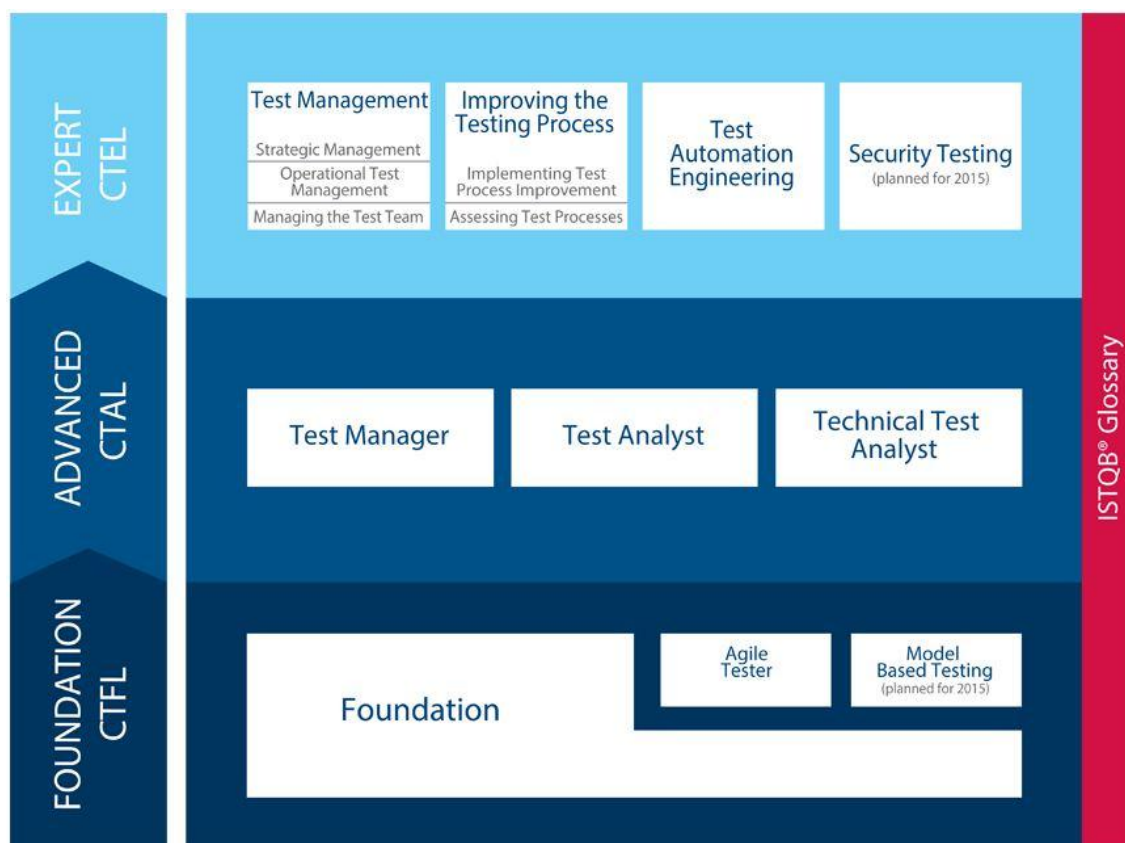
Tato organizace zajišťuje mezinárodně uznávané certifikace v oblasti testování. Tyto certifikace mají vlastní pečlivě zpracované směrnice a jednotný formát zkoušek, které mají zajistit garanci kvality nositelů certifikace ať už je udělena v jakékoli zemi.

Certifikace ISTQB mají vícevrstvou strukturu (Obrázek), která odděluje základní, pokročilou a expertní úroveň. Certifikace ISTQB vycházejí z předpokladu využívání agilních metodik. Základní úroveň je pro všechny, kdo se zabývají testováním software na obecné úrovni. Požadavky na zvládnutí základní úroveň certifikace jsou povědomí o základech testování. Patří sem testovací principy, znalost testovacího procesu, povědomí o životním cyklu vývoje softwaru, typech testů, statistických metodách, ovládání testovacích nástrojů, nebo testovacích technikách. Tato certifikace je doporučena primárně pro roli Tester.

Pro druhou prostřední úroveň je nutné mít prokazatelnou praxi v oblasti testování alespoň 5 let. Tvoří ji tři typy certifikace podle zaměření. Jsou to Test

manažer, Test analytik a Technický test analytik. Právě poslední dvě jmenované zahrnují činnosti z oblasti Analýzy a návrhu testování, proto je rozeberu podrobněji a zvláště v samostatných kapitolách. Test manažer je pozice řídící. Materiály ISTQB (ISTQB, 2015) definují požadavky na tuto roli velmi podrobně. Zahrnují sem znalosti z oblastí plánování a monitoringu testování, hodnocení požadavků, testovací dokumentace, kontroly a auditu, metriky, životní cyklus projektu, zlepšování testovacího procesu, používání nástrojů, reporting a i sociální dovednosti. Oblasti v druhé úrovni certifikací se mezi jednotlivými rolemi mírně prolínají. K tomu ale dochází i v běžné praxi.

Poslední úroveň je expertní, která rozšiřuje pokročilou úroveň. Expertní certifikace řeší problematiku vybrané oblasti více do hloubky a prakticky. Tyto certifikace jsou zaměřeny především na přínos v obchodní oblasti. Oblasti, ve kterých je možné se certifikovat, jsou Management testování, Zlepšování procesu testování a Automatizace testování. Do budoucna se připravuje certifikace v oblasti testování bezpečnosti (Obrázek).



Obrázek 8: Struktura certifikací ISTQB (ISTQB, 2015)

5.2 Role Test Analytik dle ISTQB

Role test analytika zastává podle ISTQB (McKay a kol., 2012) nejvíce analytických činností v oblasti testování. V této kapitole provedu jejich podrobný rozbor. Činnosti test analytika se v některých oblastech mírně prolínají s rolí Test manažera a Technického test analytika. Není ani vyloučeno, že jedna osoba zastává více rolí. ISTQB rozděluje (McKay a kol., 2012) úlohy test analytika do následujících oblastí:

- Plánování, monitorování a kontrola
- Analýza testování
- Návrh testování
- Realizace
- Provedení testů
- Vyhodnocení výstupních kritérií a reportování
- Ukončení testování

Plánování testování se vyskytuje na samém počátku projektu ještě před samotným zahájením provedením testů. Tato testerská činnost má za cíl identifikaci zdrojů a naplánování všech činností, které se budou provádět během zbytku testovacího procesu. Plánování testování vychází z strategie testování. Během plánování je důležité zaměřit se nejen na použití funkčního testování, ale zvážit i další možné typy testů, aby mohlo být odhaleno co nejvíce chyb. Výběr typů testů se zahrnuje do plánu testování. Tvorbu plánu testování je lepší konzultovat s Test manažerem. Součástí této činnosti je i naplánování testování podle konfigurace prostředí. Je dost možné, že se testovaný softwarový produkt bude chovat jinak v různých hardwarových a softwarových konfiguracích. Je také důležité naplánovat časové kapacity na testování dokumentace. Testování dokumentace je velmi zdlouhavé, nedá se zautomatizovat a je vyhotovována zpravidla na konci projektu, kdy na její testování nezbývá čas. Plánování testování by mělo zahrnovat otestování instalačních postupů. Chyba při zprovoznění softwarového produktu je mnohem kritičtější, než například nefungující funkcionality. Vše je třeba zaznamenat do Plánu testování tak, aby byl v souladu s životním cyklem projektu (McKay a kol., 2012).

Monitoring a kontrola testování je oblastí, za kterou je odpovědný Test manažer. Test analytik však přispívá k měření, které zajišťuje relevantní vstupy pro tuto

činnost. Test analytik naplánováním testovacích činností ovlivňuje množinu kvalitativně měřitelných údajů. Vše co Test analytik naplánuje (počet testovacích scénářů, počet testovacích sad, typy testů, procento pokrytí testy, atd.), má vliv na sestavování metrik a výsledný reporting hodnocení kvality. Čím lépe Test analytik naplánuje testování, tím lépe celkové výsledky odpovídají realitě.

Analýza testování dle ISTQB (McKay a kol., 2012) vychází u předpokladu, že je již definován rozsah testování. V rámci Analýzy testování Test analytik analyzuje bázi testování a identifikuje testovací podmínky (viz kapitola 6.2.2). Pro efektivní vypracování Analýzy testování musejí být splněny následující kritéria:

- Existuje dokument, nebo dokumenty popisující testovaný objekt (softwarový produkt), který slouží jako testovací báze. V praxi to bývá například dokument Funkční specifikace, požadavky, architektonický, nebo datový model.
- Výše zmíněné dokumenty byly schváleny (zadavatelem nebo vedením projektu).
- Jsou vytvořeny odpovídající podmínky pro uskutečnění testování (testovací tým, dostatečný rozpočet a časové kapacity na testování, atd.)

Testovací podmínky jsou definovány na základě testovací báze a cílů testování. Může však nastat situace, že všechna kritéria nejsou splněna, nebo jsou splněna pouze částečně. Jde o situaci, kdy je například nekompletní dokumentace, je zastaralá, nebyly do ní zapracovány změnové požadavky, anebo dokonce úplně chybí. Může k tomu dojít například při plánování sprintu (viz kapitola 4.3). Testovací podmínky se promítají do toho, co se bude testovat, testovacích technik, strategie testování, nebo plánu. Proto v případě nekompletní dokumentace se doporučuje aktuální stav konzultovat přímo s odpovědnými rolemi (Analytik, Vývojář).

V případě, že kritéria pro Analýzu testování nejsou kompletně splněna, stále existují standartní postupy pro definici Testovacích podmínek. Obvykle se definují Testovací podmínky na různých úrovních detailu. Nejprve podmínky na vysoké úrovni, které stanovují obecné cíle (například spuštění, přihlášení do aplikace). Poté se jde postupně na čím dál nižší úroveň, přes stanovení základních funkcionalit (ukládání dat do databáze) až po konkrétní vlastnosti systému (například vlastnosti textového pole). Tímto hierarchickým přístupem k definování Testovacích podmínek lze dosáhnout

pokrytí testy bez splnění základních kritérií pro Analýzu testování. Nedostatky by však měly být zmíněny v Analýze rizik testování.

Návrh testování dle ISTQB (McKay a kol., 2012) se musí držet v rozsahu Plánu testování, které bylo specifikováno v úvodu projektu. Test analytik navrhuje v této fázi testovací případy, které budou realizovány později. Proces Návrh testovacích případů spočívá v určení testované oblasti, nebo úrovně, pro kterou budou testovací případy vytvářeny. Určí vhodné techniky a postupy, které zaručí požadované pokrytí testy. Nakonec vytvoří Testovací případy, které splňují všechny Testovací podmínky.

Během vytváření testovacích případů je dobré mít na paměti, že některé věci je lépe zařadit do Testovacích podmínek, než pro ně vytvářet samostatné testy. Každý testovací případ musí mít jasně vyznačena kritéria pro splnění, či nesplnění testu. Testovací případy by měly být také navrženy tak, aby byly srozumitelné i pro ostatní lidi, kteří se budou testování účastnit, ne jen pro autora. Každý by měl být chopen porozumět cíli a významu testu, včetně ostatních zúčastněných stran (zadavatelé, vývojáři, popřípadě i auditoři). Vytvořené testovací případy by měli zaručit co nejširší pokrytí testovaného softwarového produktu.

Jedním z hlavních úkolů test analytika je určit správný typ testů vždy pro danou část systému. Jednotlivé Testovací případy by měly být co nejvíce konkrétní. To by mělo zaručit, že provedení testů různými osobami bude mít vždy stejný výsledek. Přesnost a podrobnost testovacího případu je výhodná v případě, že testy vykonávají méně zkušení pracovníci. Na druhou stranu přesnost testovacího případu omezuje vynalézavost zkušených pracovníků. Míra podrobnosti specifikace Testovacího případu tvoří míru zastupitelnosti v procesu testování.

Testovací případy jsou tvořeny z posloupně jdoucích kroků. Tento postup umožňuje opakovatelnost provádění testů. Každý testovací scénář by měl podle návrhu ISTQB (McKay a kol., 2012) obsahovat:

- Název – výstižný název často doplněný o unikátní ID.
- Testovací předpoklady – podmínky, které musejí být splněny před spuštěním testovacího případu.
- Požadavky na testovací data – definice požadavků na vstupní data.
- Očekávaný výsledek – popis jak se softwarový produkt zachová.
- Požadavky po provedení testu – dopad na data, stav systému, atd.

Realizace je vytvoření testů podle Návrhu testování. Zahrnuje třídění a organizaci testovacích scénářů, automatizaci testů, finální přípravu testovacích dat a testovacího prostředí. Zahrnuje rovněž rozdělení zdrojů. Kontroluje vstupní a výstupní kritéria vlastní realizace testovacích scénářů. Vyhotovené testovací scénáře musejí být schváleny. Realizace není vlastní spuštění testů, jde o zajištění splnění všech potřebných kritérií pro jejich spuštění. Realizace testování zajišťuje nastavení a kontrolu testovacího prostředí ještě před spuštěním testovacích scénářů. Realizace zajišťuje hladký chod provádění testů, v jeho průběhu zajišťuje podporu a reaguje na případné neočekávané změny. Dojde-li během provádění testů ke změnám, má na starosti posoudit dopad změn na testy, které byly spuštěny.

Provedení testů je vlastní spuštění navrhnutých a vytvořených testovacích scénářů. Provedení testů začíná dodáním testovaného objektu (ve většině případů specifická verze softwarového produktu) a splněním vstupních kritérií pro testování. Vše by mělo probíhat v souladu s Plánem testování. Hlavní rolí, která se účastní provádění testů, je Tester. Test analytik se provádění testů účastní také, ale měl by mu být vyhrazen i čas na pokrytí testy nově objevených chování systému, či opravení Analýzy testování a Testovacích scénářů při zjištění různých odchylek během testování.

Hlavní úlohou provádění testů je porovnávání skutečných výsledků testování s očekávanými výsledky jednotlivých testovacích scénářů. V případě, že dojde k rozporu mezi skutečností a očekávaným výsledkem jedná se o incident. V případě incidentu nejde vždy bezpodmínečně nutně o chybu. Incidentů je mnoho druhů, může se jednat o část systému, která nebyla ještě implementována, nebo část systému, která není pokrytá analýzou. U každého incidentu se musejí pečlivě prověřit příčiny a zaznamenat do bug-tracking systému, ten je spravován zpravidla za pomoci nějakého nástroje (viz kapitola 7.5) Může se však stát, že chyba je v testovací dokumentaci (Testovací případy). V takovém případě je nutné, aby byla opravena a test spuštěn znovu. Může nastat i situace, že test projde několikrát za sebou v pořádku, ale chování je nesprávné, protože pro danou funkcionalitu byl zadán změnový požadavek, který nebyl zatím zadán ani do dokumentace a nebyl ani implementován.

Průběh provádění testů musí být pečlivě zaznamenáván. Je nutné evidovat, které testy mají být spuštěny, které byly spuštěny a které ještě nebyly. Absence evidence průběhu testování vede ke snižování efektivnosti testování, zdvojování práce a

ztrátě času. Evidence průběhu testů by se měla provádět pro každý testovací scénář a každou verzi testovaného objektu zvlášť.

Záznam z průběhu testování a výsledky testů slouží jako podpora pro řízení testování, vstupní data pro měření testování, reportingu o průběhu testování, hodnocení kvality testovaného objektu a podklady pro zlepšování procesu testování. Pro snazší evidenci Provádění testů je v současné době dostupná řada sofistikovaných softwarových nástrojů (viz kapitola 7.5).

Během provádění testů je třeba brát zřetel na jakékoli zvláštnosti a odchylky. Ty se mohou na první pohled zdát nepodstatné, ale ve skutečnosti jsou spolehlivými indiciemi, že v dané části systému se nalézá chyba. Během testování je třeba testovat i negativní testovací případy. Jinými slovy je třeba otestovat nejen to, jak se má testovaný objekt chovat, ale i že se nechová tak, jak se chovat nemá. Tento fakt by se měl odrážet již v Analýze testování. Během provádění testování je vhodné dělat si poznámky a vstřebávat poznatky, které dají užítkovat v dalším kole testování. S každou iterací by mělo současně stoupat know-how zaměstnanců a mělo by docházet ke zlepšování procesu testování.

Pro provádění testů platí ještě pravidlo, že je dobré hledat chyby testovaného objektu dříve, než provádět regresní testování. Vlastní regresní testování je zdoluhavé a nemá šanci najít velkou část závad. Regresní testy ověřují, že se nerozbily funkce systému od dodání posledního testovaného objektu. Neznamená to však, že jsou zbytečné a nemusí se provádět. Regresní testy je vhodné zautomatizovat.

Vyhodnocení výstupních kritérií a reportování, jak už bylo uvedeno výše, slouží k monitorování testovacího procesu. Předpokladem je správné definování požadavků a pokrytí testů pro sběr informací pro reporting a zlepšování procesů. Vyhodnocování a reportování slouží zároveň jako podpora řízení projektu a testování. Za správnost nashromážděných informací je odpovědný test analytik.

ISTQB uvádí (McKay a kol., 2012) jako poslední úlohu testovacího procesu činnost **ukončení testování**. Po dokončení provádění testů a jejich vyhodnocení Test analytik zpracovává klíčové výstupy. Zaznamenávání incidentů a pověření odpovědných osob se v praxi provádí během vlastního provádění testů. Na základě získaných poznatků rozšiřuje stávající pokrytí testy, dokumentuje změny, provádí úpravu prostředí, nebo spouští automatické testy.

5.3 Role Technický test analytik dle ISTQB

Další rolí provádějící analytické činnosti v oblasti testování je role Technického test analytika. Zdroje ISTQB uvádějí (Bath a kol., 2012), že tato role je úzce specializována na hodnocení rizik v oblasti testování a testování za pomoci nejrůznějších testovacích technik, obecně známé jako white-box nebo code-based testování. Od této role se očekává vysoká míra technických znalostí pro plnění následujících úloh v oblasti testování:

- Identifikace rizik
- Hodnocení rizik
- Zmírnění rizik
- White-box testování

Identifikace rizik v oblasti testování má za úkol odhalit co největší počet skutečností, které by mohly ohrozit realizaci a průběh testování. Na osobu, která je za identifikaci rizik odpovědná, jsou kladeny vysoké nároky na úroveň technických znalostí. Identifikace rizik se provádí na základě znalostí testovaného objektu, analýzy získaných zkušeností z předešlých testování odborných rozhovorů s odpovědnými osobami, nebo třeba informací, které vyplynou z pravidelných meetingů.

Možná rizika, která mohou nastat, jsou velice specifická a odvíjejí se od druhu a náročnosti projektu. Příkladem takových rizik je například nedostatek zdrojů (ať už časových nebo lidských), bezpečnostní rizika, nebo výkonnostní rizika (při nahrání velkého množství dat může být systém pomalý). Všechna nalezená rizika musejí být pečlivě zaznamenána. Většinou je výstupem dokument Analýza rizik v oblasti testování.

Po dokončení hledání a identifikace přichází na řadu **hodnocení rizik**. Úkolem této činnosti je ohodnotit míru, pravděpodobnost a dopady rizika. Technický test analytik stanoví míru nebezpečí každého rizika, přičemž musí uvažovat plno dalších faktorů. Faktory, které hodnocení rizik ovlivňují, jsou například používané softwarové nástroje, použité technologie, dostupnost zdrojů, případné komunikační problémy, nebo časová náročnost.

Zmírnění rizik jsou nápravná opatření, která mají za úkol omezit nalezená rizika. Rizika ovlivňují průběh testování. Zvýší se priorita testovacích scénářů

v rizikových oblastech. Zmírnění rizik se promítne do strategie testování a plánu testování (Bath a kol., 2012).

White-box testování je návrh a provádění testů na vysoké úrovni za pomoci sofistikovaných testovacích technik. Tento druh testování vyžaduje pokročilé analytické dovednosti a technické znalosti, neboť k návrhu testování využívá zdrojový kód, data, nebo znalosti o architektuře testovaného objektu. Každá testovací technika umožňuje tvorbu různých odvozených testovacích scénářů, které nejsou patrné z testovací báze. Spočívá v testování na velmi nízké úrovni testovaného objektu. ISTQB mezi techniky řadí (Bath a kol., 2012) například:

- Testování pomocí podmínek
- Rozhodovací parametry
- Podmiňovací parametry
- Víceúrovňové rozhodování
- Testování podle Xpath
- Testování pomocí API

Použití testovacích technik výrazně zvyšuje pokrytí testy testovaného objektu. Rozhodovací tabulky napomáhají například tvorbě negativních scénářů. Použití technik obecně zvyšuje efektivitu celého testovacího procesu. Mezi další hojně používané testovací techniky se řadí Testování tříd ekvivalence, Analýza okrajových hodnot, nebo Kombinatorická analýza (Page, Johnston a Rollison, 2009).

5.4 Shrnutí

ISTQB je nejrozšířenější a nejznámější sdružení testerů softwaru na světě. Na základě zkoušek uděluje mezinárodně uznávané certifikace v různých úrovních a oblastech testování. Je možné získat certifikace ze základů testování, přes jednotlivé role testovacího týmu až po expertní oblasti s praxí několika let. Znalosti potřebné pro certifikace vycházejí z agilního postoje k projektům. ISTQB má svou pobočku i pro region České a Slovenské republiky.

ISTQB se do hloubky zabývá i disciplínou Analýza a návrh testování. ISTQB rozebírá Analýzu a návrh testování z pohledů rolí na projektu. Role, které provádějí nejvíce analytických činností v oblasti testování, jsou Test analytik a Technický test analytik. Jednotlivé role se však mírně prolínají.

Analytické činnosti v oblasti testování jsou obsaženy v činnostech Plánování testování, Analýza testování, Návrh testování, Vyhodnocení výstupních kritérií, Ukončení testování, Identifikace rizik, hodnocení rizik a testování pomocí sofistikovaných testovacích technik. Během plánování testování dochází k identifikaci cílů a zdrojů testování. Na jejich základě jsou naplánovány všechny činnosti testovacího procesu. V rámci Analýzy testování se identifikují testovací podmínky a analyzuje se báze testování. Následuje Návrh testování, který vychází ze všech doposud provedených činností. Dochází k navržení typů testů a testovacích případů, které jsou vytvořeny v rámci činnosti Realizace. Po provedení testů jsou výsledky analyzovány. Poznatky jsou zahrnuty do znalostní báze. Identifikace rizik odhaluje skutečnosti, které by mohli ohrožit proces testování. Rizika mají vliv na průběh testování a tvorbu plánu testování. Analytické dovednosti jsou nezbytné i pro testování pomocí testovacích technik.

6 Analýza a návrh testování z pohledu normy ČSN ISO/IEC/IEEE 29119

V této kapitole se zaměřuji na analýzu mezinárodní řady norem ISO/IEC/IEEE 29119 a provádím podrobnou analýzu, jak se tyto normy staví k otázce Analýzy a návrhu testování. V řadě norem ISO/IEC/IEEE 29119 se uvádí (ISO/IEC/IEEE 29119-1, 2013), že má sloužit k definování mezinárodních norem v oblasti testování softwaru. S mohutným rozvojem informačních technologií se software dostává na mnoho platform. Zároveň vzniká velké množství druhů softwaru. Normy ISO/IEC/IEEE 29119 mají udržet jednotný rámec v oboru testování softwaru a usnadnit tak jejich použitím testování v jakémkoli typu organizace. Normy sdružují současné obecné koncepty testování, postupy, plánování, použití automatizace, zapojení do životního cyklu vývoje softwaru, nebo cílů testování softwaru. Normy v sobě neobsahují roli test analytika, ale částečně zahrnují prvky disciplíny Analýzy a návrhu testování. Cílem této kapitoly je nalézt tyto prvky pro pozdější zařazení do metodiky MMSP.

6.1 Norma ČSN ISO/IEC/IEEE 29119

ČSN ISO/IEC/IEEE 29119 je česká technická norma (ČSN ISO/IEC/IEEE 29119-1, 2014), která začala vznikat koncem roku 2014 lokalizací mezinárodní normy ISO/IEC/IEEE 29119 (ISO/IEC/IEEE 29119-1, 2013). Jedná se o normu spadající do oblasti „Softwarové a systémové inženýrství“ a konkrétně popisuje „Testování softwaru“. Na překladu její české lokalizace se v současné době stále ještě pracuje a nebyla zatím plně zavedena. Za překlad má odpovědnost Úřad pro technickou normalizaci, metrologii a státní zkušebnictví, který na překladu spolupracuje s předními pedagogy na Vysoké škole ekonomické v Praze, konkrétně pak s doc. Ing. Alenou Buchalceovou, Ph.D., která je zároveň i vedoucí mé diplomové práce. Český překlad normy má plnohodnotný status jako oficiální anglická verze.

Původní anglická lokalizace normy ISO/IEC/IEEE 29119 uvádí (ISO/IEC/IEEE 29119-1, 2013), že se norma dělí celkem na pět částí:

- ISO/IEC/IEEE 29119-1: Concepts & Definitions (Koncepty a definice)
- ISO/IEC/IEEE 29119-2: Test Processes (Testovací procesy)
- ISO/IEC/IEEE 29119-3: Test Documentation (Dokumentace testování)

- ISO/IEC/IEEE 29119-4: Test techniques (Techniky testování)
- ISO/IEC/IEEE 29119-5: Keyword Driven Testing (Testování pomocí klíčových slov)

Kromě poslední páté části, která má být vydána do konce roku 2015, jsou všechny čtyři části dostupné v anglickém jazyce. V rámci české lokalizace normy ČSN ISO/IEC/IEEE 29119 jsou zatím dostupné pouze první a druhá část.

Podle informací zveřejněných na webu mezinárodní společnosti pro standardy v oblasti testování (Reid, 2014) zavedení řady norem ISO/IEC/IEEE 29119 nahrazuje následující existující normy a standardy:

- IEEE 829 Test Documentation (Dokumentace testování)
- IEEE 1008 Unit Testing (nepřekládá se)
- BS 7925-1 Vocabulary of Terms in Software Testing (Slovník pojmů v testování softwaru)
- BS 7925-2 Software Component Testing Standard (Standard pro testování softwarových komponent)

6.2 Definice pojmů v oblasti analýzy a návrhu testování podle ČSN ISO/IEC/IEEE 29119-1

Norma ČSN ISO/IEC/IEEE 29119-1 uvádí definice velkého množství pojmů v oblasti testování. V této kapitole se však zaměřím pouze na definice, které souvisí bezprostředně s disciplínou Analýza a návrh testování. Některé z nich později použiji při rozšiřování metodiky MMSP. Definice normy ČSN ISO/IEC/IEEE 29119-1 jsem se rozhodl ve své diplomové práci využít, protože se mi zdálo zbytečné snažit se definovat znovu něco, co již bylo definováno odborníky.

6.2.1 Definice pojmů rolí

Je velmi zajímavé, že norma ČSN ISO/IEC/IEEE 29119-1 neuvádí vedle role Test manažera a Testera (myšleno manuální tester) dnes hojně propagovanou roli Test analytika (nebo též Analytik testování). Norma místo této role obsahuje definici role Stratéga testování. Stratég testování má podle normy na starosti: „*zajištění a ustanovení shody s procesem testování v rámci celé organizace*“ (ČSN ISO/IEC/IEEE 29119-1,

2014, str. 49). Tato role se do jisté míry roli Test analytika velmi podobá. Specialista zastávající tuto roli má také na starosti tvorbu strategie testování, plánu testování a analýzu podmínek pro testování, avšak na úrovni celé organizace. Domnívám se, že lze tuto roli chápat i jinak. Mohlo by jít o roli s řídicími pravomocemi, která má na starosti organizaci a určování priorit testování. Ve své podstatě by stratég testování mohl být chápán i jako role Test manažera z hlediska celé organizace. Stratég testování není v normě ČSN ISO/IEC/IEEE 29119-1 nijak blíže specifikován.

Roli Analytika testování norma uvádí pouze nepřímo. Norma vychází z předpokladu, že míra podílu na autorství testovaného objektu ovlivňuje přímo úměrně objektivitu testování. Zjednodušeněji řečeno testuje-li autor svůj vlastní produkt, je pro něj mnohem složitější nalézt chyby, než pro nezávislého Testera. Domnívám se, že stejně je tomu i u autorství testovacích scénářů a jejich provádění. Norma uvádí (ČSN ISO/IEC/IEEE 29119-1, 2014), že je vhodné, aby návrh a provádění testování bylo prováděno jinou osobou, než je autor testovaného objektu. V rámci zavedení disciplíny Analýza a návrh testování doporučuji, aby návrh testování (včetně analýzy testování) a provádění testování bylo vykonáváno jinou nezávislou osobou v rámci produkční části organizace.

6.2.2 Definice pojmů s ohledem na disciplíny metodiky MMSP

Norma ČSN ISO/IEC/IEEE 29119-1 uvádí rozsáhlý počet hojně používaných definic termínů v disciplíně testování. Oddělím-li Analýzu a návrh testování do samostatné disciplíny, spadají do ní podle mezinárodní normy tyto definice:

Testovaný objekt (test object), nebo také testovaná položka je „*výsledek práce, který je předmětem testování*“ (ČSN ISO/IEC/IEEE 29119-1, 2014). V praxi se jedná o softwarový produkt, kterým může být informační systém, nebo nejrůznější softwarové programy a aplikace. Disciplína Analýza a návrh testování tento software zkoumá kvůli vytvoření podkladů pro testování a co nejlepší ohodnocení jeho kvality.

Produktové riziko (product risk) je „*riziko, že může být produkt vadný v některých specifických aspektech své funkce, kvality nebo struktury*“ (ČSN ISO/IEC/IEEE 29119-1, 2014). Úkolem Analýzy a návrhu testování bude tyto rizika nalézat z pohledu ohrožení testování. Analýza a návrh testování se při analýze produktových rizik soustředí primárně na odhalení rizikových oblastí testovaného objektu, kde se může nacházet zvýšená koncentrace chyb.

Testovací podmínka (test condition) je „*testovatelný aspekt komponenty nebo systému, jako například funkce, transakce, vlastnost, atribut kvality nebo strukturální element identifikovaný jako báze testování*“ (ČSN ISO/IEC/IEEE 29119-1, 2014). Testovací podmínky definují okolnosti a podstatu testovacího případu. Testovací podmínky jsou identifikovány analýzou testování a jsou zahrnuty do Návrhu testování.

Kritéria úspěchu/selhání (pass/fail criteria) jsou „*rozhodovací pravidla používaná k určení toho, zda testovaná položka nebo její funkce prošla nebo neprošla testem*“ (ČSN ISO/IEC/IEEE 29119-1, 2014). Tyto kritéria jsou identifikována během návrhu a realizace testů ještě před samotným vykonáváním testování. Kritéria úspěchu/selhání jsou součástí každého testovacího scénáře. K jejich identifikaci dochází podrobnou analýzou Báze testování.

Báze testování (test basis) je „*soubor znalostí používaný jako základ návrhu testů a testovacích případů. Báze testování může mít formu dokumentace, jako například specifikace požadavků, specifikace návrhu nebo specifikace modulů, ale může mít také formu nezdokumentovaného porozumění požadovanému chování*“ (ČSN ISO/IEC/IEEE 29119-1, 2014). Jinými slovy báze testování je kompletní souhrn informačních zdrojů, které slouží jako vstupy pro další úlohy disciplíny Analýza a návrh testování. Z Báze testování se identifikuje požadované chování testovaného objektu, Testovací podmínky a v neposlední řadě i kritéria úspěchu/selhání. Bázi testování je nutné vždy podrobně analyzovat. Vynechání tohoto kroku by mělo velmi negativní dopad na výsledky kontroly kvality testovaného objektu. V praxi bohužel častěji dochází k opačnému problému. Velmi často se setkávám s tím, že Báze testování se není kompletní, anebo v některých případech úplně chybí. Tento problém, který je způsoben nedostatkem v provádění disciplíny Analýzy, značně komplikuje práci celému testovacímu týmu. Rovněž může nastat situace, že se Báze testování v průběhu projektu dynamicky mění. V tomto případě je nutné v rámci Analýzy a návrhu testování neustále aktualizovat analýzu testování a doplňovat a opravovat návrhy a specifikace testovacích případů, aby mohli testéři plynule pracovat.

Testování založené na specifikaci (specification-based testing) je „*testování, jehož základ tvoří externí vstupy a výstupy testované položky, běžně založené na specifikaci, spíše než na implementaci ve zdrojovém kódu nebo spustitelném softwaru*“ (ČSN ISO/IEC/IEEE 29119-1, 2014). Jedná se v současné době o nejrozšířenější způsob testování. Někdy se mu říká také black-box testování. Disciplína Analýza a

návrh testování zajišťuje vstupy pro toto testování ve formě realizovaných návrhů testovacích scénářů, které obsahují kritéria úspěchu/selhání. Tyto vstupy jsou vypracovány na základě analýzy Báze testování a Testovacích podmínek.

6.2.3 Definice pojmů technik

Norma ČSN ISO/IEC/IEEE 29119-1 počítá s využíváním sofistikovaných technik v procesu testování. Tyto techniky pomáhají usnadňovat testování díky ověřeným postupům. Většina pojmů uváděná mezinárodní normou pro testování týkající se testovacích technik, jsou přesně ty samé techniky, které jsou uváděny i v knize „*Jak testuje software Microsoft*“ od manažerů testování z Microsoftu (Page, Johnston a Rollison, 2009). Jedná se o přehled nejdůležitějších testovacích technik (více viz kapitola 7.3). V rámci této kapitoly vysvětlím zatím pouze nejdůležitější pojmy, které uvádí norma ČSN ISO/IEC/IEEE 29119-1 a vztahují se k technikám testování využívaných v disciplíně Analýza a návrh testování. Norma ČSN ISO/IEC/IEEE 29119-1 uvádí z této oblasti následující pojmy:

Testování založené na rizicích (risk-based testing) je „*testování, při kterém je řízení, výběr, nastavení priorit a použití testovacích aktivit a prostředků vědomě založeno na odpovídajících typech a úrovních analyzovaných rizik*“ (ČSN ISO/IEC/IEEE 29119-1, 2014). Jde o techniku Analýzy testování, která je zásadním způsobem ovlivněna Analýzou produktových rizik. Analýza a návrh testování se v tomto případě primárně soustředí na oblasti systému, kde je nejvyšší pravděpodobnost snížené kvality testovaného objektu.

Další technikou testování, kterou norma ČSN ISO/IEC/IEEE 29119-1 zmiňuje je testování tříd ekvivalence (viz kapitola 7.3.2). **Dělení na třídy ekvivalence** (equivalence partitioning) je „*technika návrhu testování, u které jsou testovací případy navrženy tak, aby využívaly tříd ekvivalence použitím jednoho nebo více reprezentativních zástupců každé třídy*“ (ČSN ISO/IEC/IEEE 29119-1, 2014). Rozdělení na třídy ekvivalence je Technika návrhu testování. Technika spočívá v rozdělení testovaného objektu na jednotlivé třídy ekvivalence. Množství tříd je závislé na zvolené granularitě (míra podrobnosti) podmnožin. Analýza a návrh testování se snaží o jejich co nejlepší identifikaci a pokrytí.

Technika návrhu testování (test design technique) jsou „*aktivity, koncepty, procesy a vzory používané k vytváření modelu testů, který se použije k určení*

testovacích podmínek testované položky, odvození odpovídajících položek pokrytí testů a následnému odvození nebo výběru testovacích případů“ (ČSN ISO/IEC/IEEE 29119-1, 2014).

Třída ekvivalence (equivalence partition) je „*podmnožina rozsahu hodnot jedné nebo více proměnných, uvnitř testované položky nebo na jejích rozhraních taková, že se u všech hodnot v oddílu dá odůvodněně očekávat, že s nimi bude testovaná položka zacházet obdobným způsobem (tj. mohou být testovanou položkou považovány za „ekvivalentní“*“ (ČSN ISO/IEC/IEEE 29119-1, 2014). Otestování reprezentativního vzorku z třídy ekvivalence předpokládá stejný výsledek provedení testování i u všech ostatních zástupců třídy.

Pokrytí tříd ekvivalence (equivalence partition coverage) je „*část identifikovaných tříd ekvivalence testované položky, které jsou pokryty sadou testů“*“ (ČSN ISO/IEC/IEEE 29119-1, 2014). Analýza a návrh testování je schopna obsáhnout širokého pokrytí Návrhu testování na mnohem užším vzorku prvků testovaného objektu.

6.2.4 Definice pojmů pracovních produktů

Norma ČSN ISO/IEC/IEEE 29119-1 popisuje mnoho definic výstupů testovacího procesu. Metodika MMSP označuje tyto výstupy jako takzvané pracovní produkty. Pracovní produkty jsou hmatatelné výstupy jednotlivých disciplín v rámci procesu vývoje softwarového produktu a mají většinou podobu dokumentace. V této podkapitole se zaměřím pouze na ty, které souvisí s disciplínou Analýza a návrh testování.

Mezinárodní normy ISO kladou na testování velký důraz. Jejich česká lokalizace, norma ČSN ISO/IEC/IEEE 29119-1 (ČSN ISO/IEC/IEEE 29119-1, 2014) popisuje potřebu testování kvality hned z několika aspektů. Testování softwarového produktu je vyžadováno lidmi na řídicích pozicích v organizaci. Znalost stavu jim pomáhá k dalšímu rozhodování. Obecně nelze předpokládat, že výsledek je vždy bezchybný, chyby se stávají a testovaný objekt se musí vyladit, aby dělal to, co se od něj očekává. Testování by mělo být prováděno během celého životního cyklu, aby se zajistila co nejvyšší kvalita. Analytické činnosti v oblasti testování tomu významnou měrou napomáhají.

Chyby obsažené v testovaném objektu mohou mít celou řadu příčin. Může jít o selhání lidského faktoru, změnu požadavků, nebo změnu technologií. Norma ČSN ISO/IEC/IEEE 29119-1 zahrnuje myšlenku, že současný vývoj softwaru probíhá v agilním prostředí. Díky agilnímu vývoji softwaru je nutné provádět kombinaci dynamického a statického testování. Dynamické testování není vázáno podrobnými testovacími kroky a dává testerovi větší volnost. Dynamické testování vyžaduje testera s pokročilými znalostmi technologií a testovaného objektu. Provádí se například pro potřeby testování na základě Analýzy rizik. Statické testování podle přesných postupů doplňuje celý testovací proces.

Všechny činnosti testování a tudíž i činnosti Analýzy a návrhu testování podléhají politice a strategii testování celé organizace. **Politika testování organizace** (Organizational Test Policy) je „dokument na exekutivní úrovni, který popisuje účel, cíle a celkový rozsah testování v rámci organizace a vysvětluje, proč se testování provádí a čeho se snaží dosáhnout“ (ČSN ISO/IEC/IEEE 29119-1, 2014). **Strategie testování organizace** (Organizational Test Strategy) je „dokument, který vyjadřuje obecné požadavky na testování prováděné v rámci všech projektů dané organizace, obsahující konkrétní informace o tom, jak by testování mělo probíhat“ (ČSN ISO/IEC/IEEE 29119-1, 2014). Obě dohromady tvoří jakýsi rámec, mezi jehož mantinely se provádí plánování, monitorování a řízení testovacího procesu v celé organizaci. Oba termíny lze však uplatnit i na úrovni konkrétního projektu. Disciplína Analýza a návrh testování, bude z hlediska testování zásadní, neboť zajišťuje pro všechny činnosti testování podklady. Většinu podkladů pro analýzu testování tvoří testovací báze.

Hlavní analytickou činností je na základě analýzy testovací báze, rizik, celkového rozpočtu na testování, časových kapacit, rozsahu projektu, metod, a v neposlední řadě technologického zázemí vytvořit plán testování. **Plán testování** (Test Plan) je „podrobný popis cílů testování, kterých má být dosaženo, a prostředků a časového plánu potřebného k jejich dosažení, organizovaných za účelem koordinování testovacích aktivit testované položky nebo sady testovaných položek“ (ČSN ISO/IEC/IEEE 29119-1, 2014). Plán testování zahrnuje Strategii testování projektu, politiku a strategii testování celé organizace. Plán se může rozdělit na více dílčích plánů, které pokrývají například jednotlivé etapy. Plán testování zahrnuje výběr testovacích technik a výběr typů návrhu testů. Měl by obsahovat rovněž zvážení

různých potřeb testování a požadavků na testovací prostředí. Plán testování je hlavním pracovním produktem analytických činností v oblasti testování.

Dalšími pracovními produkty Analýzy a návrhu testování jsou produkty týkající se Návrhu testování. Tyto produkty mají většinou podobu textového dokumentu. Návrh testování vychází z plánu testování a zahrnuje následující pracovní produkty:

- Specifikace návrhu testování – je shrnutím analýzy, které funkce a vlastnosti testovaného objektu mají být testovány. Dokument uvádí odpovídající testovací podmínky.
- Specifikace testování – je podřízena plánu testování a zahrnuje kompletní návrh testování, testovací případy a postupy pro otestování testovaného objektu.
- Specifikace testovacího případu – je dokumentace popisující analýzu konkrétního testovacího případu, nebo sady případů na jejich základě se vytvářejí testovací scénáře.

Norma ČSN ISO/IEC/IEEE 29119-1 rozděluje pracovní produkty Návrhu testování velmi podrobně. Konečným pracovním produktem Návrhu testování je testovací případ. **Testovací případ** (test case) je *„sada vstupních podmínek, vstupů (ve vhodných případech včetně akcí) a očekávaných výsledků testovacího případu, vytvořených k řízení provádění testované položky tak, aby vyhovělo cílům testování, včetně správné implementace, identifikace chyb, kontroly kvality a dalších hodnotných informací“* (ČSN ISO/IEC/IEEE 29119-1, 2014). Jedná se o popis konkrétního případu užití testovaného objektu. Testovací případ je popis konkrétních kroků vedoucích k otestování konkrétní funkce, nebo vlastnosti testovaného objektu. Testovací případy je třeba neplést si s testovacími scénáři. Testovací scénář je množina testovacích případů uspořádaná do logického celku pro otestování vybrané části testovaného softwarového produktu.

V praxi k vypracování kompletní dokumentace Návrhu testování z důvodu časové tísně většinou dochází jen zřídka. Testovací tým je většinou nucen návrh testování přeskočit a od analýzy testovací báze přejít rovnou k tvorbě Testovacích případů a vytvoření postupu testování. **Postup testování** (test procedure) je *„posloupnost testovacích případů uvedených v pořadí jejich provádění spolu se souvisejícími akcemi, které mohou být nutné k zajištění vstupních podmínek, a*

závěrečnými aktivitami následujícími po provedení testů“ (ČSN ISO/IEC/IEEE 29119-1, 2014).

Některé složitější testovací případy vyžadují jako vstup specifická testovací data. **Testovací data** (test data) jsou *„data vytvořená nebo vybraná tak, aby splňovala požadavky na vstupy pro provádění jednoho nebo více testovacích případů, které mohou být definovány v Plánu testování, testovacím případě nebo postupu testování“* (ČSN ISO/IEC/IEEE 29119-1, 2014). Testovací data jsou předem připravována v rámci analytických činností testování. Příkladem takové potřeby testovacích dat jako vstupu může být například potřeba otestování funkce exportu a importu souborů. Chceme-li ověřit správnost exportu dat ze softwarové aplikace, je třeba mít v aplikaci vytvořená data, která se mají exportovat. Pro správné otestování je třeba exportovat rozsáhlejší množství, nebo složitější strukturu dat, kterou nelze tvořit kvůli časové náročnosti během vlastního provádění testování, proto je vhodné data v rámci analýzy připravit předem. Ze stejného důvodu si musíme připravit i data v případě importu. Popis stavu připravenosti testovacích dat shrnuje dokument Zpráva o připravenosti testovacích dat.

Analytické činnosti v oblasti testování musejí podle normy ČSN ISO/IEC/IEEE 29119-1 zahrnovat i analýzu požadavků na testovací prostředí. Na základě analýzy testovací báze testovací tým formuluje požadavky na testovací prostředí, které musejí být splněny, aby bylo možné provádět testování. V praxi se může jednat o požadavky na běh, výkon, instalace nebo přístupy na testovací prostředí. Popis splnění všech požadavků na testovací prostředí shrnuje dokument Zpráva o připravenosti testovacího prostředí.

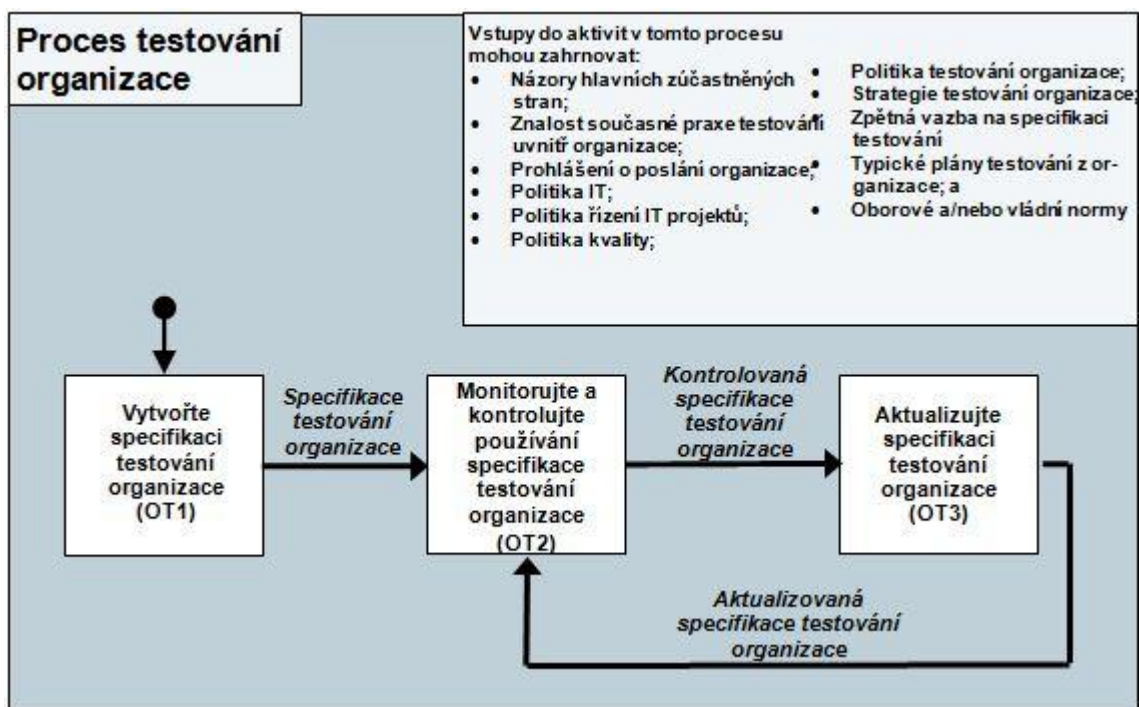
6.3 Analýza a návrh testování v testovacím procesu podle normy ČSN ISO/IEC/IEEE 29119-2

Norma ČSN ISO/IEC/IEEE 29119-2 popisuje soubor norem pro testování softwaru z hlediska řízení testování a dynamického testování. Norma se zaměřuje na procesy globálně z hlediska celé organizace. Znamená to, že popisuje řízení testování a řízení dokumentace testování obecnými směnicemi, které lze aplikovat na jednotlivé projekty. V této podkapitole se zaměřím na praktiky testování organizace, které se týkají analytických činností.

6.3.1 Úloha analytických činností v procesu testování organizace

Většina analytických činností v procesu testování organizace se podle normy ČSN ISO/IEC/IEEE 29119-2 nachází na začátku testovacího procesu. Ještě před zahájením samotného testovacího procesu je vytvořena dokumentace Politiky testování organizace a Strategie testování organizace. Ty jsou zahrnuty do procesu testování organizace a slouží k vývoji a řízení veškerých dokumentů testování.

Na obrázku níže (Obrázek 9) je vidět, že zahájení testovacího procesu začíná vytvořením specifikace testování organizace. Analýzou testovací báze, zápisů z meetingů a konzultace s příslušnými osobami se formulují požadavky na specifikaci testování. Specifikace testování musí být schválena osobami zastávajícími řídicí role. Po jejím vypracování je nutné seznámit se Specifikací testování všechny zúčastněné strany.



Obrázek 9: Proces testování organizace (ČSN ISO/IEC/IEEE 29119-2, 2014)

Během provádění testování, které probíhá většinou v iteracích, se monitoruje používání Specifikace testování, odhalují se její nedostatky a vypracovává se zpětná vazba. Hodnotí se její kvalita a efektivita jejího využívání. Specifikace testování musí být zpřístupněna všem příslušným osobám. Po dokončení iterace testovacího cyklu je dokončena zpětná vazba. Na základě zpětné vazby, empiricky nabytých zkušeností a nově přidáných požadavků je Specifikace testování aktualizována. Se zvyšujícím se

počtem iterací testovacího cyklu se zvyšuje její kvalita (ČSN ISO/IEC/IEEE 29119-2, 2014).

6.3.2 Proces plánování testování

Proces řízení testování může být použit na úrovni celého projektu, nebo i jeho dílčích částí. Podle normy ČSN ISO/IEC/IEEE 29119-2 se dělí na tři podprocesy:

- Plánování testování
- Monitorování a kontrola testování
- Dokončení testování

Řízení testování je vykonáváno IT specialistou, který zastává řídicí funkci. Plánování testování obsahuje však plno analytických činností, které spadají do disciplíny Analýzy a návrhu testování a mohou být vykonávány rolí Analytika testování.

Proces plánování testování slouží k vytvoření dokumentu Plánu testování. Proces plánování testování zobrazený na obrázku (Obrázek 10) zahrnuje posloupný seznam doporučených analytických aktivit, jejichž postupným provedením odpovídá osoba, nebo osoby dospějí k vytvoření záznamu úplného Plánu testování.

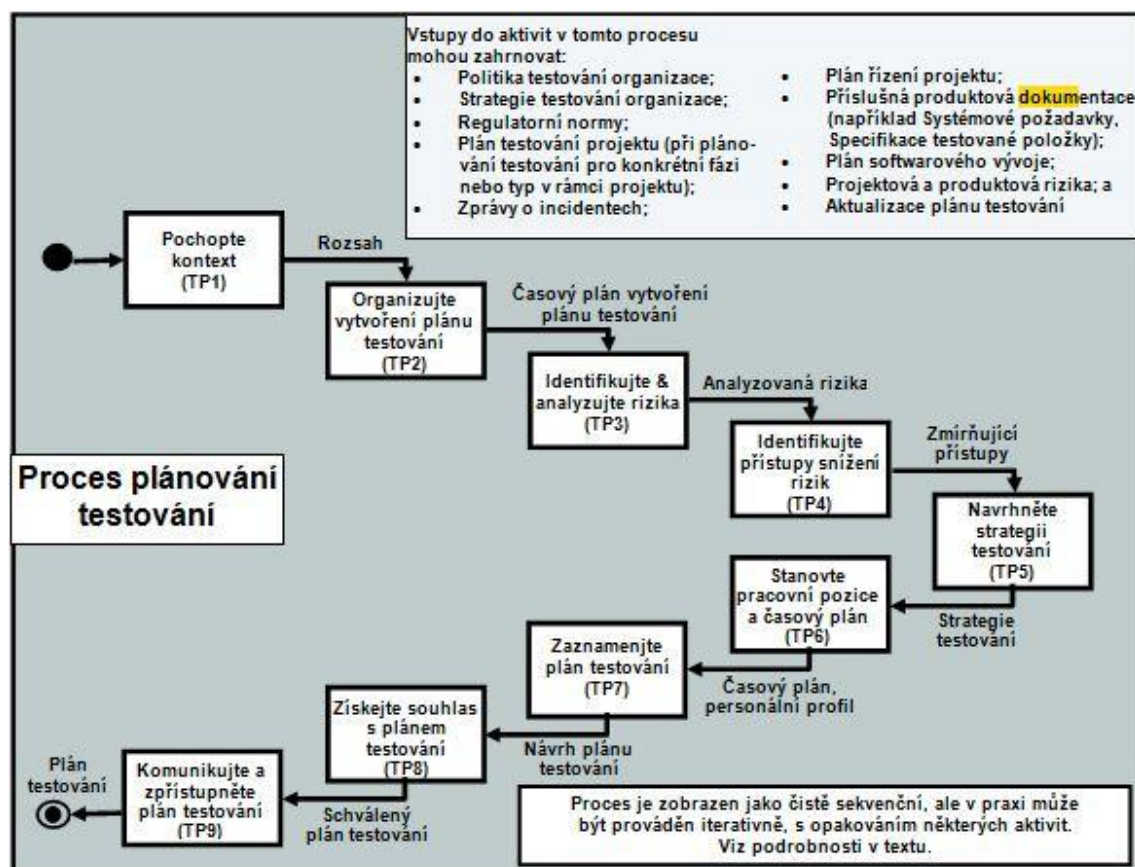
První aktivitou je pochopení celého kontextu testovaného objektu. Provádí se nejčastěji důkladnou analýzou Báze testování. Z analýzy je možné identifikovat rozsah testování nad celým testovaným objektem a zorganizovat si tak vytvoření zbytku Plánu testování. Na vytvoření Plánu testování je nutné vytvořit si také časový plán pro vytvoření Plánu testování, který je třeba dodržet, aby nezasahoval v rámci harmonogramu do ostatních aktivit.

Další aktivitou je Analýza a identifikace rizik. Jak jsem uvedl v předchozí podkapitole (kapitola 6.2) norma ČSN ISO/IEC/IEEE 29119 zahrnuje techniku testování založené na rizicích (risk-based testing), která ovlivňuje návrh testování z hlediska především z hlediska priorit testovaných oblastí. Odpovědná osoba analyzuje rizika, na jejichž základě identifikuje prostředky, techniky a přístupy k minimalizaci identifikovaných rizik.

Analýza rizik se promítne do další aktivity, kterou je vytvoření návrhu strategie testování. Strategie testování v sobě zahrnuje poznatky analýzy Báze testování, analýzy rizik, politiku a strategii testování organizace. Na základě Strategie testování je

vytvořen harmonogram testování. Ten obsahuje stanovení a rozvržení rolí v rámci testovacího týmu a podrobný časový plán na jednotlivé aktivity testování.

Všechny výstupy jsou zaznamenány do Návrhu plánu testování, který je předán na schválení. Schválení Návrhu plánu testování provádí role odpovědná za řídicí procesy v oblasti testování. Dokud Návrh plánu testování není schválen, nemůže být zpřístupněn jako pokyn všem členům testovacího týmu a zúčastněným stranám (ČSN ISO/IEC/IEEE 29119-2, 2014).



Obrázek 3: Proces plánování testování (ČSN ISO/IEC/IEEE 29119-2, 2014)

6.4 Shrnutí

ISO/IEC/IEEE 29119 je mezinárodní technická norma zabývající se testováním softwaru v oblasti softwarového a systémového inženýrství. V současné době má i svou českou lokalizaci ČSN ISO/IEC/IEEE 29119, která definuje mezinárodně dohodnutý soubor procesů a aktivit, jenž poskytuje jednotný rámec oboru testování softwaru pro jakoukoli organizaci.

Norma ČSN ISO/IEC/IEEE 29119 popisuje podrobně definice pojmů v oblasti testování, z nichž velká část spadá i do disciplíny Analýzy a návrhu testování. Norma popisuje většinu aktivit z pohledu celé organizace, včetně definice jednotlivých rolí v oblasti testování. Manažer testování zastává řídicí funkce a činnosti. Tester vykonává vlastní provádění testování. Analytické činnosti zastává na úrovni organizace Stratég testování. Na úrovni projektu, nebo testovacího týmu je však zastává Analytik testování. Toho norma ČSN ISO/IEC/IEEE 29119 uvádí pouze nepřímo.

Norma dále uvádí plno dalších definic pojmů, které spadají do oblastí testování s ohledem na disciplíny, techniky a pracovní produkty. Patří mezi ně například vysvětlení, co je to Báze testování, testovaný objekt, Strategie testování, Plán testování, nebo Testovací případy.

Z hlediska procesu testování norma ČSN ISO/IEC/IEEE 29119 preferuje testování založené na rizicích. Jedná se o moderní přístup k testování, který klade největší důraz na nejrizikovější oblasti testovaného objektu. Analýza a návrh testování díky analýze rizik hraje v procesu testování důležitou roli. Většina analytických činností v oblasti testování se provádí na začátku testovacího procesu. Mezi nejdůležitější analytické činnosti patří analýza Báze testování, analýza rizik, definice Strategie testování, Návrh plánu testování, identifikace testovacích podmínek a tvorba testovacích případů.

7 Návrh na rozšíření metodiky MMSP

V této kapitole se zaměřuji na rozšíření metodiky MMSP (Metodiky pro malé softwarové projekty) uvedené v kapitole 4.4, o disciplínu Analýza a návrh testování, kterou jsem charakterizoval v kapitolách 3, 5 a 6. V rozšiřování metodiky MMSP vycházím z poznatků získaných z literatury uvedené ve zdrojích, analýzy podkladů pro certifikace ISTQB, analýzy české lokalizace mezinárodní technické normy pro testování ČSN ISO/IEC/IEEE 29119, best-practice a vlastní praxe. Verze metodiky, ze které vycházím, je poslední verze MMSP 1.5 zpracovaná pomocí nástroje Eclipse Process Framework Composer (Rejnková, 2011b).

Rozšíření metodiky, jejích procesů a aktivit, probíhá na úrovni projektu. Tato úroveň na rozdíl od vyšší úrovně (na úrovni organizace), nebo nižší úrovně (úroveň dílčích částí projektu) nejlépe odpovídá použití metodiky MMSP. Metodika je na této úrovni rovněž popisována. Rozšíření metodiky MMSP se týká hlavně oblasti testování a do ostatních činností, jako jsou Analýza, vlastní Implementace, nebo Řízení projektu zasahuje pouze minimálně. Rozšíření metodiky MMSP respektuje její stávající strukturu a dotkne se následujících oblastí:

- Role
- Disciplíny a úlohy
- Pracovní produkty
- Životní cyklus

V rámci rozšíření se zaměřuji primárně na rozšíření oblasti testování, přesto budu v rámci práce uvádět i obecný kontext metodiky MMSP s elementy výše uvedených oblastí metodiky MMSP, ale zaměřím se hlavně na ty, které se týkají testování.

Cílem kapitoly je vytvoření návrhu na aktualizaci a rozšíření metodiky MMSP o analytické činnosti v oblasti testování tak, aby byla konkurenceschopná s ostatními metodikami vývoje softwaru a odpovídala současným potřebám testování. Rozšířením metodiky MMSP o disciplínu Analýza a návrh testování bude možné tuto disciplínu možné využít i na malých projektech, což zatím nebylo možné. Propracované analytické činnosti v oblasti testování se zatím vyskytují převážně v metodikách vývoje softwaru, které jsou vhodné především pro velké projekty. Zavedení disciplíny Analýzy a návrhu testování do metodiky MMSP má za úkol rozšíření metodiky MMSP v oblasti analytických úloh testování a zavedení nové role Analytik testování.

7.1 Úprava rolí metodiky MMSP

Metodika MMSP popisuje role, které plní jednotlivé úlohy v rámci životního cyklu projektu. Tyto role vykonávají alokovaní IT specialisté. Metodika MMSP uvádí (Buchalcegová a Stanovská, 2013) následující role:

- Architekt
- Projektový manažer
- Analytik
- Vývojář
- Tester
- Nedefinovaná role
- Zainteresovaná strana

Metodika MMSP obsahuje v oblasti testování pouze roli Tester. IT specialista zastávající tuto roli vykonává plánování, přípravu a provádění testů. Z analýzy definované role Tester vyvozují, že se jedná o základní pojetí disciplíny testování, které má na starosti základní odhalování chyb a jejich následnou komunikaci s ostatními členy týmu. Role testera je s ohledem na současný stav minimalistická. Ostatní role pro testování vytvářejí vstupy anebo zpracovávají jeho výstupy.

V rámci rozšíření metodiky MMSP v oblasti rolí počítám se zavedením nové role Analytik testování. Tato role vykonává disciplínu Analýza a návrh testování, vymezenou v kapitole 3, která zahrnuje veškeré analytické činnosti v oblasti testování a s tím spojený i návrh testování. Analytik testování přebírá analytické činnosti od role Tester a je dále rozšířen o další činnosti vyplývající z disciplíny Analýza a návrh testování. Role Testera je zredukována na jednoduché činnosti manuálního testování, které je schopen zastávat i méně kvalifikovaný IT specialista.

Návrh na rozšíření metodiky MMSP v oblasti rolí testování nevyžaduje striktní dodržování poměru rolí a osob 1:1. V rámci návrhu na rozšíření metodiky MMSP o disciplínu Analýza a návrh testování počítám s tím, že se jednotlivé role mohou překrývat v závislosti na počtu osob v testovacím týmu.

Vzhledem k tomu, že testování považuji za plnohodnotnou a komplexní disciplínu, rozšíření metodiky MMSP v oblasti testování vyžaduje také řízení testovacího procesu. Rozšíření metodiky MMSP v oblasti řízení testovacího procesu je popsáno v diplomové práci *„Testování a kvalita softwaru v metodikách vývoje*

softwaru“ (Vachalec, 2014), která se sice zabývá rozšířením metodiky MMSP o agilní přístupy, ale definuje roli Manažera testování. Tato role však zatím nebyla zahrnuta do veřejně dostupné verze MMSP 1.5 na <http://mmsp.czweb.org/> (Rejnková, 2011b). Role Manažera testování má na starosti řízení testování a celého testovacího procesu, tato role však neplní hlavní část analytických činností v oblasti testování, je již definována, a proto se na ní v rámci diplomové práce pouze odkazují.

Další rozšíření rolí metodiky MMSP je popsáno v diplomové práci *„Prototypování při vývoji softwaru“* (Nývlt, 2012). Tato práce popisuje rozšíření metodiky MMSP o role:

- Návrhář uživatelského rozhraní (Návrhář UI)
- Grafik

Tyto role také zatím nejsou zahrnuty do veřejně dostupné verze MMSP 1.5 na <http://mmsp.czweb.org/> (Rejnková, 2011b). Ostatní stávající role jsou rozšířením metodiky MMSP dotčeny pouze okrajově.

V rámci rozšíření metodiky MMSP v oblasti analýzy a návrhu testování obsahuje rozšířená metodika MMSP tyto role:

- Architekt
- Projektový manažer
- Analytik
- Vývojář
- Návrhář uživatelského rozhraní (Návrhář UI)
- Grafik
- Manažer testování
- Analytik testování
- Tester
- Nedefinovaná role
- Zainteresovaná strana

7.1.1 Manažer testování

Manažer testování (nebo také Test manažer) má odpovědnost za proces testování a za kvalitu výsledného softwarového produktu. Vachalec uvádí (Vachalec, 2014) členění úloh, které rozděluje analytické činnosti v oblasti testování mezi role Manažera testování a Testera (viz kapitola 7.1.3). Manažer testování podle Vachalce plní úlohy Plánování testů, Revizi požadavků, Řízení testování a reportování, Vyhodnocení a uzavření testování. V rámci rozšíření přechodu analytické činnosti v oblasti testování, především úlohu Plánování testů, do nové transformované úlohy Plánování testování, a úlohu Vyhodnocení a uzavření testování, které vykonává nová role Analytika testování (viz kapitola 7.1.2). Ostatní úlohy vykonává Manažer testování i nadále. Nově vykonává úlohu Revize Analýzy a návrhu testování, která v sobě zahrnuje kontrolu analytických činností v oblasti testování prováděné Analytikem testování.

Manažer testování má na starosti také vedení testovacího týmu a řešení problémů, které souvisí s nedostatkem během testování. Rovněž také spolupracuje na analytických činnostech v oblasti testování, přičemž primárně schvaluje výstupy plynoucí z Analýzy a návrhu testování. Vymezení oblasti řízení testování bylo zpracováno například v rámci diplomové práce *„Metodika testování podle mezinárodních praktik a standardů“* (Králová, 2013), která se zabývá touto problematikou.

Vachalec dále uvádí (Vachalec, 2014), že Manažer testování je odpovědný za pracovní produkty Plán testování, Závěrečná zpráva o testování, Záznam o defektech v požadavcích a Zpráva o stavu testování. Plán testování v rámci zavedení disciplíny Analýza a návrh testování přechází na novou roli Analytika testování. Odpovědnost za Závěrečnou zprávu o testování a Záznam o defektech v požadavcích sdílí manažer testování společně s Analytikem testování. Odpovědnost za Zprávu o stavu testování manažerovi testování zůstává.

V rámci rozšíření metodiky MMSP nebude role dále upravována. V tabulce (Tabulka 1) uvádím aktualizovanou podobu role Manažera testování po rozšíření metodiky MMSP s ohledem na disciplínu Analýza a návrh testování.

Tabulka 1: Schéma vztahů role Manažer testování (Zdroj: autor)

Úlohy	Řízení testování a reportování Revize požadavků Revize Analýzy a návrhu testování
Odpovědnost	Závěrečná zpráva o testování (sdíleno) Záznam o defektech v požadavcích (sdíleno) Zpráva o stavu testování
Techniky	--
Životní cyklus	Fáze Zahájení Fáze Rozpracování Fáze Konstrukce Fáze Zavedení

7.1.2 Analytik testování

Analytik testování je nově zavedená role, je součástí testovacího týmu a je podřízený manažerovi testování. V rámci testovacího týmu je Analytik testování odpovědný za veškeré analytické činnosti v oblasti testování. Zároveň přebírá stávající analytické činnosti role Testera. Má na starosti nejen analýzu, ale i návrh a přípravu veškerých podkladů potřebných pro provádění testování, včetně jejich aktualizace na základě zpětné vazby po dokončení jednotlivých iterací testovacího cyklu.

Role Analytika testování se aktivně účastní i provádění testování. Díky analytickému přístupu není tato role vázána během testování návrhem testů, ale může testovat spontánně na základě svých zkušeností. Analytik testování provádí zároveň tzv. white-box testování (viz kapitola 5.3), během kterého je schopen nejen odhalovat chyby testovaného objektu jako Tester, ale zjišťovat i jejich povahu a příčiny.

Analytik testování je proto aktivní během celé doby trvání projektu. Nejčastěji komunikuje s manažerem testování, analytikem a ostatními testery. V tabulce níže (Tabulka 2) uvádím aktualizovanou podobu role Manažera testování po rozšíření metodiky MMSP s ohledem na disciplínu Analýza a návrh testování.

Tabulka 2: Schéma vztahů role Analytik testování (Zdroj: autor)

Úlohy	Analýza testovatelnosti Analýza rizik testování Analýza Báze testování Analýza a identifikace testovacích podmínek Definování Strategie testování Plánování testování Návrh testování Příprava a tvorba testů Provedení testování Vyhodnocení a ukončení testování
Odpovědnost	Dokument Analýza testovatelnosti Dokument Analýza rizik testování Báze testování Testovací podmínky Strategie testování Plán testování Návrh testování Testovací scénář Testovací sada Testovací data Záznam výsledků testů Report hodnocení kvality
Techniky	Testování založené na rizicích Testování tříd ekvivalence Analýza okrajových hodnot Kombinatorická analýza
Životní cyklus	Fáze zahájení Fáze Rozpracování Fáze Konstrukce Fáze Zavedení

Požadavky na roli:

Osoba v roli Analytika testování by měla splňovat následující požadavky:

- Praxe v testování alespoň 3 roky
- Znalost metodologie testování
- Analytické schopnosti
- Zkušenost s návrhem testování
- Znalost SQL, Power Designer

Přiřazení role:

Roli Analytika testování může zastávat seniorní tester, který má dostatečné zkušenosti s testováním testovaného objektu, zná životní cyklus projektu a osvědčil se při vytváření testovacích případů. V případě malého testovacího týmu na malých projektech může být role sdílena s rolí Manažera testování. V tomto případě ale chybí kontrola při schvalování výstupů Analýzy a návrhu testování, protože si osoba schvaluje výstupy sama. V rámci metodiky MMSP roli Analytika testování zastává vždy jedna osoba. Tato osoba v roli Analytika testování může vykonávat částečně i jinou roli.

7.1.3 Tester

Tester je role, kterou metodika MMSP již obsahuje. V rámci testovacího týmu je Tester podřízen nově Manažerovi testování. Tester má na starosti provádění předem připravených testovacích případů (provádí statické testování). Kvůli zavedení role Analytika testování je role Testera výrazně zjednodušena. Tester nemá odpovědnost za analytické činnosti v oblasti testování. Tester nejčastěji konzultuje své problémy s Analytikem testování, nebo Manažerem testování. Tabulka 3 zobrazuje přehled role Tester po rozšíření metodiky MMSP o analýzu a návrh testování.

Tabulka 3: Schéma vztahů role Tester (Zdroj: autor)

Úlohy	Provedení testů
Odpovědnost	Záznam výsledků testů
Techniky	--
Životní cyklus	Fáze Rozpracování Fáze konstrukce Fáze zavedení

Požadavky na roli:

Osoba v roli Analytika testování by měla splňovat následující požadavky:

- znalosti a zkušenosti s různými testovacími přístupy i vlastním testováním
- schopnost rozpoznat a řešit problémy

Přiřazení role:

Role Tester tak může být přidělena komukoli s elementární schopností rozpoznávat problémy. Kvůli zavedení role Analytika testování, nemá Tester na starosti tvorbu podkladů pro testování. To však nevylučuje, že Tester nesmí Analytikovi testování pomáhat a naopak. Obecně se doporučuje do role Testera obsadit více osob.

7.1.4 Architekt

Role architekta je odpovědná za vytvoření a popis architektury testovaného objektu, včetně zásadních technických rozhodnutí, která vymezují způsob implementace vyvíjeného softwarového produktu (Buchalceková a Stanovská, 2013). Metodika MMSP roli Architekta již definuje. Popis architektury je součástí Báze testování, která slouží jako vstup pro analýzu a návrh testování. Role Architekt a disciplína Analýza a návrh testování se dále nijak neovlivňují.

7.1.5 Projektový manažer

Role Projektového manažera je v metodice MMSP odpovědná za plánování, vedení a koordinaci celého projektu vývoje softwarového produktu (Buchalceková a Stanovská, 2013). V rámci rozšíření metodiky MMSP v oblasti testování není role Projektového manažera nijak zasažena.

7.1.6 Analytik

Roli Analytika metodika MMSP již definuje. Analytik se stará o sběr požadavků od všech zainteresovaných stran. Tyto požadavky analyzuje, identifikuje priority požadavků a na jejich základě vytváří návrh softwarového produktu (Buchalceková a Stanovská, 2013). Návrh softwarového produktu má formu dokumentace je součástí Báze testování. Analytik tedy vytváří podklady pro vstupy

disciplíny Analýzy a návrhu testování a jejich úloh. V rámci rozšíření metodiky MMSP v oblasti analýzy a návrhu testování nebude role měněna.

7.1.7 Vývojář

Další již definovaná role metodiky MMSP je Vývojář. Tato role je odpovědná za implementaci navrženého softwarového řešení. Při implementaci IS/ICT řešení Vývojář vychází z popisu architektury, požadavků a analýzy návrhu softwarového produktu (Buchalcegová a Stanovská, 2013). Vývojář se podílí na tvorbě testovaného objektu, který je podroben testování. V rámci rozšíření metodiky MMSP v oblasti analýzy a návrhu testování nebude role měněna.

7.1.8 Návrhář uživatelského rozhraní (Návrhář UI)

Návrhář uživatelského rozhraní má na starosti vytvoření návrhu grafického rozhraní softwarového produktu. Vytváří ho pomocí definování prototypů, které jsou součástí dokumentace. Tato role byla do metodiky MMSP přidána podle Nývlt v rámci rozšíření o oblast prototypování (Nývlt, 2012). V rámci rozšíření metodiky MMSP v oblasti analýzy a návrhu testování nebude role měněna.

7.1.9 Grafik

Grafik má na starosti kompletní vývoj grafického rozhraní softwarového produktu. Vychází přitom z analýzy návrhu softwarového produktu a prototypů. Tato role byla do metodiky MMSP přidána podle Nývlt v rámci rozšíření o oblast prototypování (Nývlt, 2012). V rámci rozšíření metodiky MMSP v oblasti analýzy a návrhu testování nebude role měněna.

7.1.10 Nedefinovaná role

Metodika MMSP popisuje (Buchalcegová a Stanovská, 2013) nedefinovanou roli. Tuto roli zastává IT specialista, kterého není možné nikam zařadit. V rámci rozšíření metodiky MMSP v oblasti analýzy a návrhu testování nebude role měněna.

7.1.11 Zainteresaná strana

Zainteresaná strana je poslední rolí definovanou v metodice MMSP (Buchalcegová a Stanovská, 2013). Zainteresaná strana reprezentuje zadavatele a

budoucí uživatele softwarového produktu, kteří mají zájem na jeho úspěšném dokončení. Zainteresaná strana definuje požadavky a tím poskytuje vstupy pro analýzu návrhu softwarového produktu, který je následně i součástí podkladů pro analytické činnosti v oblasti testování. V rámci rozšíření metodiky MMSP v oblasti analýzy a návrhu testování nebude role měněna.

7.1.12 Rozvržení testovacího týmu

Metodika MMSP, která je určena pro malé projekty, jež mají za účel vývoj nekritických softwarových produktů, je určena také pro malé projektové týmy. Při rozšiřování metodiky MMSP je proto nutné brát ohled na omezené lidské zdroje. Návrh rozšíření metodiky MMSP počítá s omezeným množstvím členů testovacího týmu. Zavedením disciplíny Analýzy a návrhu testování se díky zvýšené efektivitě testování sníží požadavek na počet Testerů. Také díky sníženým požadavkům na roli Testera se zvýší jeho zastupitelnost. V praxi potom půjde testery substituovat sdílením některých rolí, nebo zaměstnáním brigádníků během období provádění testování. Tabulka 4 ukazuje doporučené rozvržení členů testovacího týmu v závislosti na počtu jeho členů.

Tabulka 4: Doporučené rozvržení členů testovacího týmu (Zdroj: autor)

Role/počet členů test. týmu	1	2	3	4	5
Manažer testování	1/3	1/2	1/2	1	1
Analytik testování	1/3	1/2	1/2	1	1
Tester	1/3	1	2	2	3

7.2 Zavedení disciplíny Analýza a návrh testování do metodiky MMSP

Cílem této kapitoly je zavedení nové disciplíny Analýza a návrh testování mezi ostatní disciplíny metodiky MMSP. V kapitole popisují jednotlivé úlohy disciplíny Analýza a návrh testování. Pro lepší přehlednost jsou vztahy jednotlivých úloh shrnuty do tabulek, které uvádím i kvůli dodržení integrity s metodikou MMSP. Strukturální rozřazení disciplín a jejich posloupné seřazení v metodice MMSP je nutné nezaměňovat s cyklem životního cyklu projektu. Toto rozřazení slouží pouze pro snazší orientaci mezi úlohami jednotlivých disciplín metodiky MMSP.

Na začátku diplomové práce jsem čelil rozhodnutí, zda začlenění analytických činností v oblasti testování je pouze o rozšíření stávající disciplíny testování, nebo se jedná o samostatnou a plnohodnotnou disciplínu. Po zanalyzování dostupných zdrojů v teoretické části považuji Analýzu a návrh testování za samostatnou disciplínu (viz kapitola 3). Disciplínu Analýza a návrh testování vykonává IT specialista zastávající roli Analytika testování. Tato disciplína obsahuje specifickou množinu analytických činností rozčleněných do následujících úloh:

- Analýza testovatelnosti
- Analýza rizik
- Analýza Báze testování
- Analýza a identifikace testovacích podmínek
- Definování Strategie testování
- Plánování testování
- Návrh testování
- Příprava a tvorba testů
- Provedení testování
- Vyhodnocení a ukončení testování

Úlohy disciplíny mají dopad na řídicí procesy, činnosti a úlohy, především v oblasti řízení testování. Tyto činnosti a úlohy mají především kontrolní povahu. Manažer testování provádí revize výstupů některých disciplín. Stejně, jako provádí revizi požadavků, které tvoří výstupy disciplíny Analýza, provádí i revizi výstupů disciplíny Analýzy a návrhu testování, které tvoří výstupy analytických činností v oblasti testování.

7.2.1 Úloha Analýza testovatelnosti

Analýza testovatelnosti testovaného objektu je jednou ze základních úloh disciplíny Analýza a návrh testování, kterou provádí Analytik testování. S touto úlohou analytických činností se setkávám ve své praxi velmi často. Analýza testovatelnosti identifikuje do jaké míry a s jakou efektivitou je testovací tým schopen testovaný objekt pokrýt testy, otestovat a vyhodnotit kvalitu. Analýza testovatelnosti se provádí zevrubnou analýzou výstupů disciplíny Analýza. Provádí se především v rané fázi iterace životního cyklu projektu a zpravidla není vždy kompletní, což vyžaduje face-to-

face konzultace s Analytikem a někdy i s Vývojářem pro doplnění nezbytných informací (viz Tabulka 5). Analýza testovatelnosti má dopad na Návrh testování.

Tabulka 5: Vztahy Analýzy testovatelnosti (Zdroj: autor)

Role	Primární role	Další role
	Analytik testování	Vývojář Analytik
Vstupy	Povinné	Volitelné
	Popis architektury Požadavky	Báze testování
Výstupy	Dokument Analýza testovatelnosti	

Alan Page uvádí (Page, Johnston a Rollison, 2009), že Analýza testovatelnosti slouží i jako zpětná vazba pro návrh implementace softwarového produktu. Na základě analýzy testovatelnosti je vhodné doplnit analýzu a případně ji i rozšířit pro potřeby testování. Takovým příkladem může být vytvoření dodatečné funkcionality pro pozdější zautomatizování testů. Další možností zvýšení testovatelnosti na základě Analýzy testovatelnosti je dodržování jednoduchého modelu, který se skládá následujících čtyř kritérií (Page, Johnston a Rollison, 2009):

- Jednoduchost – čím jsou komponenty testovaného objektu jednodušší, tím jdou i lépe testovat.
- Sledovatelnost – čím je viditelnost dat a čitelnost kódu lepší, tím lze i lépe vyhodnotit výsledek testu.
- Ovladatelnost – obsahuje-li testovaný objekt prahové (okrajové) hodnoty, tak přidáním možnosti je měnit, se testování zjednodušuje.
- Informovanost – čím kompletnější je dokumentace (Báze testování), tím větší má testovací tým jistotu, že výsledky testování jsou správné.

7.2.2 Úloha Analýza rizik testování

Úloha Analýza rizik testování se provádí v případě, že životní cyklus projektu zahrnuje techniku Testování založené na rizicích (viz kapitola 7.3.1). V tomto případě se testování včetně analýzy a návrhu testování zaměřuje primárně na nejkritičtější části testovaného objektu, tak jak to uvádí česká lokalizace normy pro testování (ČSN ISO/IEC/IEEE 29119-2, 2014). Role Analytika aplikuje techniku testování založené na rizicích a podrobně analyzuje výstupy analýzy a návrhu implementace a snaží se o co nejpresnější identifikaci hrozeb a rizikových míst testovaného objektu. Závěry Analýzy

rizik testování jsou podrobené revizi Manažera testování, který musí výstup verifikovat. Výstupem je shrnutí poznatků analýzy rizik do dokumentu Analýza rizik testování. Analýza rizik testování má dopad na ostatní úlohy disciplíny Analýzy a návrhu testování: Návrh testování a Příprava a tvorba testů.

Tabulka 6: Vztahy Analýzy produktových rizik (Zdroj: autor)

Role	Primární role	Další role
	Analytik testování	Manažer testování
Vstupy	Povinné	Volitelné
	Popis architektury Požadavky	
Výstupy	Dokument Analýza rizik testování	

7.2.3 Úloha Analýza Báze testování

Analýza báze testování je jednou z nejdůležitějších úloh disciplíny Analýza a návrh testování. Tuto úlohu provádí Analytik testování. Jejím základem je analýza a porozumění všech možných vstupů pro testování. Česká lokalizace normy pro testování uvádí (ČSN ISO/IEC/IEEE 29119-1, 2014), že se jedná o kompletní souhrn znalostí potřebný pro Návrh testování a má nejčastěji formu dokumentace, avšak může mít i formu nezdokumentovaného porozumění požadovaného chování testovaného objektu.

V praxi se Analýza Báze testování se provádí podrobnou analýzou popisu architektury, funkčních požadavků a dokumentu Funkční specifikace. Jejím cílem je shromáždit veškeré dostupné informace o testovaném objektu za účelem co nejpresnějšího pochopení a porozumění. Tomuto nashromáždění se říká vytvoření Báze testování. Báze testování je hlavním vstupem pro Návrh testování.

Může se stát, že podklady pro vytvoření Báze testování jsou nedostatečné. Jedná se o případ, kdy je dokumentace analýzy neúplná, nebo dokonce chybí. V tomto případě doporučuji obracet se s otázkami na Analytika a konzultovat s ním pro doplnění potřebných informací.

Dalším případem komplikace je častá změna požadavků v průběhu projektu, nebo doplnění nových požadavků. Stává se to především během agilního vývoje. V tomto případě doporučuji Bázi testování pravidelně aktualizovat. Přehled vztahů Analýzy Báze testování je zobrazen v tabulce níže (Tabulka 7).

Tabulka 7: Vztahy Analýzy Báze testování (Zdroj: autor)

Role	Primární role	Další role
	Analytik testování	Analytik Manažer testování
Vstupy	Povinné	Volitelné
	Popis architektury Požadavky	
Výstupy	Báze testování	

7.2.4 Úloha Analýza a identifikace testovacích podmínek

Testovací podmínky jsou, jak uvádí české lokalizace normy pro testování (ČSN ISO/IEC/IEEE 29119-1, 2014), testovatelné aspekty testovaného objektu. Cílem analýzy testovacích podmínek je identifikace těchto aspektů z Baze testování. Jako testovací podmínky mohou být identifikovány například:

- Požadované funkce
- Vlastnosti
- Nároky na testovací prostředí
- Specifická testovací data
- Vstupní požadavky

Tyto testovací podmínky jsou zahrnuty do návrhu testování a tvoří vstupní okolnosti pro konkrétní testovací případy. V případě neúplné Baze testování doporučuji konzultovat identifikaci testovacích podmínek s Analytikem. Konečné testovací podmínky musí schválit manažer testování. Přehled vztahů Analýzy a identifikace testovacích podmínek zobrazuje Tabulka 8.

Tabulka 8: Vztahy analýzy a identifikace testovacích podmínek (Zdroj: autor)

Role	Primární role	Další role
	Analytik testování	Analytik Manažer testování
Vstupy	Povinné	Volitelné
	Báze testování	
Výstupy	Testovací podmínky	

7.2.5 Úloha Definování Strategie testování

Metodika MMSP definování strategie testování popisuje jako krok úlohy Plánování testů. V rámci rozšíření metodiky MMSP v oblasti testování zařazují Definování strategie jako samostatnou úlohu. Buchalceková uvádí (Buchalceková,

2005) strategii testování jako definici přístupu k testování, typů a úrovně testů. Definování strategie testování zahrnuje i definici základního způsobu reportování chyb.

V rámci rozšíření metodiky MMSP navrhuji rozšířit definování strategie testování o:

- Plán a vizi projektu
- Popis různých typů testů
- Návod na vytvoření návrhu testů
- Popis procesů
- Výběr použitých metod testování
- Obecný rámec hodnocení rizik
- Plán zdokonalování znalostí a schopností testovacího týmu
- Sdílení znalostí
- Vizi testovacího týmu

Do úlohy definování strategie může zasáhnout Projektový manažer a manažer testování. Výsledným pracovním produktem po dokončení definování strategie testování je dokument strategie testování, kterou musí oba schválit. Vztahy definování strategie testování zobrazuje Tabulka 9.

Tabulka 9: Vztahy definování Strategie testování (Zdroj: autor)

Role	Primární role	Další role
	Analytik testování	Projektový manažer Manažer testování
Vstupy	Povinné	Volitelné
	Plán projektu Vize	Popis architektury Požadavky Případy užití
Výstupy	Strategie testování	

7.2.6 Úloha Plánování testování

Úloha Plánování testování v rámci rozšíření metodiky MMSP v oblasti testování nahrazuje stávající úlohu Plánování testů. Plánování testů bylo zredukováno o definování strategie testování, která se stává samostatnou úlohou (viz kapitola 7.2.5). Z tohoto důvodu úloze zůstal pouze krok Plánování provedení testů.

V rámci rozšíření metodiky MMSP navrhuji transformaci úlohy Plánování testů (respektive Plánování provedení testů) do rozšířené úlohy Plánování testování,

kteřá vychází částečně z české lokalizace normy pro testovací procesy (ČSN ISO/IEC/IEEE 29119-1, 2014) a je popsána v kapitole 6.3.2 Proces plánování testování a zahrnuje Plánování provedení testů do svého kontextu.

Nově vytvořená úloha Plánování testování je stěžejní úlohou disciplíny Analýza a návrh testování, která je prováděna rolí Analytik testování. Plánování testování podléhá strategii testování. V rámci rozšíření metodiky MMSP rozšiřuji i doporučené kroky vedoucí k vytvoření správného dokumentu Plán testování, který je výstupem této úlohy. Doporučené kroky se mohou měnit v případě nutnosti přizpůsobit metodiku projektu. Správné sestavení Plánu by mělo probíhat pomocí těchto kroků:

1. Pochopení Báže testování – Tento krok je důležitý pro pochopení kontextu a významu testovaného objektu a jeho dílčích částí. Tuto aktivitu Analytik tetování provádí v rámci analýzy Báže testování.
2. Pochopení Strategie testování – Strategie testování definuje obecný rámec naplánování celého zbytku testování. Pro vytvoření správného plánu testování je nutné dodržet celkový přístup k testování vycházející ze Strategie testování
3. Zahrnutí analýzy rizik – V případě, že životní cyklus projektu, Strategie testování, nebo plán projektu zahrnuje testování založené na rizicích, je nutné určit prioritu kritických oblastí. Tento krok rozhoduje o dopadu případné Analýzy rizik na Návrh testování, Přípravu a tvorbu testů.
4. Definice milníků testování – V rámci Plánu testování je doporučené vytyčit si důležité kontrolní body. Ty slouží ke kontrole nejen průběhu testovacího procesu, ale i celého projektu.
5. Vytvoření časových odhadů pracnosti – Časové odhady pracnosti jednotlivých oblastí testování, testovacích scénářů, nebo případů napomáhají přesnějšímu vytvoření harmonogramu průběhu testů.
6. Určení odpovědných osob – Tato činnost je zahrnuta v metodice MMSP. Spočívá v alokaci jednotlivých členů testovacího týmu do příslušných rolí (viz kapitola 7.1).
7. Vytvoření harmonogramu průběhu testů (MMSP) – Tato činnost je zahrnuta v metodice MMSP. Cílem činnosti je specifikace časového plánu jednotlivých aktivit provádění testů v iteracích.
8. Seznam předpokládaných výstupů – Tato činnost je zahrnuta v metodice MMSP. Jde o definování očekávaného stavu po dokončení

testování. Výsledný stav provádění testování se na konci testovacího procesu porovnává s předpokládanými výstupy. Očekávaný stav může být definován na různých úrovních (od jednotlivých iterací až po konkrétní testovací scénáře).

9. Zaznamenání plánu testování – Celé plánování testování musí být zaznamenáno do dokumentu Plán testování.
10. Schválení plánu testování – Plán testování musí být schválen Manažerem testování, nebo Projektovým manažerem.

Rozšíření metodiky MMSP v úloze plánování testování nevylučuje použití metody FURPS+ zaměřenou na ošetřování kvality. Přehled vztahů úlohy Plánování testování zobrazuje Tabulka 10.

Tabulka 10: Vztahy Plánování testování (Zdroj: autor)

Role	Primární role	Další role
	Analytik testování	Manažer testování Projektový manažer
Vstupy	Povinné	Volitelné
	Báze testování Strategie testování	Dokument Analýza testovatelnosti Dokument Analýza rizik testování
Výstupy	Plán testování (Milníky testování, časové odhady pracnosti testování, harmonogram průběhu testů, seznam předpokládaných výstupů)	

7.2.7 Úloha Návrh testování

Návrh testování je nová úloha, kterou provádí role Analytika testování v rámci disciplíny Analýza a návrh testování. Při definování této úlohy vycházím částečně z osnov ISTQB (viz kapitola 5.2, str. 30), české lokalizace normy pro testování (ČSN ISO/IEC/IEEE 29119-1, 2014) a částečně z vlastní praxe. Návrh testování podle ISTQB (McKay a kol., 2012) slouží k návrhu testovacích případů a testovacích scénářů, které budou realizovány později. Během vytváření návrhu je nutné dodržovat rozsah Plánu testování. Návrh testování zahrnuje definování použitých technik, metod a postupů pro správné vytvoření testů. Úloha Návrh testování rovněž zahrnuje přiřazení testovacích podmínek k návrhům konkrétních testovacích scénářů. Testovací podmínky jsou analyzovány a identifikovány v úloze Analýza a identifikace podmínek (viz kapitola 7.2.4).

Analytik testování v rámci této úlohy navrhuje, co by mělo být testováno v jednotlivých oblastech testovaného objektu. V případě testování založené na rizicích

navrhuje testovací případy pro rizikové oblasti. V praxi se zaměřuje na návrh funkčních i nefunkčních typů testů. Návrh funkčních typů testů zahrnuje návrh pozitivních scénářů, které vyplývají z Báze testování. Analytik testování se snaží o co nejširší pokrytí testy tak, aby byly pokryty všechny požadavky vyplývající z analýzy. Čím Analytik testování dosáhne lepšího návrhu testování, tím je větší šance, že testovaný objekt projde akceptačními testy zainteresované strany. Opakem pozitivních scénářů jsou scénáře negativní. Návrh negativních scénářů kontroluje, že se testovaný objekt nechová tak, jak nemá. Návrh nefunkčních typů testů slouží k návrhu testovacích případů, které nesouvisí přímo s funkcemi testovaného objektu. Mezi nefunkční testy patří například návrh výkonnostních nebo penetračních testů (ČSN ISO/IEC/IEEE 29119-1, 2014). Součástí této úlohy je i vytvoření návrhu testovacích případů vhodných pro automatizaci.

Návrh testování zahrnuje obrovské množství typů testů. Vytvoření správné kombinace typů, úrovní a správného návrhu testování závisí čistě na úsudku a schopnostech Analytika testování. Analytik může své poznatky konzultovat s ostatními testery. Přehled vztahů úlohy Návrh testování zobrazuje Tabulka 11.

Tabulka 11: Vztahy Návrhu testování (Zdroj: autor)

Role	Primární role	Další role
	Analytik testování	Manažer testování Tester
Vstupy	Povinné	Volitelné
	Strategie testování Plán testování Báze testování Testovací podmínky Dokument Analýza rizik testování	Popis architektury Požadavky Dokument Analýza testovatelnosti
Výstupy	Dokumentace návrhu testování	

7.2.8 Úloha Příprava a tvorba testů

Další úlohou disciplíny Analýza a návrh testování je Příprava a tvorba testů. Tuto úlohu metodika MMSP z větší části obsahuje. Metodika MMSP uvádí (Buchalceková, 2005), že se jedná o tvorbu konkrétních testovacích případů a sad. Osnovy ISTQB (McKay a kol., 2012) nazývají tuto úlohu Realizace. V rámci rozšíření metodiky MMSP doplňují tuto úlohu o fakt, že se jedná o realizaci konkrétních testovacích případů na základě provedeného Návrhu testování (viz kapitola 7.2.7), jejich

setřídění a přiřazení testovacích podmínek (viz kapitola 7.2.4). Přehled vztahů úlohy přípravy a tvorby testů zobrazuje Tabulka 12.

Tabulka 12: Vztahy Přípravy a tvorby testů (Zdroj: autor)

Role	Primární role	Další role
	Analytik testování	Manažer testování Tester
Vstupy	Povinné	Volitelné
	Báze testování Dokumentace návrhu testování Testovací podmínky	
Výstupy	Testovací případy, Testovací scénáře, Automatické testy, Testovací data	

7.2.9 Úloha Provedení testování

Role Analytik testování je činná v průběhu celého trvání včetně provádění testování. Tato úloha je součástí disciplíny Testování. V praxi se často setkávám s tím, že aby Analytik testování nebyl během provádění testování, kdy iterace nevyžaduje provádění analytických činností, nečinný, tak pomáhá s prováděním testování. Analytik testování díky svým znalostem o testovaném objektu, není vázán nutností provádět testovací scénáře, ale provádí tzv. průzkumné testování. Průzkumné testování je testování založené na zkušenosti, kdy Analytik testování spontánně navrhuje a testuje skryté vlastnosti testovaného objektu (ČSN ISO/IEC/IEEE 29119-1, 2014). Při provádění průzkumného testování vychází z Báze testování. Průzkumné testování je specifický případ úlohy Provedení testování (viz Tabulka 13).

Tabulka 13: Vztahy Provedení testování (Zdroj: autor)

Role	Primární role	Další role
	Analytik testování	
Vstupy	Povinné	Volitelné
	Báze testování	Testovací případy Testovací scénáře
Výstupy	Záznam výsledků testů	

7.2.10 Úloha Vyhodnocení a ukončení testování

Vyhodnocení a uzavření testování je poslední úlohou v testovacím cyklu nebo iteraci. Tuto úlohu v praxi provádí Analytik testování, Manažer testování, nebo oba dva. V rámci rozšíření metodiky MMSP úlohu popisují pro vykonávání Analytikem testování. Manažer testování má na starosti pouze kontrolu výstupů této úlohy, kvůli rozdělení činností provedení a kontroly úlohy mezi dvě různé role.

Hlavním vstupem je záznam provádění testů. V rámci této úlohy Analytik testování analyzuje poznatky získané v iteraci testovacího procesu. V rámci úlohy analytik testování a příslušné zaznamenávají nalezené defekty testovaného objektu. Na základě zpětné vazby z výsledku testování aktualizuje Bázi testování, Návrh testování, Testovací podmínky a Testovací scénáře. Výstupem je reportování a zhodnocení kvality testovaného objektu. Vztahy vyhodnocení a ukončení testování znázorňuje Tabulka 14.

ISTQB uvádí (McKay a kol., 2012), že součástí této úlohy může být nejen dokumentování změn, ale i rozšíření stávajícího pokrytí testy nad rámec stávající Báze testování, úprava prostředí, nebo spouštění automatických testů (viz kapitola 5.2, str 32).

Tabulka 14: Vztahy Vyhodnocení a ukončení testování (Zdroj: autor)

Role	Primární role	Další role
	Analytik testování	Manažer testování Tester
Vstupy	Povinné	Volitelné
	Záznam provádění testů	
Výstupy	Report o hodnocení kvality testovaného objektu, aktualizace Báze testování, Návrhu testování, Testovacích podmínek a Testovací scénáře	

7.3 Vybrané techniky návrhu testování používané při analýze a návrhu testování

Jako pojem technika je obecně chápán uměle vytvořený nástroj člověka, který shrnuje určité postupy, které člověku umožní dosažení určitého cíle. Podle profesora Chlapka (Chlapek a kol., 2011) technika napomáhá docílit požadovaného výsledku. Technika určuje postupy poslopně jdoucích činností a vymezuje způsob jejich použití. Technikou vývoje softwaru je obecně doporučený postup a definice činností mající za cíl vytvoření softwarový produkt.

Pro aplikaci disciplíny Analýza a návrh testování existuje velké množství analytických technik návrhu testování a popsání všech v této diplomové práci není možné. Z tohoto důvodu popisují pouze čtyři vybrané techniky, které mi přišly nejzajímavější. Neznamená to ale, že ostatní techniky jsou nezajímavé. Popis technik se zaměřuje na jejich praktické využití. Vybranými technikami jsou:

- Testování založené na rizicích

- Testování tříd ekvivalence
- Kombinatorická analýza
- Zlepšování procesů

7.3.1 Testování založené na rizicích

Rozhodnutí použít testování založené na rizicích je definováno ve Strategii testování a v plánu testování (viz kapitola 7.2.2). Podle české lokalizace normy pro testování (ČSN ISO/IEC/IEEE 29119-1, 2014) technika testování založené na rizicích spočívá v následujících krocích:

1. Porozumění Strategii a Plánu testování – je třeba vycházet z obecného rámce pro testování v rámci projektu.
2. Porozumění Bázi testování – je třeba znát požadované chování testovaného objektu.
3. Identifikace rizik testování – identifikace krizových oblastí testovaného objektu.
4. Identifikace hrozeb testování – identifikace krizových oblastí, které se netýkají přímo testovaného objektu, ale jeho okolí.
5. Ohodnocení rizik – analýza rizik a stanovení míry jeho dopadu na testovaný objekt.
6. Stanovení priority rizik – analýza a identifikace priority rizik.
7. Výběr nejvhodnějších forem testování – zvolení nejvhodnějších typů testů a zapracování do návrhu testování.
8. Vytvoření dokumentu Analýza rizik – výstup úlohy Analýzy rizik (viz kapitola 7.4.2).

Příkladem takového rizika testování může být například nekompletní dokumentace požadavků, nefunkčnost testovacího prostředí, nedostatečné kapacity testovacího týmu, absence, nebo nedodržování metodiky vývoje softwaru.

7.3.2 Testování tříd ekvivalence

Jedná se o jednu z velmi užitečných technik návrhu testování používaných v rámci Analýzy a návrhu testování. Z této techniky vzešla, nebo se jí inspirovala celá řada dalších technik používaných při testování softwarových aplikací. Alan Page popisuje tuto techniku (Page, Johnston a Rollison, 2009), jako techniku založenou na

rozdělení velké množiny prvků do logických skupin a podskupin a dále pak vytvoření návrhu na otestování jednotlivých zástupců z každé skupiny. Tato technika je vhodná pro testování komplexních systémů, složitých funkcí s vysokým počtem možných kombinací, nebo testování s malým počtem členů testovacího týmu, jako je tomu na malých projektech kde se používá metodika MMSP.

Použití techniky rozdělení na třídy ekvivalence zvyšuje testovatelnost složitých systémů na základě snížení počtu testů při stejné garanci kvality softwaru. Efektivitu techniky ovlivňují v zásadě dva faktory. Je to míra znalostí IT specialisty a závislost na granularitě tříd (míra hloubky podrobnosti). Podle Page (Page, Johnston a Rollison, 2009) se technika skládá z následujících kroků:

1. Analýza tříd ekvivalence – porozumění testovanému objektu, identifikace oblastí a funkcí testovaného objektu a zvolení vhodné granularity rozdělení případů užití do množin.
2. Dekompozice hodnot – vytvoření množin a podmnožin testování, přičemž každá množina obsahuje prvky, nebo chování, u kterých se předpokládá stejný generovaný výstup. Zároveň se provádí rozdělování prvků množin do validních a nevalidních tříd. Technika rozdělení tříd ekvivalence testuje pozitivní i negativní testovací případy, kdy se u pozitivních případů očekává stejný výstup a u negativních případů ideálně chybová hláška.
3. Testování tříd ekvivalence – spočívá v testování určitého počtu prvků z každé množiny a podmnožiny, jejichž otestováním je kontrolována kvalita celé množiny. Technika rozdělení tříd ekvivalence netestuje okrajové hodnoty.

Technika rozdělení tříd ekvivalence nevylučuje použití dalších technik návrhu testování. V praxi se běžně při návrhu testování využívá kombinace několika technik návrhu testování pro co nejširší pokrytí testy testovaného objektu.

7.3.3 Analýza okrajových hodnot

Analýza okrajových hodnot je další technika návrhu testování, kterou doporučuji použít v rámci disciplíny analýza a návrh testování. Tato technika spočívá v analýze a identifikaci okrajových podmínek a hodnot. Tato technika je velmi účinná. Podle Page (Page, Johnston a Rollison, 2009) jde o historickou zkušenost získanou na základě často se opakujících problémů v mezních hodnotách. V okrajových hodnotách

je statisticky nejvyšší pravděpodobnost výskytu chyb testovaného objektu. Změřím-li návrh testování na okrajové hodnoty, zvýším tím efektivitu úlohy Provádění testů. Doporučený postup pro návrh testování pomocí analýzy okrajových hodnot se skládá z následujících kroků (Page, Johnston a Rollison, 2009):

1. Porozumění Bázi testování – je třeba znát požadované chování testovaného objektu.
2. Identifikace okrajových hodnot – jedná se zdánlivě jednoduchou činnost. Opak je ale pravdou. V rámci analýzy okrajových hodnot identifikujeme nejen číselné vstupy a výstupy, ale i různorodé veličiny včetně skrytých hranic funkcionalit v testovaném objektu. Kód testovaného objektu obsahuje například různé cykly. Úkolem techniky je tyto skryté hranice v cyklech identifikovat, cykly rozparsovat a skryté okrajové hodnoty otestovat. Tato technika zasahuje do testování na úrovni kódu.
3. Definování testů okrajových hodnot – vytvoření, nebo zahnutí testů okrajových hodnot do návrhu testování (viz kapitola 7.2.7).

Technika návrhu testování Analýza okrajových hodnot nevylučuje použití dalších technik návrhu testování.

7.3.4 Kombinatorická analýza

Kombinatorická analýza je technika návrhu testování, která vychází z předpokladu, že některé elementy testovaného objektu jsou na sobě závislé. V běžném provádění testování testujeme tyto elementy náhodným ad hoc testováním. Systematičtější přístup, který zahrnuje návrh testování tak, že otestujeme každou proměnou alespoň jednou v kombinacích založených na párech.

Page uvádí (Page, Johnston a Rollison, 2009) doporučený postup vedoucí k určení kombinatorických testů, následující postup popisuje jeho zjednodušenou formu:

1. Vytvoření vstupní matice základního testu
2. Vytvoření vstupní matice kritických testů
3. Aplikace vstupního modelu na testovaný objekt
4. Úprava vstupního modelu na základě zpětné vazby z předchozího kroku
5. Provedení testů

6. Kontrola výstupů

Technika Kombinatorické analýzy nevylučuje použití dalších technik návrhu testování, ale dokonce je vhodné techniku kombinovat s ostatními technikami. Obzvláště pak s technikou Rozdělení tříd ekvivalence. Tento typ testů doporučuji také automatizovat

7.4 Pracovní produkty

V této kapitole charakterizují nové a upravené pracovní produkty metodiky MMSP, které vznikly dopadem rozšíření metodiky MMSP v oblasti analýzy a návrhu testování. Pracovní produkty jsou výstupy jednotlivých úloh, činností a procesů. Pracovní produkty metodiky MMSP jsou rozřazeny do oblastí, které odpovídají jednotlivým disciplínám. Stávající oblasti pracovních produktů metodiky MMSP (verze 1.5) jsou:

- Projektové řízení
- Řízení změn a konfigurací
- Požadavky
- Architektura
- Testování
- Vývoj

Stávající oblasti pracovních produktů rozšiřují o novou oblast „Analýza a návrh testování“, která koresponduje s disciplínou Analýza a návrh testování. Rozšíření pracovních produktů o novou oblast respektuje strukturu metodiky MMSP. V rámci rozšíření metodiky MMSP v oblasti analýzy a návrhu testování obsahuje rozšířená metodika MMSP v oblasti Analýza a návrh testování tyto pracovní produkty:

- Dokument Analýza testovatelnosti
- Dokument Analýza rizik testování
- Báze testování
- Testovací podmínky
- Strategie testování
- Plán testování
- Návrh testování

- Report hodnocení kvality

Rozšíření metodiky MMSP v oblasti analýzy a návrhu testování se dotýká pracovních produktů i v oblasti Testování. Pracovní produkty týkající se analytických činností se přesouvají do oblasti Analýza a návrh testování. Pro oblast testování budou zachovány následující pracovní produkty:

- Testovací scénář
- Testovací sada
- Testovací data
- Záznam výsledků testů

7.4.1 Dokument Analýza testovatelnosti

Nový pracovní produkt dokument Analýza testovatelnosti je výstupem úlohy Analýza testovatelnosti (viz kapitola 7.2.1) a spadá do oblasti pracovních produktů Analýza a návrh testování. Pracovní produkt má formu textového dokumentu.

Tabulka 15: Vztahy pracovního produktu dokument Analýza testovatelnosti (Zdroj: autor)

Role	Primární role	Modifikována
	Analytik testování	Analytik
Úlohy	Vstup pro:	Výstup pro:
	Plánování testování Návrh testování	Analýza testovatelnosti
Dopad absence	V případě, že chybí, tak hrozí, že se banální chyby objeví až v pozdější fázi testování.	
Důvody potřeby	Pracovní produkt tvoří zpětnou vazbu pro disciplínu Analýza. Rozšiřuje vstupy pro plánování a návrh testování.	
Možnosti zastoupení	Pracovní produkt ve formě textového dokumentu lze zastoupit nepsanou formou (schůzky, ústní sdělení).	

7.4.2 Dokument Analýza rizik testování

Nový pracovní produkt dokument Analýza rizik testování je výstupem úlohy Analýza rizik testování (viz kapitola 7.2.2) a spadá do oblasti pracovních produktů Analýza a návrh testování. Pracovní produkt zahrnuje přehled identifikovaných rizik testování včetně jejich kategorizace a popisu. Dokument by měl obsahovat následující doporučené údaje o rizicích:

- Číslo rizika
- Kategorie rizika

- Situace kdy nastane
- Pravděpodobnost s jakou nastane
- Dopad na testovaný objekt
- Popis rizika
- Návrh řešení

Tabulka 16: Vztahy pracovního produktu dokument Analýza rizik testování (Zdroj: autor)

Role	Primární role	Modifikována
	Analytik testování	Manažer testování
Úlohy	Vstup pro:	Výstup pro:
	Plánování testování Návrh testování	Analýza rizik testování
Dopad absence	Pracovní produkt je určen pro projekty, které zahrnují testování založené na rizicích. V případě že ji projekt zahrnuje a dokument chybí, má jeho absence negativní dopad na plánování a návrh testování. Může znemožnit návrh testování kritických oblastí testovaného objektu.	
Důvody potřeby	Pracovní produkt je nutný pro návrh testování kritických oblastí testovaného objektu.	
Možnosti zastoupení	Pracovní produkt ve formě textového dokumentu lze zastoupit nepsanou formou (schůzky, ústní sdělení). Nejdůležitější části mohou být zahrnuty do Báze testování.	

7.4.3 Báze testování

Nový pracovní produkt Báze testování je výstupem úlohy Analýza Báze testování (viz kapitola 7.2.3) a spadá do oblasti pracovních produktů Analýza a návrh testování. Báze testování je shrnutí všech dostupných informačních zdrojů a podkladů pro testování. Tento pracovní produkt má nejčastěji formu textové dokumentace. Důležité je, aby celá Báze testování byla shromážděna na jednom místě. V praxi může jít o vytvoření interního dokumentového serveru, který je v tomto případě zaměřen konkrétně na testování. V případě Analytika testování dochází zároveň k vytvoření Báze testování v nepsané formě porozumění. Analytik testování si vytváří Bázi testování ve formě znalostí.

Tabulka 17: Vztahy pracovního produktu Baze testování (Zdroj: autor)

Role	Primární role	Modifikována
	Analytik testování	Analytik Manažer testování
Úlohy	Vstup pro:	Výstup pro:
	Analýza a identifikace testovacích podmínek Plánování testování Návrh testování Příprava a tvorba testů	Analýza Baze testování
Dopad absence	Bez pracovního produktu není možné sestavit Plán testování, Návrh testování a vůbec testování provádět.	
Důvody potřeby	Pracovní produkt je nezbytný pro analýzu a identifikaci testovacích podmínek, vytvoření Plánu a Návrhu testování.	
Možnosti zastoupení	Pracovní produkt může mít nepsanou formu znalostí Analytika testování.	

7.4.4 Testovací podmínky

Testovací podmínky jsou nový produkt metodiky MMSP. Tento produkt je výstupem úlohy Analýza a identifikace testovacích podmínek (viz kapitola 7.2.4), která spadá do oblasti pracovních produktů Analýza a návrh testování. Analytik testování analyzuje testovací podmínky z Baze testování. Pracovní produkt má formu seznamu, nebo dokumentace testovacích podmínek. Testovací podmínky definují okolnosti testování a jsou následně zahrnuty do návrhu, přípravy a tvorby testovacích scénářů.

Tabulka 18: Vztahy pracovního produktu Testovací podmínky (Zdroj: autor)

Role	Primární role	Modifikována
	Analytik testování	Analytik Manažer testování
Úlohy	Vstup pro:	Výstup pro:
	Návrh testování Příprava a tvorba testů	Analýza a identifikace testovacích podmínek
Dopad absence	Absence Testovacích podmínek značně komplikuje a snižuje kvalitu Návrhu testování a Přípravy a tvorby testů.	
Důvody potřeby	Pracovní produkt definuje okolnosti	
Možnosti zastoupení	--	

7.4.5 Strategie testování

Strategie testování je nový produkt metodiky MMSP. Tento pracovní produkt je výstupem úlohy Definování strategie testování (viz kapitola 7.2.5) a je součástí oblasti pracovních produktů Analýza a návrh testování. Analytik testování na definování Strategie testování může spolupracovat s projektovým manažerem a

manažerem testování. Strategie testování definuje procesy testování, typy testování a celkový přístup k testování v rámci projektu. Výčet obsahu dokumentace Strategie testování je popsán v kapitole 7.2.5.

Tabulka 19: Vztahy pracovního produktu Strategie testování (Zdroj: autor)

Role	Primární role	Modifikována
	Analytik testování	Projektový manažer Manažer testování
Úlohy	Vstup pro:	Výstup pro:
	Plán testování Návrh testování Příprava a tvorba testů	Definování strategie testování
Dopad absence	Absence Strategie testování znemožňuje provedení úlohy Plánování testování.	
Důvody potřeby	Pracovní produkt definuje celkový přístup k testování	
Možnosti zastoupení	Pracovní produkt nelze zastoupit.	

7.4.6 Plán testování

Pracovní produkt Plán testování je v metodice MMSP již definován. V rámci rozšíření metodiky MMSP v oblasti analýzy a návrhu testování dochází k jeho změně. Tento pracovní produkt je nově výstupem úlohy Plánování testování (viz kapitola 7.2.6) a je součástí oblasti pracovních produktů Analýza a návrh testování. Pracovní produkt Plán testování je dokumentace, která zahrnuje definici milníků, časové odhady, harmonogram testování, přiřazení odpovědných rolí za jednotlivé úlohy a seznam předpokládaných výstupů testování. Tabulka 20 popisuje aktualizovanou podobu pracovního produktu po rozšíření metodiky MMSP.

Tabulka 20: Vztahy pracovního produktu Plán testování (Zdroj: Buchalceová a Stanovská, 2013)

Role	Primární role	Modifikována
	Analytik testování	Projektový manažer Manažer testování
Úlohy	Vstup pro:	Výstup pro:
	Návrh testování Příprava a tvorba testů	Plánování testování
Dopad absence	Pokud není plán testů vytvářen, neexistuje souhrnný dokument definující základní strategii a metodiku testování.	
Důvody potřeby	Plán testů, i když ne tak podrobný, by měl být vypracováván vždy.	
Možnosti zastoupení	Plán testů může být vytvořen jako textový dokument, který je doplněn potřebnými výstupy z nástrojů pro podporu plánování (např. MS Project).	

7.4.7 Návrh testování

Nový pracovní produkt návrh testování je výstupem úlohy Návrhu testování (viz kapitola 7.2.3) a spadá do oblasti pracovních produktů Analýza a návrh testování. Návrh testování je shrnutí všech dostupných podkladů pro přípravu a tvorbu testovacích scénářů a testovacích sad. Tento pracovní produkt má nejčastěji formu textové dokumentace.

Tabulka 21: Vztahy pracovního produktu Návrh testování (Zdroj: autor)

Role	Primární role	Modifikována
	Analytik testování	Manažer testování Tester
Úlohy	Vstup pro:	Výstup pro:
	Příprava a tvorba testů	Návrh testování
Dopad absence	Pokud není Návrh testování vytvořen, neexistuje souhrnný zdroj podkladů pro tvorbu testovacích scénářů.	
Důvody potřeby	Pracovní produkt definuje oblasti testování, návrh testovacích scénářů a sad včetně rizikových oblastí a testovacích podmínek.	
Možnosti zastoupení	Návrh testování má formu textového dokumentu.	

7.4.8 Testovací scénář

Metodika MMSP tento pracovní produkt již definuje. Pracovní produkt Testovací scénář je nově výstupem úlohy Příprava a tvorba testů (viz kapitola 7.2.8) a je ponechán v oblasti pracovních produktů Testování. V rámci rozšíření metodiky MMSP v oblasti analýzy a návrhu testování přechází primární odpovědnost za testovací scénáře na Analytika testování. Podíl Testera se ale neztrácí. Testovací scénář může být Testerem modifikován.

7.4.9 Testovací sada

Metodika MMSP tento pracovní produkt již definuje. Pracovní produkt Testovací sada je nově výstupem úlohy Příprava a tvorba testů (viz kapitola 7.2.8) a je ponechána v oblasti pracovních produktů Testování. V rámci rozšíření metodiky MMSP v oblasti analýzy a návrhu testování přechází primární odpovědnost za testovací sady na Analytika testování. Podíl Testera se ale neztrácí. Testovací sada může být Testerem modifikována.

7.4.10 Testovací data

Metodika MMSP tento pracovní produkt již definuje. Pracovní produkt Testovací data jsou nově výstupem úlohy Příprava a tvorba testů (viz kapitola 7.2.8) a jsou ponechána v oblasti pracovních produktů Testování. V rámci rozšíření metodiky MMSP v oblasti analýzy a návrhu testování přechází primární odpovědnost za testovací data na Analytika testování. Podíl Testera se ale neztrácí. Testovací data mohou být Testerem modifikována.

7.4.11 Záznam výsledků testů

Další pracovní produkt, který metodika MMSP již definuje, je Záznam výsledků. Tento pracovní produkt není v rámci rozšíření metodiky MMSP v oblasti analýzy a návrhu testování měněn. Pouze v případě, že se Analytik testování podílí provádění testování (viz kapitola 7.2.9), musí se podílet na jeho vytvoření, nebo modifikaci.

7.4.12 Report hodnocení kvality

Report hodnocení kvality je nový pracovní produkt, který je výstupem úlohy Vyhodnocení a ukončení testování (viz kapitola 7.2.10) a spadá do oblasti pracovních produktů Analýza a návrh testování. Pracovní produkt Report hodnocení kvality dokumentuje vyhodnocení záznamu výsledků testování a identifikuje stav kvality testovaného objektu po dokončení testování v rámci iterace.

Tabulka 22: Vztahy pracovního produktu Report hodnocení kvality (Zdroj: autor)

Role	Primární role	Modifikována
	Analytik testování	Manažer testování
Úlohy	Vstup pro:	Výstup pro:
	Úlohy řízení testování a řízení projektu	Vyhodnocení a ukončení testování
Dopad absence	Neexistence reportu hodnocení kvality zapříčiňuje absenci zpětné vazby o kvalitě testovaného objektu a kvalitě testovacího procesu.	
Důvody potřeby	Pracovní produkt uspokojuje informační potřebu o stavu kvality testovaného objektu.	
Možnosti zastoupení	Report hodnocení kvality má formu textového dokumentu.	

7.5 Dopad rozšíření metodiky MMSP na životní cyklus

Metodika MMSP uvádí (Buchalcegová a Stanovská, 2013) čtyři fáze životního cyklu projektu:

- Fáze Zahájení
- Fáze Rozpracování
- Fáze Konstrukce
- Fáze Zavedení

Rozšíření metodiky MMSP v oblasti analýzy a návrhu testování nevyžaduje tento životní cyklus, který popisuje proces vývoje softwaru podle metodiky MMSP, žádným způsobem měnit. Rozšíření metodiky MMSP doplňuje jednotlivé fáze životního cyklu o role, úlohy a pracovní produkty disciplíny Analýza a návrh testování. V kapitole popisují pouze rozšíření dílčích fází o úlohy analýzy a návrhu testování včetně jejich milníků v obecném kontextu dílčích etap, iterací, aktivit a úloh životního cyklu dle metodiky MMSP.

7.5.1 Fáze Zahájení

Fáze Zahájení projektu má za cíl definici a pochopení zadání, pro něž podklady poskytuje Zainteresaná strana (Buchalcegová a Stanovská, 2013). V oblasti analýzy a návrhu testování se definuje celkový přístup k testování a analyzuje se testovatelnost zadání, rizika testování a vytváří se souhrn informačních zdrojů pro potřeby návrhu testování.

Fáze Zahájení – Iterace Zahájení ve vztahu k disciplíně Analýza a návrh testování:

- Definování Strategie testování
- Analýza testovatelnosti
- Analýza rizik testování
- Analýza Báze testování
- Analýza a identifikace testovacích podmínek
- Plánování testování

Hlavními milníky pro disciplínu Analýza a návrh testování je schválení Strategie testování a Plánu testování.

7.5.2 Fáze Rozpracování

Fáze rozpracování projektu spočívá ve zpřesňování zadání a započetí vývoje softwarového produktu. Z hlediska disciplíny Analýza a návrh testování je nejdůležitější průběžně aktualizovat Bázi testování a vytvořit Návrh testování. Po dokončení Návrhu testování se Analytik testování soustředí na Přípravu a tvorbu testů. V této fázi životního cyklu projektu dle metodiky MMSP (Buchalceová a Stanovská, 2013) začíná i samotné testování.

Fáze Rozpracování – Iterace Rozpracování ve vztahu k disciplíně Analýza a návrh testování:

- Analýza Baze testování
- Návrh testování
- Příprava a tvorba testů
- Provedení testování
- Vyhodnocení a ukončení testování

Hlavními milníky pro disciplínu Analýza a návrh testování je schválení Návrhu testování a akceptace Testovacích scénářů. Dalším milníkem je vytvoření Reportu hodnocení kvality, který zhodnocuje výsledný stav kvality testovaného objektu po dokončení testování v rámci iterace.

7.5.3 Fáze Konstrukce

Fáze Konstrukce se zaměřuje na vývoj softwarového produktu (Buchalceová a Stanovská, 2013). Z hlediska disciplíny Analýza a návrh testování se provádí kontinuální aktualizace Baze testování a Návrhu tetování na základě zpětné vazby z předchozích iterací. Analytik testování má za úkol udržování aktuálnosti testovacích scénářů a testovacích sad. Rovněž se podílí na úloze Provádění testování. Důležitou úlohou Analytika testování je vyhodnocování stavu kvality testovaného produktu v každé iteraci.

Fáze Konstrukce – Iterace Konstrukce ve vztahu k disciplíně Analýza a návrh testování:

- Analýza Baze testování
- Návrh testování

- Příprava a tvorba testů
- Provedení testování
- Vyhodnocení a ukončení testování

Milníky pro disciplínu Analýza a návrh testování – viz fáze Rozpracování (kapitola 7.5.2).

7.5.4 Fáze Zavedení

Fáze Zavedení softwarového produktu ke klientovi je poslední fází životního cyklu projektu dle metodiky MMSP (Buchalceková a Stanovská, 2013). V této fázi končí testování. Z hlediska disciplíny Analýza a návrh testování se provádí vyhodnocení kvality stavu testovaného objektu po dokončení procesu testování.

Fáze Zavedení – Iterace Zavedení ve vztahu k disciplíně Analýza a návrh testování:

- Příprava a tvorba testů
- Provedení testování
- Vyhodnocení a ukončení testování

Hlavními milníky pro disciplínu Analýza a návrh testování je vytvoření závěrečné zprávy, nebo reportu hodnocení kvality, který zhodnocuje výsledný stav kvality a naplnění cílů testovaného objektu po dokončení testování. Dosažení požadované kvality by mělo garantovat akceptaci od Zainteresované strany.

7.6 Shrnutí

Klíčovým prvkem rozšíření metodiky MMSP o analytické činnosti v oblasti testování je zavedení nové disciplíny Analýza a návrh testování. Analýza a návrh testování je disciplína zahrnující analytické činnosti v oblasti testování. Disciplína Analýza a návrh testování byla až doposud popisována v metodikách, které nejsou vhodné pro použití na malých projektech. Tento návrh na rozšíření Metodiky pro malé softwarové projekty (MMSP) o disciplínu Analýza a návrh testování použití na malých nekritických projektech umožňuje.

Rozšíření metodiky MMSP zahrnuje zavedení disciplíny Analýza a návrh testování včetně všech jejích doporučených úloh, činností a aktivit. Mezi úlohy disciplíny Analýza a návrh testování patří definování Strategie testování, Analýza testovatelnosti, Analýza rizik, Analýza Báze testování, Analýza a identifikace testovacích podmínek, Plánování testování, Návrh testování, Příprava a tvorba testů, kam patří i tvorba testovacích scénářů, testovacích sad a testovacích dat, a Vyhodnocení a uzavření testování.

S tím rozšířením metodiky MMSP o disciplínu Analýza a návrh testování souvisí i zavedení nové role Analytika testování, který provádí naprostou většinu analytických úloh v oblasti testování. Zavedení nové role Analytika testování má dopad i na aktualizaci stávajících rolí.

V rámci rozšíření metodiky MMSP v oblasti analýzy a návrhu testování uvádím i příklad vybraných technik testování a návrhu testování, jako jsou například technika testování založené na rizicích, nebo testování tříd ekvivalence.

Rozšíření metodiky MMSP zahrnuje i definici nových pracovních produktů metodiky MMSP, které souvisí s nově zavedenou disciplínou Analýza a návrh testování. Pracovní produkty jsou výstupy úloh disciplíny Analýza a návrh testování, které mají převážně formu dokumentace.

8 Závěr

Na závěr své diplomové práce bych rád vyjádřil své přesvědčení, že jsem splnil hlavní cíl práce, který jsem si stanovil v úvodu. V diplomové práci charakterizuji poměrně novou disciplínu Analýza a návrh testování a provádím její zapracování do Metodiky pro malé softwarové projekty (MMSP) tak, aby bylo možné tuto disciplínu reálně použít i na menších projektech vývoje nekritického softwaru.

Hlavního cíle diplomové práce bylo dosaženo splněním dílčích cílů. Prvním dílčím cílem byla charakteristika disciplíny Analýza a návrh testování. V práci charakterizuji disciplínu Analýza a návrh testování, jako disciplínu v oblasti vývoje IS/ICT technologií, která se zaměřuje na analytické činnosti v oblasti testování softwaru. Analýza a návrh testování se zabývá poznáváním, rozbořem a identifikací zadání, potřeb a požadavků na oblast testování. Také popisují principy a úlohu disciplíny Analýza a návrh testování v kontextu testování softwaru a použití při vývoji softwarových produktů.

Druhým dílčím cílem bylo přiblížit, jak se k problematice testování a jeho analytickým činnostem staví obecně známé modely a metodiky vývoje softwaru. Na vybraných příkladech modelů (Vodopád, Spirála) a metodik (RUP, Scrum, OpenUP, MMSP) analyzují různé přístupy z hlediska přístupu k testování a jeho analytickým činnostem.

Třetím dílčím cílem bylo analyzovat, jak se k problematice analytických činností v oblasti testování staví certifikace, standardy a normy v oblasti testování. Jako nejzásadnější jsem pro oblast analýzy a návrhu testování identifikoval ISTQB (mezinárodní sdružení testerů), které poskytuje certifikace v oblasti testování softwaru, a českou lokalizaci mezinárodní normy pro testování ČSN ISO/IEC/IEEE 29119. V práci analyzují přístup k analytickým činnostem v oblasti mezinárodních certifikací v testování ISTQB a požadavky na certifikaci odborných rolí „Test analytik“ a „Technický test analytik“. Dále v rámci dílčího cíle analyzují technickou normu ČSN ISO/IEC/IEEE 29119, která spadá do norem kategorie „Softwarové a systémové inženýrství“ a konkrétně popisuje „Testování softwaru“. Tato norma definuje základní pojmy oblasti testování a jejích analytických činností, které analyzují a v pozdější části diplomové práce z nich vycházím při rozšiřování metodiky MMSP o disciplínu Analýza

a návrh testování. Také analyzují doporučené postupy při provádění analytických činností, které jsou uvedeny obecně v kontextu testování.

Posledním dílčím cílem bylo vypracovat vlastní návrh na rozšíření vybrané metodiky, kterou je již zmiňovaná Metodika pro malé softwarové projekty, o charakterizovanou oblast analýzy a návrhu testování. Klíčovým prvkem rozšíření je identifikovaná disciplína Analýza a návrh testování, kterou jsem v rámci diplomové práce zapracoval do metodiky MMSP jako novou disciplínu, obsahující úlohy Strategie testování, Analýza testovatelnosti, Analýza rizik, Analýza Báze testování, Analýza a identifikace testovacích podmínek, Plánování testování, Návrh testování, Příprava a tvorba testů, kam patří i tvorba testovacích scénářů, testovacích sad a testovacích dat, a Vyhodnocení a uzavření testování. Rozšíření metodiky MMSP o disciplínu Analýza a návrh testování zahrnuje i zavedení nové role Analytika testování, zapracování dopadu rozšíření na stávající role a definici nových pracovních produktů. V práci popisují i výběr doporučených technik testování a návrhu testování, které je vhodné v roli Analytika testování v rámci disciplíny Analýza a návrh testování použít. Rozšíření metodiky MMSP v oblasti analýzy a návrhu testování respektuje stávající strukturu metodiky a zachovává její životní cyklus projektu, který je rozšířen o úlohy disciplíny Analýza a návrh testování.

Za největší přínos diplomové práce považuji vymezení a charakteristiku disciplíny Analýza a návrh testování a její zahrnutí do testovacího procesu metodiky MMSP, které umožňuje využít tuto disciplínu i na malých projektech vyvíjející nekritická IS/ICT řešení, což doposud nebylo možné. Disciplína Analýza a návrh testování zavádí do metodiky MMSP značné množství analytických činností, které zefektivňují celou oblast testování, zvyšují kvalitu pokrytí testy, testovacích scénářů a testovacích sad. Disciplína Analýza a návrh testování zahrnuje detailní popis doporučených postupů, aktivit a úloh od efektivního stanovení Strategie testování, komplexního vytvoření Plánu testování, vytvoření Báze testování až po Návrh testování. To má za následek včasnější a účinnější odhalování chyb testovaného objektu. Oblast testování softwaru je v současné době složitou a plnohodnotnou disciplínou vývoje IS/ICT řešení a její pozice v této oblasti bude v budoucnu ještě nabírat na významu.

Seznam literatury a zdrojů

BALDUINO, Ricardo. *Introduction to openUP* (online). 2007 (cit. 2015-11-04).

Dostupné z: <https://eclipse.org/epf/general/OpenUP.pdf>

BATH, Graham, Paul JORGENSEN a Jamie MITCHELL. *Certified Tester: Advanced Level Syllabus Technical Test Analyst* (online). ISTQB. 2012, : 51 (cit. 2015-11-05).

BOEHM, Barry. *A Spiral Model of Software Development and Enhancement*. IEEE Press Piscataway, NJ, USA, 1989. ISBN 0-8186-8906-4.

BRUCKNER, Tomáš, VOŘÍŠEK, Jiří, BUCHALCEVOVÁ, Alena a kol., 2012. *Tvorba informačních systémů: principy, metodiky, architektury*. 1. vyd. Praha: Grada, 357 s. Management v informační společnosti. ISBN 978-80-247-4153-6.

BUCHALCEVOVÁ, A. *Návrh metodického rámce IS/ICT*. Doktorská disertační práce. Praha: KIT VŠE, 2004. 172 s.

BUCHALCEVOVÁ, Alena. *Metodiky vývoje a údržby informačních systémů: kategorizace, agilní metodiky, vzory pro návrh metodiky*. 1. vyd. Praha: Grada, 2005, 163 s. Management v informační společnosti. ISBN 80-247-1075-7.

BUCHALCEVOVÁ, Alena. Stav používání agilních metodik v ČR. *Systémová integrace* [online]. 4/2006, 2006(4): 9 [cit. 2015-12-05]. Dostupné z: <http://www.cssi.cz/cssi/stav-pouzivani-agilnich-metodik-v-cr>

BUCHALCEVOVÁ, Alena a Iva STANOVSKÁ, 2013. *Příklady modelů analýzy a návrhu aplikace v UML*. Vyd. 1. Praha: Oeconomica, 197 s. ISBN 978-80-245-1922-7.

CASTB. Certifikace. *CaSTB: Czech and Slovak Testing Board* (online). 2013 (cit. 2015-11-04). Dostupné z: <http://castb.org/cz/certifikace/>

COBIT. ISACA: *Trast in, and value from, information systems* (online). 2015 (cit. 2015-11-05). Dostupné z: <http://www.isaca.org/cobit/pages/default.aspx>

ČERMÁK, Miroslav, 2008. *Clever and smart* (online). (cit. 2015-10-29). Dostupné z:

<http://www.cleverandsmart.cz/>

ČSN ISO/IEC/IEEE 29119-1. *Softwarové a systémové inženýrství – Testování softwaru: Koncepty a definice*. 2014.

ČSN ISO/IEC/IEEE 29119-2. *Softwarové a systémové inženýrství – Testování softwaru: Testovací procesy*. 2014.

Google (online). 2015 (cit. 2015-10-29). Dostupné z: <https://www.google.com>

Google+ (online). 2015 (cit. 2015-10-29). Dostupné z: <https://plus.google.com>

HAVLÍČKOVÁ, Anna, 2014. *Testování softwaru* (online). 1. května 2014 (cit. 2015-10-29). Dostupné z: <http://testovanisoftware.blogspot.cz/>

HLAVA, Tomáš, 2015. *Testování softwaru: Portál zabývající se problematikou ověřování kvality software. Manuální i automatizované testování*. (online). 14.1.2015 (cit. 2015-10-29). Dostupné z: <http://testovanisoftware.cz>

CHLAPEK, Dušan, Václav ŘEPA a Iva STANOVSKÁ. *Analýza a návrh informačních systémů*. 1. vyd. Praha: Oeconomica, 2011, 157 s. ISBN 978-80-245-1782-7.

IBM. *OpenUP* (online). November 30, 2011 (cit. 2015-11-04). Dostupné z: http://kitscm.vse.cz/openup_published/index.htm

ISO/IEC/IEEE 29119-1. *Software Testing: The International Software Testing Standard*. 2013.

ISTQB: *International Software Testing Qualifications Board* (online). 2015 (cit. 2015-10-29). Dostupné z: <http://www.istqb.org>

ISTQB. *What is fundamental test process in software testing?* (online). ISTQB. 2015, : 1 (cit. 2015-11-05). Dostupné z: <http://istqbexamcertification.com/what-is-fundamental-test-process-in-software-testing/>

ISTQB. *What is Test analysis? or How to identify the test conditions?* (online). ISTQB. 2015, : 1 (cit. 2015-11-05). Dostupné z: <http://istqbexamcertification.com/what-is-test-analysis-or-how-to-identify-the-test->

conditions/

KRÁLOVÁ, Iveta. *Metodika testování podle mezinárodních praktik a standardů*. Praha, 2013. Diplomová práce. Vysoká škola ekonomická v Praze. Vedoucí práce Doc. Ing. Alena Buchalceková, Ph.D.

KRUCHTEN, Philippe. *The rational unified process: an introduction* (online). Boston: Addison-Wesley, c2000, xviii, 298 s. (cit. 2015-11-04). Addison-Wesley Object Technology Series. ISBN 02-017-0710-1.

LIBÍK, Jakub. *Zlepšování softwarových procesů v oblasti testování*. Praha, 2014. Diplomová práce. Vysoká škola ekonomická v Praze. Vedoucí práce Doc. Ing. Alena Buchalceková, Ph.D.

MCKAY, Judy, Mike SMITH a Erik VAN VEENENDAAL. *Certified Tester: Advanced Level Syllabus Test Analyst* (online). ISTQB. 2012, : 64 (cit. 2015-11-05).

MOJZISEK, Julius. Pojem analýza. *Slovník cizích slov* (online). 2015 (cit. 2015-11-01). Dostupné z: <http://slovník-cizich-slov.abz.cz/web.php/slovo/analýza>

MOUNTAIN GOAT SOFTWARE. *Scrum Overview for Agile Software Development* (online). 2005 (cit. 2015-11-04). Dostupné z: <https://www.mountaingoatsoftware.com/agile/scrum/overview>

MÜLLER, Thomas a kol. ISTQB. *Certifikovaný tester: Učební osnovy pro základní stupeň* (online). Verze 2011 CZ Beta. 2011 (cit. 2015-11-01). Dostupné z: <http://castb.org/wp-content/uploads/2013/11/ISTQB-CTFL-Syllabus-v2011-CZ-Beta1.pdf>

NÝVLT, David. *Prototypování při vývoji softwaru*. Praha, 2012. Diplomová práce. Vysoká škola ekonomická v Praze. Vedoucí práce Doc. Ing. Alena Buchalceková, Ph.D.

PAGE, Alan, Ken JOHNSTON a Bj ROLLISON, 2009. *Jak testuje software Microsoft*. Vyd. 1. Brno: Computer Press, 384 s. ISBN 978-80-251-2869-5.

(pro)Test! (online). 2013 (cit. 2015-10-29). Dostupné z: <http://www.pro-test.info/>

REID, Stuart. The International Software Testing Standard. *ISO/IEC/IEEE 29119*

Software Testing: The international standard for software testing (online). September 10 2014 (cit. 2015-10-30). Dostupné z:
<http://www.softwaretestingstandard.org/index.php>

REJNKOVÁ, Petra. *Lokalizace a přizpůsobení metodiky OpenUP*. Praha, 2011. Diplomová práce. Vysoká škola ekonomická v Praze. Vedoucí práce Doc. Ing. Alena Buchalceková, Ph.D.

REJNKOVÁ, Petra. *MMSP* (online). 2011 (cit. 2015-11-04). Dostupné z:
<http://mmsp.czweb.org/>

ROSENAU, Milton D. *Řízení projektů*. Vyd. 1. Praha: Computer Press, 2000, xiv, 344 s. Business books (Computer Press). ISBN 80-722-6218-1.

ROYCE, Winston. *Managing the development of large software systems: concepts and techniques*. IEEE Computer Society Press Los Alamitos, CA, USA, 1970. ISBN 0-89791-216-0.

ROUDENSKÝ, Petr a Anna HAVLÍČKOVÁ, 2013. *Řízení kvality softwaru: průvodce testováním*. 1. vyd. Brno: Computer Press, 208 s. ISBN 978-80-251-3816-8.

Seznam.cz (online). 2015 (cit. 2015-10-29). Dostupné z: <https://www.seznam.cz/>

VACHALEC, Vladan. *Testování a kvalita softwaru v metodikách vývoje softwaru*. Praha, 2014. Diplomová práce. Vysoká škola ekonomická v Praze. Vedoucí práce Doc. Ing. Alena Buchalceková, Ph.D.

VŠE. *Elektronické informační zdroje dostupné na VŠE* [online]. 2015 [cit. 2015-11-05]. Dostupné z: <http://www.vse.cz/zdroje/>

WIKIPEDIA. Vodopádový model. *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001-, 10. 12. 2014 [cit. 2015-12-05]. Dostupné z: https://cs.wikipedia.org/wiki/Vodop%C3%A1dov%C3%BD_model

Seznam obrázků

Obrázek 1: Obecné schéma životního cyklu vývoje softwaru (Zdroj: autor)	12
Obrázek 2: Vodopádový model (Royce, 1970).....	17
Obrázek 3: Spirálový model (Boehm, 1989).....	18
Obrázek 4: Schéma iterativního modelu metodiky RUP (Kruchten, 2000)	20
Obrázek 5: Schéma cyklu metodiky Scrum (Mountain Goat Software, 2005)	21
Obrázek 6: Úloha role testera v metodice OpenUP (IBM, 2011).....	23
Obrázek 7: Testovací cyklus metodiky MMSP (Rejnoková, 2011)	25
Obrázek 8: Struktura certifikací ISTQB (ISTQB, 2015).....	28
Obrázek 9: Proces testování organizace (ČSN ISO/IEC/IEEE 29119-2, 2014).....	46
Obrázek 10: Proces plánování testování (ČSN ISO/IEC/IEEE 29119-2, 2014)	48

Seznam tabulek

Tabulka 1: Schéma vztahů role Manažer testování (Zdroj: autor)	54
Tabulka 2: Schéma vztahů role Analytik testování (Zdroj: autor)	55
Tabulka 3: Schéma vztahů role Tester (Zdroj: autor).....	56
Tabulka 4: Doporučené rozvržení členů testovacího týmu (Zdroj: autor).....	59
Tabulka 5: Vztahy Analýzy testovatelnosti (Zdroj: autor)	61
Tabulka 6: Vztahy Analýzy produktových rizik (Zdroj: autor).....	62
Tabulka 7: Vztahy Analýzy Báže testování (Zdroj: autor).....	63
Tabulka 8: Vztahy analýzy a identifikace testovacích podmínek (Zdroj: autor).....	63
Tabulka 9: Vztahy definování Strategie testování (Zdroj: autor).....	64
Tabulka 10: Vztahy Plánování testování (Zdroj: autor)	66
Tabulka 11: Vztahy Návrhu testování (Zdroj: autor)	67
Tabulka 12: Vztahy Přípravy a tvorby testů (Zdroj: autor)	68
Tabulka 13: Vztahy Provedení testování (Zdroj: autor).....	68
Tabulka 14: Vztahy Vyhodnocení a ukončení testování (Zdroj: autor)	69
Tabulka 15: Vztahy pracovního produktu dokument Analýza testovatelnosti (Zdroj: autor)	74
Tabulka 16: Vztahy pracovního produktu dokument Analýza rizik testování (Zdroj: autor)	75
Tabulka 17: Vztahy pracovního produktu Báže testování (Zdroj: autor).....	76
Tabulka 18: Vztahy pracovního produktu Testovací podmínky (Zdroj: autor).....	76
Tabulka 19: Vztahy pracovního produktu Strategie testování (Zdroj: autor).....	77
Tabulka 20: Vztahy pracovního produktu Plán testování (Zdroj: Buchalceková a Stanovská, 2013)	77
Tabulka 21: Vztahy pracovního produktu Návrh testování (Zdroj: autor).....	78
Tabulka 22: Vztahy pracovního produktu Report hodnocení kvality (Zdroj: autor).....	79

Terminologický slovník

Pojem	Popis
Akceptační kritéria	Definice hlavních výstupů, nutných pro naplnění očekávání zákazníka, které vychází z agilního způsobu plánování.
Analýza	Rozbor, nebo metoda zkoumání složitějších skutečností rozkladem na jednodušší, nebo konkrétnější elementy.
Analýza a návrh testování	Nová disciplína zahrnující analytické činnosti v oblasti testování softwaru.
Granularita	Zrnitost. Relativní míra velikosti, nebo podrobnosti rozdělení množiny.
Iterace	Opakování neboli kolo cyklu procesu.
Metodika vývoje SW	Souhrn postupů, pravidel, technik a nástrojů používané pro vývoj softwarového produktu
Projekt	Časově ohraničené úsilí, směřující k vytvoření unikátního produktu nebo služby.
Technika	Souhrn posloupně jdoucích činností a vymezení způsobu jejich použití.