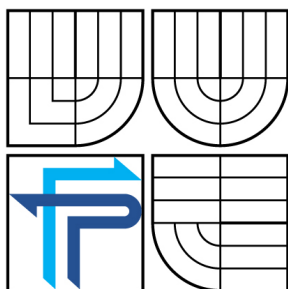




VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA PODNIKATELSKÁ
ÚSTAV MANAGEMENTU

FACULTY OF BUSINESS AND MANAGEMENT
INSTITUTE OF MANAGEMENT

INFORMAČNÍ SYSTÉM PRO SPRÁVU PROJEKTŮ

PROJECT MANAGEMENT INFORMATION SYSTEM

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR

Mgr. MIROSLAV PATOČKA

VEDOUcí PRÁCE
SUPERVISOR

doc. Ing. MILOŠ KOCH, CSc.

BRNO 2010

ZADÁNÍ DIPLOMOVÉ PRÁCE

Patočka Miroslav, Mgr.

Řízení a ekonomika podniku (6208T097)

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

Informační systém pro správu projektů

v anglickém jazyce:

Project Management Information System

Pokyny pro vypracování:

Úvod

Cíle práce, metody a postupy zpracování

Teoretická východiska práce

Analýza problému

Vlastní návrhy řešení

Závěr

Seznam použité literatury

Přílohy

Seznam odborné literatury:

BASL, Josef, BLAŽÍČEK, Roman. Podnikové informační systémy : Podnik v informační společnosti – 2. výrazně přepracované a rozšířené vydání. 2008. vyd. Praha : Grada Publishing, a.s., 2008. 283 s. ISBN 978-80-247-2279-5.

MOLNÁR, Zdeněk. Efektivnost informačních systémů. 1. vyd. Praha : Grada Publishing, spol. s r.o., 2000. 144 s. ISBN 80-7169-410-X.

SODOMKA, Petr. Informační systémy v podnikové praxi. 1. vyd. Brno : Computer Press, a.s., 2006. 351 s. ISBN 80-251-1200-4.

DOSTÁL, Petr, RAIS, Karel, SOJKA, Zdeněk. Pokročilé metody manažerského rozhodování. 1. vyd. Praha : Grada Publishing, a.s., 2005. 168 s. ISBN 80-247-1338-1.

Vedoucí diplomové práce: doc. Ing. Miloš Koch, CSc.

Termín odevzdání diplomové práce je stanoven časovým plánem akademického roku 2009/2010.

L.S.

PhDr. Martina Rašticová, Ph.D.
Ředitel ústavu

doc. RNDr. Anna Putnová, Ph.D., MBA

V Brně, dne 18.04.2010

Abstrakt

Tato práce se zaměřuje na analýzu softwarových nástrojů firmy s podrobným posouzením stavu informačního systému pro správu projektů. Dále obsahuje návrh procesních změn a úprav stávajícího systému. Součástí práce je také kompletní realizace modulu pro výpočet nákladů softwarového projektu. Modul je založen na metodě Use Case Points.

Abstract

The diploma thesis focuses on analysis of software tools used in a softwarehouse company. It contains detailed assessment of a used project management information system, business process design and information system modification. The thesis also includes complete implementation of a module for cost calculation of software projects. The module is based on Use Case Points method.

Klíčová slova

Informační systém pro správu projektů, PMIS, procesní model, BPM, McFarlan, HOS 8, Unified Modeling Language, UML, případy užití, Use Case Points.

Key words

Project Management Information System, PMIS, Business Process Model, BPM, McFarlan, HOS 8, Unified Modeling Language, UML, Use Case, Use Case Points.

Bibliografická citace dle ČSN ISO 690

PATOČKA, M. *Informační systém pro správu projektů*. Brno: Vysoké učení technické v Brně, Fakulta podnikatelská, 2010. 130 s. Vedoucí diplomové práce doc. Ing. Miloš Koch, CSc.

Čestné prohlášení

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

V Brně dne 15. května 2010

.....

Poděkování

Děkuji vedoucímu mé diplomové práce, panu doc. Ing. Miloši Kochovi, CSc., za vedení této práce. Dále bych chtěl poděkovat vedení společnosti Blue Pixel Solutions, a.s. (jmenovitě panu Ing. Pavlu Knappovi) za spolupráci, poskytnuté informace, ochotu při konzultacích a za možnost vykonávat v této firmě diplomovou práci. Poděkování bych rád věnoval také panu Ing. Liboru Krulovi a panu Ing. Vlastimilu Kalužovi za diskuze na téma určování pracnosti a řízení softwarových projektů.

OBSAH

ÚVOD	11
1 TEORIE.....	14
1.1 McFarlanův model	14
1.2 Metoda HOS8.....	14
1.3 Projektový management.....	17
1.3.1 Projektové procesy	17
1.3.2 Trojimperativ	18
1.3.3 Plánování	19
1.3.4 Nástroje.....	20
1.3.5 Příklady softwarových nástrojů pro podporu řízení projektu.....	21
1.3.6 Organizační formy.....	21
1.3.7 Nástroje ke sledování průběhu projektu	22
1.3.8 Změny projektu	24
1.4 Metoda UCP	25
2 BLUE PIXEL SOLUTIONS, A.S.	32
2.1 Popis společnosti.....	32
2.1.1 Organizační struktura firmy	32
2.1.2 Popis strategie a cílů firmy	35
2.2 Hodnocení firmy	36

3	ANALÝZA PROCESŮ	38
3.1	Výrobní procesy	38
3.2	Řídící procesy	42
3.3	Reinženýring procesů	44
4	ANALÝZA IS/IT FIRMY	47
4.1	Analýza McFarlan	47
4.2	Analýza HOS 8	48
5	INFORMAČNÍ SYSTÉM SPRÁVY PROJEKTŮ	52
5.1	Výběr vhodného systému	52
5.1.1	Project management software	53
5.1.2	Hostované systémy	54
5.1.3	Volba vhodného systému pro podporu řízení projektů	55
5.2	Postup návrhu nových funkcí	56
5.2.1	Varianty řešení	57
5.3	Funkce systému Blue Camp	57
5.3.1	Stávající funkcionalita	57
5.3.2	Nově navržená funkcionalita	60
5.4	Hodnotová analýza	63
5.4.1	Párové srovnávání funkcí	63
5.4.2	Ocenění funkcí a výpočet hodnoty pro zákazníka	64
5.5	Inovativní přínos	68

5.6	Analýza rizik	69
5.6.1	Model změny	69
5.6.2	Agent změny	69
5.6.3	Intervence	70
5.6.4	Rizika spojená se změnou	70
5.7	Harmonogram nasazení	75
5.7.1	Alokace pracovníků	75
5.7.2	Alokace prostor a dalších zdrojů	77
5.7.3	Školení	77
6	MODUL UCP	78
6.1	Aplikace metody	78
6.2	Popis modulu UCP	81
6.3	Praktické použití modulu	97
	ZÁVĚR	102
	LITERATURA	107
	SEZNAM OBRÁZKŮ	111
	SEZNAM TABULEK	112
	SEZNAM PŘÍLOH	113

ÚVOD

Cílem této diplomové práce je analýza výrobních a řídicích procesů ve společnosti Blue Pixel Solutions, a.s., která se zabývá vývojem softwarových produktů. Předmětem řešení je identifikace slabých míst a navržení vhodných opatření. Analýza by měla být zaměřena na posouzení stavu softwarových nástrojů určených k podpoře firemních procesů. Za účelem optimalizace firemních procesů by mělo být navrženo nasazení informačního systému, který by eliminoval nalezená slabá místa. Vybraná oblast systému by pak měla být v rámci diplomové práce zrealizována.

Tvorba a zavádění softwarového systému je složitý proces, který s sebou nese nemalá finanční rizika a vyžaduje důkladné plánování. Již na začátku projektu je nutné stanovit náklady, které každý takový projekt obnáší. Při určování nákladů se vychází z údajů o vytvářeném softwarovém produktu. Je potřeba zjistit velikost výsledného produktu, a to často již v okamžiku, kdy je jeho vývoj teprve na začátku. Dle zkušeností se však odhady v úvodních fázích softwarových projektů mohou od skutečnosti lišit až čtyřikrát oběma směry. Přitom nesprávný odhad se při nadhodnocení i podhodnocení projeví ve zvýšených nákladech oproti situaci, kdy je projekt naplánován realisticky.

Odhad pracnosti a ceny vyvíjeného softwaru je tedy jedním z klíčových problémů, kterým se produktový manažer zabývá. Z tohoto důvodu je žádoucí, aby měl projektový manažer k dispozici systém, který by odhady pracnosti pomáhal vytvářet. Součástí této práce je navržení takového systému, který by poskytoval podporu procesům řízení a správy projektů a který by pomáhal vytvářet odhady pracnosti již v úvodních fázích projektu.

Před rozhodnutím nasadit systém pro správu projektů byla provedena analýza procesů firmy s posouzením stavu informačních systémů. Teprve na základě této analýzy byly identifikovány nedostatky v procesech a nástrojích správy projektů, které vedly k rozhodnutí zpracovat v rámci této práce systém pro správu projektů a zejména pak funkcionalitu pro vytváření odhadů pracnosti.

V teoretické části práce jsou popsány metody analýzy informačních systémů McFarlan a HOS 8. Dále je zpracována problematika projektového managementu - zejména procesy plánování a řízení, parametry projektu (tzv. trojimperativ), nástroje projektového řízení a plánování i organizační formy. Všechny tyto aspekty mají vliv na návrh systému pro správu a řízení projektů. V poslední části teoretické pasáže je popsána metoda Use Case Points, která je určena k výpočtům pracnosti softwarových projektů. V rámci popisu jsou shrnuty základní skupiny technik pro odhad velikosti a ceny softwaru, příbuzné metodiky a důvody, proč je v úvodních fázích projektu vhodné používat právě metodu Use Case Points.

V další kapitole je uveden popis společnosti Blue Pixel Solutions, a.s., její strategie a cíle, organizační struktura a celkové hodnocení firmy.

Po zhodnocení firmy jsou v následující kapitole podrobeny analýze její pracovní a řídicí procesy. Je vytvořen procesní model, který identifikuje vazby mezi jednotlivými procesy a zachycuje využití podpůrných softwarových nástrojů. Za účelem optimalizace procesů a zefektivnění podpory procesů pomocí softwarových nástrojů je proveden procesní reinženýring.

Klíčově důležité aplikace jsou v další kapitole identifikovány pomocí analýzy McFarlan. Na tuto analýzu je navázáno analýzou HOS 8, která poskytuje ucelený pohled na informační systém s důrazem na zhodnocení vyváženosti celého systému.

Po analytické části práce následuje část návrhová, která se věnuje definici funkcionalit systému pro správu a řízení projektů. Vyjmenovány jsou nevýhody současného řešení, které by měly být nasazením nového systému odstraněny. Dále jsou uvedeny funkční a technologické vlastnosti, které by měl nový systém splňovat. Tato kapitola obsahuje také výčet vybraných systémů, které jsou na trhu dostupné.

Pro nasazení je po zvážení výhod a nevýhod všech dostupných řešení vybrán systém Blue Camp, který byl vyvinut ve firmě Blue Pixel. Jeho použitelnost je však omezená, protože neobsahuje všechny požadované funkce pro správu a řízení projektů. Označeny tedy jsou ty navrhované funkcionality, které je nutné do tohoto systému naimplementovat.

Na návrh funkcí navazuje jejich hodnotová analýza. Ta by měla určit, které z navržených funkcí mají největší hodnotu pro uživatele a měly by tedy být implementovány přednostně. Upřednostnění některých funkcí vychází z povahy tohoto interního projektu, který bude moci využívat realizačních kapacit, pouze pokud budou uvolněny z hlavních projektů firmy. Realizace tedy nebude kompaktní, ale bude nejspíše probíhat v nepravidelných iteracích.

V kapitole jsou dále pomocí analýzy rizik identifikovány potencionální hrozby, zjištěna míra rizika a navržena vhodná protipatření.

Poslední částí práce je praktická realizace modulu pro výčet pracnosti softwarového projektu. Modul implementuje metodu Use Case Points a je realizován jako webová služba použitelná libovolným externím systémem. Modul má pomoci projektovému manažerovi ve vytváření včasných a přesných odhadů pracnosti projektů.

V samotném závěru práce je modul vyzkoušen na reálném softwarovém projektu. Výsledný odhad pracnosti je srovnán s realitou i s expertními odhady. Následuje zhodnocení výsledků a formulace závěru. Součástí závěru jsou také návrhy na možná rozšíření modulu, která by mohla zajistit vyšší přesnost v odhadech pracnosti.

1 TEORIE

Součástí této práce je analýza IS/IT ve společnosti Blue Pixel Solutions, a.s. Pro úvodní analýzu byl použit McFarlanův model aplikačního portfolia. Sestavení modelu pomáhá ujasnit, které aplikace jsou pro další rozvoj podniku důležité. Na analýzu pomocí McFarlanova modelu bylo navázáno analýzou pomocí metody HOS8. Touto metodou byl zpracován systém pro správu projektů. V následujícím textu je uveden teoretický základ obou metod.

1.1 McFarlanův model

McFarlanův model rozděluje aplikace v podniku do čtyř oblastí podle jejich důležitosti pro zajištění firemních procesů a podle možností, které přinášejí nebo v budoucnu mohou přinést. Zmiňované oblasti odpovídají oblastem v tzv. *Bostonské matici*. Jedná se o:

- Strategické aplikace – odpovídají „*Hvězdám*“ v Bostonské matici. Jde o aplikace, které jsou podmíněné cíly firmy. Přínosné by měly být v budoucnu.
- Potenciální aplikace – odpovídají „*Problémovým dětem*“ v Bostonské matici. Aplikace přímo nevyplývají z firemní strategie, ale mohou být přínosné v budoucnu.
- Klíčové aplikace – odpovídají „*Dojným krávám*“ v Bostonské matici. Jsou nutné pro chod firmy.
- Podpůrné aplikace – odpovídají „*Psům*“ v Bostonské matici. Jde o podpůrné aplikace, které ale nejsou nutné pro současný chod firmy. (12)

1.2 Metoda HOS8

Metoda HOS8 poskytuje ucelený pohled na informační systém s důrazem na zhodnocení vyváženosti celého systému. Systém je hodnocen z pohledu osmi oblastí: hardware, software, orgware, peopleware, dataware, customers, suppliers a management IS. Stav systému v jednotlivých oblastech je odhalován pomocí definovaných kritérií,

které jsou formulovány jako kontrolní otázky. Sady kontrolních otázek jsou uvedeny ve specifikaci metody. (12)

Na stanovené otázky se odpovídá výběrem jedné možnosti z definované škály odpovědí:

Ano – Spíše ano – Částečně – Spíše ne – Ne

Odpovědi jsou ohodnocené číslem od 5 do 1 (nebo obráceně, pokud odpověď „Ne“ odpovídá vysokému stupni stavu dané oblasti).

Vyloučena je otázka s maximálním bodovým ohodnocením odpovědi a otázka s minimálním bodovým ohodnocením odpovědi pro danou oblast. Hodnota stavu oblasti se pak vypočítá jako aritmetický průměr z hodnot zbývajících otázek. Výsledek je zaokrouhlen na celé číslo.

Na základě zjištěných výsledků je určen podrobný stav informačního systému. Model podrobného stavu je ve tvaru:

$m = (u_{hw}, u_{sw}, u_{ow}, u_{pw}, u_{dw}, u_{cu}, u_{su}, u_{ma})$, kde u_i je hodnota stavu pro i-tou oblast.

Z modelu podrobného stavu systému lze odvodit souhrnný stav systému. Souhrnný stav systému odpovídá hodnotě nejnižší složky z modelu podrobného stavu:

$u = \min(u_{hw}, u_{sw}, u_{ow}, u_{pw}, u_{dw}, u_{cu}, u_{su}, u_{ma})$, kde u_i je hodnota stavu pro i-tou oblast.

Na základě modelu podrobného stavu informačního systému a souhrnného stavu informačního systému lze stanovit charakter vyváženosti.

Za zcela vyvážený se považuje takový informační systém, kde všechny zkoumané oblasti vykazují stejné hodnoty stavu. Pokud platí podmínka, že v souboru hodnot stavů oblastí se vyskytnou pouze dvě sousední hodnoty u a $u+1$, kde jedna z hodnot převažuje, pak takový systém označujeme za vyvážený. Pokud informační systém není vyvážený, pak jej označujeme jako nevyvážený.

Matematicky je vyváženost informačního systému definována takto:

Pro všechna u_i platí: $(u_i - u) \leq 1$

a dále platí $\sum_{i=1}^8 (u_i - u) \leq 3$

kde u_i je hodnota stavu i -té oblasti a u je souhrnný stav informačního systému.

Snahou firmy by mělo být udržování informačního systému ve stavu vyváženosti. Současně by však firma měla usilovat také o dosažení takové výše souhrnného stavu, který odpovídá významu informačního systému pro firmu.

Dle metodiky HOS8 je význam informačního systému rozdělen do tří stupňů. Pro každý stupeň je doporučena hodnota souhrnného stavu takto:

- Pokud není zkoumaný systém důležitý pro chod firmy a nepřináší zvýšení produkce, zisku, ani výraznou úsporu pracnosti, pak je doporučeno udržovat souhrnný stav systému na hodnotě 2.
- Pokud je systém pro chod firmy důležitý, ale jeho krátkodobý výpadek neovlivní chod firmy, zisk, nebo spokojenost zákazníků, pak je doporučeno udržovat souhrnný stav systému na hodnotě 3.
- Pokud je systém pro chod firmy klíčově důležitý a krátkodobý výpadek by výrazně ovlivnil chod firmy, zisk či spokojenost zákazníků, pak je doporučeno udržovat souhrnný stav systému na hodnotě 4.

Pro různé kombinace hodnot významu informačního systému a zjištěného souhrnného stavu definuje metoda HOS8 vhodná opatření.

Vedle těchto opatření poskytuje metoda HOS8 i doporučení pro další rozvoj na základě zjištěného souhrnného stavu, významu informačního systému, vyváženosti oblastí a zvolené strategie ve vztahu k systému.

Jednotlivé kombinace hodnot a odpovídající doporučení včetně celkové specifikace metody HOS8 lze nalézt v (12).

1.3 Projektový management

Projektový management je proces plánování, koordinování a řízení úloh a zdrojů pro dosažení definovaného cíle, obvykle za daný čas, s definovanými zdroji a s omezenou cenou.

Projektový management se zabývá řízením projektu. Projekt můžeme definovat jako jedinečnou soustavou činností směřujících k předem stanovenému cíli, který má určitý začátek i konec. Projekt vyžaduje spolupráci různých profesí, váže a spotřebovává jejich kapacity a využívá je pro vytvoření výstupu.

1.3.1 Projektové procesy

Projekty se skládají z procesů. Proces je posloupnost činností, které přinášejí nějaký výsledek.

Projektové procesy lze rozdělit na dvě základní skupiny:

- Procesy řízení projektů – tyto procesy popisují, organizují a vykonávají práci na projektu.
- Produktově orientované procesy – tyto procesy vytvářejí produkt projektu.

1.3.1.1 Procesy řízení projektů

- Inicializační procesy – procesy vedoucí ke vzniku či zahájení procesu nebo fáze projektu.
- Plánovací procesy – procesy definující a upřesňující cíle a způsoby dosažení těchto cílů.
- Realizační procesy – procesy koordinující lidské a další zdroje pro uskutečnění plánu.
- Kontrolní procesy – procesy zajišťující dosahování cílů monitorováním a měřením postupu, zjišťováním odchylek od plánu a inicializací nápravných opatření.
- Závěrečné procesy – procesy formující převzetí projektu. (23)

1.3.2 Trojimperativ

Základní charakteristikou projektů je tzv. *trojimperativ*. Trojimperativ je tvořen třemi základními vzájemně propojenými parametry projektu – specifikací cílů projektu, harmonogramem a náklady.

Uvedené tři parametry trojimperativu jednoznačně definují projekt z hlediska jeho výsledku v těchto rovinách:

- v rovině kvalitativní - specifikace cílů (popis produktu, rozsah díla)
- v rovině časové - harmonogram (termíny)
- v rovině finanční - náklady (náklady na práci, materiální zdroje)

1.3.2.1 Vzájemná závislost parametrů trojimperativu

Každý projekt má určité minimální hodnoty všech parametrů a tím stanovenou minimální plochu trojimperativu. Určité kombinace parametrů specifikace cílů, požadavků na harmonogram projektu a nákladů není možné realizovat.

Druhým projevem vzájemné závislosti je chování trojimperativu při změně hodnoty libovolného jeho parametru. Změna jednoho parametru automaticky vyvolává změnu u ostatních dvou. Například při snížení rozpočtu projektu by měly následovat zásahy do rozsahu cílů (zjednodušením cílů), což povede ke změně harmonogramu projektu.

1.3.2.2 Určení parametrů trojimperativu

Výsledkem procesu plánování projektu je určení parametrů trojimperativu. Plánování zahrnuje kroky, kde jsou jednotlivé parametry trojimperativu vstupem či výstupem.

Obecný a zjednodušený postup plánování projektu je pak následující:

- Výchozím bodem pro plánování je obvykle specifikace výsledného produktu a rozsahu díla (tedy první parametr trojimperativu).

- Prvním krokem plánování je určení postupu realizace projektu – tzn. určení sledu jednotlivých projektových činností, které povedou k vytvoření požadovaného produktu. Postup realizace projektu vychází z obecně uznávaných či standardizovaných postupů v dané oblasti projektu, z předchozích zkušeností z realizace obdobných projektů a ze specifík identifikovaných pro daný projekt.
- Dalším krokem je určení zdrojů projektu. Pro každou činnost stanovenou v postupu realizace projektu je nutné přiřadit kompetentní pracovníky a další materiální zdroje potřebné pro realizaci projektu.
- Výstupem plánování projektu je plán projektu, který na základě specifikace produktu, postupu projektových činností a přiřazených zdrojů, určuje časový harmonogram a náklady realizace projektu (tedy ostatní dva parametry trojimperativu).

Plánování projektu má své hlavní místo na začátku realizace projektu. V průběhu projektu pak dochází k nutnosti změn plánu na základě aktuální situace. Projektový manažer musí sledovat aktuální vývoj projektových činností ve srovnání s plánem projektu a v případě odchylky provést změny plánu tak, aby bylo zajištěno naplnění parametrů trojimperativu.

Podmínky trojimperativu je bohužel obtížné splnit, protože vše, k čemu může během realizace projektu dojít, znamená hrozbu, že nebude dosaženo požadované specifikace provedení a práce na projektu se zpozdí, takže dojde ke skluzu projektu a tím i k překročení plánovaného rozpočtu.

1.3.3 Plánování

Plánování činností je pro řízení projektu klíčové. Plány jsou simulací projektu, obsahují popis toho, jak budou plněny parametry trojimperativu.

Dle jednotlivých dimenzí trojimperativu se plány rozpadají na tři základní oblasti:

- Oblast pro dimenzi provedení – Plány této oblasti se realizují například pomocí diagramů hierarchické struktury činností.

- Oblast pro dimenzi času – Plány této oblasti se realizují pomocí síťových diagramů, seznamu milníků či úsečkových grafů.
- Oblast pro dimenzi nákladů – Plány této oblasti se realizují pomocí finančního rozpočtu. (23)

Plány napomáhají koordinaci a komunikaci, poskytují základ pro sledování průběhu projektu a umožňují vyhnout se problémům. Mnoho projektů se uskutečňuje na základě dlouhodobých plánů organizace. Plány jsou proto často hierarchické, kde krátkodobé plány vznikají v kontextu s dlouhodobými. Takže například plány projektových úkolů jsou pak součástí celkového plánu projektu.

1.3.4 Nástroje

Projektovým manažerům při vytváření plánů všech tří dimenzí trojimperativu pomáhají softwarové nástroje. Tyto nástroje by měly manažerům pomoci nejen při tvorbě plánů, ale také při sledování průběhu prací a při provádění změn plánů probíhajících projektů.

Softwarové produkty pro podporu první dimenze trojimperativu (specifikace cílů) umožňují například vytvářet modely hierarchické struktury činností.

Pro plánování časové dimenze trojimperativu je manažerům doporučováno použití síťových grafů. Vedle síťových grafů umožňují softwarové nástroje vytvářet také úsečkové diagramy a seznamy milníků. Tyto diagramy a seznamy však na rozdíl od síťových grafů samy o sobě neukazují vzájemné závislosti činností.

Při plánování třetí dimenze trojimperativu (náklady) mohou projektoví manažeři použít softwarové nástroje podporující systémy projektového nákladového účetnictví. Jedná se o nástroje vedení a sumarizace nákladů na projekt a projektové činnosti nebo úkoly.

Dvou nebo tří dimenzí trojimperativu se týká vliv omezených zdrojů projektu. Zdroji mohou být lidé (pracovníci) nebo věci (stroje, zařízení, ...). Manažeři přiřazují zdroje k jednotlivým činnostem projektu tak, aby co nejlépe využili jejich kapacitu a zároveň splnili časový plán. Většina softwarových produktů používá pro alokaci zdrojů

tzv. histogramy zdrojů, které znázorňují přiřazení zdrojů k úkolům. Pomocí algoritmů pomáhají tyto softwarové nástroje řešit také problémy s přetížením zdrojů nebo se sestavením časových plánů.

1.3.5 Příklady softwarových nástrojů pro podporu řízení projektu

Mezi nejznámější softwarové systémy pro řízení projektů patří program Microsoft Office Project. Tento program je vhodný zejména v procesu plánování. Umožňuje vytvářet plány časové, alokaci zdrojů a plány finanční. (10)

Pro realizační a kontrolní procesy projektu jsou vhodné systémy typu BaseCamp, BlueCamp, activeCollab apod. Tyto systémy umožňují komunikaci členů týmu přes vzdálený přístup pomocí webového rozhraní aplikace. Pracovníci prostřednictvím systému odevzdávají reporty o uskutečněné práci a hlásí stav projektu. Projektovým manažerům jsou tak k dispozici on-line údaje o průběhu prací a o stavu jednotlivých činností projektu. Díky tomu mohou dynamicky reagovat na vzniklé situace a upravovat podle potřeby plány a alokace, nebo inicializovat opravná opatření. Nastavením kontrolních parametrů lze také zajistit automatizovanou kontrolu průběhu projektu, která v případě nedodržení některého milníku (například nakupením zpožděných úkolů) vygeneruje upozornění pro manažera projektu na vzniklou problematickou situaci.

1.3.6 Organizační formy

Na projekt mají vliv také okolnosti, které projektový manažer přímo ovlivnit nemůže. Jednou z nich je organizační struktura firmy, ve které je projekt realizován. Projekty mají omezený život (od zahájení po ukončení), ale u firem a organizací se předpokládá existence dlouhodobá. Struktura organizace se často nemění a tak se manažer musí při plánování projektu přizpůsobit dané organizační struktuře.

Každá forma organizační struktury má při řízení projektu svá specifika. Existují tři nejběžnější organizační formy:

Funkční organizační forma – Tato forma je zaměřena na trvalé zachování existujících funkčních odborných skupin. Pro projekt může být obtížné překřížit funkční linie a získat potřebné zdroje. Specialisté jednoho oboru jsou soustředěni v jedné

skupině a mezi skupinami navzájem existují bariéry v horizontálním toku informací. Tato organizační forma bývá z hlediska řízení projektů označována jako nejméně vhodná.

Projektová organizační forma – V této formě týmy vznikají především za účelem týmového řešení projektů do předem dané lhůty. Vznikají převážně mimo organizační struktury a většinou z rozhodnutí odpovědného manažera za účelem vyřešení problému nebo úkolu. Patří do skupiny tzv. organizačních struktur s pružnými prvky. Projektová organizační forma je nejvhodnější pro rozsáhlé a dlouhodobé projekty.

Maticová organizační forma – Tato forma organizační struktury kombinuje liniovou organizační strukturu s cílově programovou strukturou řízení. Odborná oddělení zodpovídají za personální zabezpečení a projektoví manažeři zodpovídají za definování práce a za vytvoření reálného plánu. Je nutné, aby se manažeři projektů a manažeři odborných oddělení společně domluvili na úkolech, cílech a časových plánech projektů. Tato struktura je pokládána za vhodnou pro větší organizace s větším množstvím projektů. Pro malé organizace je tato struktura drahá, protože vyžaduje jeden řídicí útvar navíc (úvar projektového managementu). Řízení maticové struktury může být složité, pokud nejsou jasně definovány role a zodpovědnosti. (27)

Softwarové nástroje, které se používají pro řízení projektů, by měly zohlednit tyto základní struktury organizací. Zejména pro maticovou organizační strukturu by mělo být pamatováno na možnost rozdělení uživatelských práv mezi manažery projektů a manažery odborných oddělení.

Vedle těchto základních organizačních forem se objevují ještě další formy jako například: pseudomaticová forma, vložená riziková organizace, zvláštní projektová skupina, projektové týmy ad hoc apod.

1.3.7 Nástroje ke sledování průběhu projektu

Důležitou řídicí činností manažera projektu je monitorovací (kontrolní) proces. Kontroly jsou nutné k sledování skutečného postupu prací na projektu oproti plánu projektu. Kontroly celkově slouží k odhalení odchylek od stanoveného plánu

trojimperativu. Existuje několik technik kontroly projektu. Jedním z kontrolních nástrojů může být například pozastavení zdrojů nebo práva nakládat s finančními prostředky podle vlastního uvážení. Projektoví pracovníci pak musejí o každý vyšší výdaj požádat a manažer si tak udržuje povědomí o čerpaných zdrojích. Tento typ omezujících kontrol se hodí pro nezkušené týmy nebo obtížné projekty. Obecně lze doporučit u malých projektů. Pokud by i u velkých projektů všechna rozhodnutí o čerpání zdrojů procházela přes projektového manažera, pak by to znamenalo znatelné zdržování projektu.

Nejlepším způsobem kontroly je zjišťování stavu projektových úkolů. Manažer projektu zjišťuje stav prací prováděných projektovým týmem. Informace o průběhu činností lze získat čtením zpráv a organizováním kontrolních schůzek. Tento druh kontroly staví na teorii, že se pracovníci budou snažit odvést dobrou práci.

Zprávy, které pracovníci zasílají, se dělí do tří základních kategorií:

- Zprávy týkající se výsledků v oblasti provedení.
- Zprávy týkající se časového postupu prací.
- Zprávy týkající se nákladů. (23)

Zprávy mohou být odevzdávány ve formě souhrnů. Tímto způsobem se získá celkový přehled o stavu projektu. Zprávy také mohou informovat o jednotlivých činnostech.

Manažer projektu se nemůže zcela spoléhat na předané zprávy a výkazy práce, protože tyto údaje mohou být nepřesné. Pracovníci, kteří vytvářejí reporty, mohou mít sklon k přehnanému optimismu. Stav prací mohou požadovat za téměř hotový, avšak situace může být taková, že stav projektu zdaleka neodpovídá jejich představě. Za splněný úkol lze považovat totiž jen takový, který je opravdu ukončen. U netriviálních úkolů je obtížné zjistit, jaké procento úkolu je skutečně splněno.

Procento dokončenosti je pro sledování postupu prací nespolehlivé. Při hlášení úrovně rozpracovanosti mohou pracovníci použít jeden z následujících dvou přístupů:

- 80 % úkolů je na 100 % splněno – Zde se předpokládá, že všechny úkoly jsou známe a přibližně stejné.
- 100 % úkolů je na 80 % splněno – Zde se předpokládá, že zbývající části úkolů jsou stejně pracné jako hotové části. (23)

Obvykle se procento rozpracovanosti uvádí na základě množství spotřebovaného času a prostředků. Ani spotřeba času a ani spotřeba prostředků však nejsou měřítkem postupu.

K přesnějšímu zjištění stavu projektu napomáhá rozdělení projektu na malé úkoly. Tím se snižuje nejistota ohledně skutečného stavu prací na projektu. Čím menší úkoly jsou, tím více existuje kontrolních bodů. Za splněný úkol se považuje ten, který je ze 100 % hotový. To usnadňuje projektovému manažerovi práci při přijímání splněných úkolů. Rozdělení projektu na malé úkoly je výhodné také proto, že je v jednotlivých úkolech obsaženo menší riziko, lze lépe vidět cíl úkolu, je dříve upozorněno na problémy a kratší termín dokončení vyvolává větší pocit naléhavosti.

1.3.8 Změny projektu

Odchytky od plánů projektu jsou zcela běžné. U menších projektů je zpravidla dosahováno přesnějších plánů a tedy menších odchylek.

Jednou z příčin překračování termínů a nákladů je to, že v zadání projektu dochází ke změnám. Tyto změny mohou přicházet zvenčí (od zákazníka). Změny mohou být vynuceny nepřímo. V průběhu realizace projektu se mohou změnit právní předpisy, růst inflace může překročit plánovanou míru apod.

Nepřesnost odhadů je dána řadou faktorů. Nejčastěji se setkáváme s nedokonalou definicí rozsahu projektu. Tuto chybu může zapříčinit jak zadavatel, tak i realizátor. U mnoha manažerů se také objevuje cenové podbízení, kterým se snaží úmyslně nabídnout nižší cenu za účelem zisku zakázky. Manažer bere v úvahu optimistické odhady a nevytváří rezervy. To má důsledek v podobě podhodnocení ceny a tlaku na snížení termínů, což vede k odchylkám v plnění časového plánu. K odchylkám dochází také při výměně projektového manažera během realizace projektu. Další příčinou vzniku odchylky může být forma zadávání úkolu. Pokud je úkol zadáván ústně, pak

roste pravděpodobnost nedorozumění. Úkoly by tedy měly být zadávány písemně. K písemné formě zadávání úkolů je vhodné použít některý ze softwarových systémů pro vedení projektu. Tyto systémy umožňují rychlé a pohodlné zadávání a akceptování úkolů.

Pro snazší přijímání změn by měl pán projektu v každé dimenzi trojimperativu obsahovat rezervu. Všichni členové týmu by měli být o vzniklých změnách plánu informováni.

1.4 Metoda UCP

Use Case Points (UCP) je označením metody pro odhad pracnosti vývoje softwaru. Informační systém pro řízení projektů, který je v této diplomové práci zpracováván, slouží primárně ve společnosti vyvíjející software na zakázku. Odhad pracnosti a ceny vyvíjeného softwaru je jedním z klíčových problémů, kterým se produktový manažer v této společnosti zabývá. Z tohoto důvodu je žádoucí mít v informačním systému integrován modul, který by odhady pracnosti pomáhal vytvářet. Modul realizovaný v této práci byl postaven právě na algoritmu metody UCP.

Zavádění nového softwarového systému je složitý proces, který s sebou nese nemalá finanční rizika a vyžaduje důkladné plánování. Již na startu projektu je nutné stanovit náklady, které každý takový projekt s sebou přináší. Při odhadování nákladů projektu, na jejichž základě se pak vytváří rozpočet a plán prací, je zapotřebí vycházet i z údajů o softwarovém produktu, který je vytvářen. Je potřeba zjistit velikost výsledného produktu, a to často již v okamžiku, kdy je jeho vývoj na začátku.

Existují tři základní skupiny technik pro odhad velikosti a ceny softwaru (29):

- Expertní odhady
- Analogie
- Odhady založené na modelu

Technika expertních odhadů vychází z odhadu provedeného expertem na základě předchozích zkušeností. Expertní odhad je užitečný tam, kde nemáme měřitelná

data, ze kterých by bylo možné vycházet. Kvalita odhadu však závisí na kvalitě experta. Objektivitu odhadů můžeme zvyšovat počtem expertů.

Technika analogie staví na odhalení rozdílů a podobností se srovnatelnými projekty a na odhadu vlivu těchto odlišností na pracnost nového projektu. Nevýhodou analogie je fakt, že musíme mít data o obdobných projektech.

Techniky založené na modelu staví na závislostech mezi velikostí projektů a pracností. Závislosti bylo možné odvodit díky velkému množství dat o skutečných projektech a jejich charakteristikách. Výpočty ovlivňuje mnoho faktorů, které jsou zadávány subjektivně v definované škále.

Určení nákladů na vyvíjený software založené na analýze modelů můžeme provádět pomocí několika různých metodik. Lze využít měření velikosti zdrojového kódu (*COCOMO – Constructive Cost Model*), funkčně orientované metriky vycházejících z principů analýzy funkčních bodů (*FPA – Function Point Analysis*) nebo metriky založené na analýze případů užití (*UCP – Use Case Points*).

Pokud jsou dostupné informace o velikosti zdrojových kódů budoucího produktu (což znamená, že řešitelé projektu jsou schopni tuto velikost odhadnout), pak lze pro výpočet nákladů použít metodu COCOMO. Tato metoda byla poprvé publikována v roce 1981 a jejím autorem je Barry Boehm. Metoda byla v průběhu času několikrát modifikována a měla by tedy být použitelná i pro současné projekty. Model COCOMO je ale špatně použitelný v situaci, kdy vývojáři nejsou schopni dostatečně přesně odhadnout velikost zdrojových souborů (19),(27).

FPA má oproti COCOMO výhodu v tom, že jako měřítko složitosti používá funkční body a nikoli počet řádků zdrojového kódu. Funkční body vycházejí z funkcionality softwaru, která bývá známa již v úvodních fázích projektu. „*S trochou nadsázky lze říct, že funkční body představují jednotku výroby při procesu tvorby softwaru. Tímto způsobem lze ohodnocovat pracnost vývoje nejen nových softwarových produktů, ale i pracnost modifikace již existujících systémů*“ (19). Získat ale spolehlivě počet funkčních bodů není triviální, je to možné až ve fázi detailního návrhu. V této fázi je už ale zpravidla uzavřena smlouva a stanovena cena projektu.

Zatímco zpracování funkčních bodů probíhá až ve fázi detailního návrhu, existují tzv. případy užití (funkcionalita systému z pohledu uživatele), které jsou zmapovány již ve fázi analýzy. Analýza může být pojata jako samostatný projekt, který předchází podpisu smlouvy na vlastní realizaci vytvářeného softwarového systému. Pokud bychom byli schopni určit na základě případů užití celkovou pracnost vyvíjeného systému, pak jsme také schopni určit cenu systému ještě před podpisem smlouvy.

Odhadem pracnosti na základě případů užití se zabývá metoda Use Case Points. Tuto metodu v roce 1993 vyvinul Gustav Karner. Jednalo se o diplomovou práci na švédské University of Linköping.

Případy užití, tak jak jsou definovány v jazyce UML, popisují činnosti, které budou uživatelé se systémem provádět (11). Vedle samotných případů užití obsahuje model také tzv. aktéry. Aktérem je osoba nebo jiný systém, který provádí činnost popsanou v případě užití. Na aktérech a případech užití je postaven algoritmus metody UCP.

V prvním kroku algoritmu dojde k ohodnocení jednotlivých aktérů a případů užití. Aktéři i případy užití se zařazují do kategorií složitosti. Aktéři v závislosti na rozhraní, přes které se systémem komunikují, spadají do těchto kategorií:

- *Simple* – externí programy, které komunikují se systémem přes API.
- *Average* – externí programy, které komunikují se systémem přes protokol (např. HTTP, FTP, ...), nebo uživatelé, kteří systém používají přes příkazový řádek.
- *Complex* – uživatelé, kteří pro komunikaci se systémem používají grafické uživatelské rozhraní.

Podobně jako aktéry ohodnotíme i případy užití. U případů užití se rozhodujeme podle množství transakcí. Každý případ užití je popsán scénářem, který se skládá z transakcí. Určení transakcí patří k nejobtížnější části výpočtu UCP. Transakce je definována jako množina aktivit ve scénáři případu užití, která je buď provedena celá a nebo není provedena vůbec. Díky této poněkud vágní definici nemusí být identifikace transakcí jednoznačná. Někteří autoři uvádějí, že transakce je možné odvodit z počtu

kroků ve scénáři (3). Jiní autoři s tímto tvrzením nesouhlasí a transakcí označují jen takovou množinu aktivit, která vytváří nějakou hodnotu. Například kliknutí na tlačítko není transakcí, ale uložení článku do systému už transakcí je (16),(13).

Pokud jsou identifikovány a spočítány transakce, lze případy užití rozdělit do kategorií složitosti:

- *Simple* – případ užití obsahuje 1 až 3 transakce.
- *Average* – případ užití obsahuje 4 až 7 transakcí.
- *Complex* – případ užití obsahuje více než 7 transakcí.

Podle kategorií složitosti přiřazujeme případům užití a aktérům váhu. Výpočet vah UUCW (*Unadjusted Use Case Weights*) a UAW (*Unadjusted Actors Weights*) provádíme dle uvedených vztahů:

$$UAW = 1 \cdot A_{simple} + 2 \cdot A_{average} + 3 \cdot A_{complex}$$

$$UUCW = 5 \cdot U_{simple} + 10 \cdot U_{average} + 15 \cdot U_{complex}$$

kde

A_i je počet aktérů patřících do kategorie složitosti i .

U_i je počet případů užití patřících do kategorie složitosti i .

Vypočítané váhy se sečtou, tím získáme proměnnou UUCP (*Unadjusted Use Case Points*):

$$UUCP = UAW + UUCW$$

Metoda UCP používá obdobné modifikační faktory jako metoda funkčních bodů. Jedná se o technické faktory (*TCF – Technical Complexity Factors*) a faktory prostředí (*EF – Environmental Factors*). Technických faktorů je celkem třináct, faktorů prostředí je osm. Popisy faktorů a definované váhy faktorů jsou uvedeny v tabulce Tab. 1 (14).

Tab. 1: Technické faktory a faktory prostředí

	Popis technických faktorů	Váha		Popis faktorů prostředí	Váha
T1	Distribuovaný systém	2,0	E1	Znalost problému, technologie	1,5
T2	Výkonnost	2,0	E2	Zkušenosti s aplikací	0,5
T3	Efektivita pro uživatele	1,0	E3	Zkušenosti s OOP	1,0
T4	Složitost procesů	1,0	E4	Analytická znalost domény	0,5
T5	Znovupoužitelnost kódů	1,0	E5	Motivace týmu	1,0
T6	Jednoduchost instalace	0,5	E6	Stabilní požadavky	2,0
T7	Jednoduchost užívání	0,5	E7	Externí experti	-1,0
T8	Přenositelnost	2,0	E8	Složitost prog. jazyku	-1,0
T9	Modifikovatelnost	1,0			
T10	Víceuživatelský přístup	1,0			
T11	Bezpečnost	1,0			
T12	Využití zdrojů třetí strany	1,0			
T13	Nutnost zaškolení	1,0			

Každý z faktorů ohodnotíme stupněm od 0 do 5. Pokud faktor nemá v projektu žádný vliv, je ohodnocen stupněm 0. Pokud má faktor na projekt vysoký vliv, je ohodnocen stupněm 5.

Váhy faktorů a stupně vlivu faktorů na projekt se použijí pro výpočet hodnot TCF a EF dle vztahů:

$$TCF = 0,6 + \left(0,01 \cdot \sum_{i=1}^{13} (TCF_váha_i + TCF_vliv_i) \right)$$

$$EF = 1,4 + \left(-0,03 \cdot \sum_{i=1}^8 (EF_váha_i + EF_vliv_i) \right)$$

kde

$TCF_váha_i$ je váha i -tého technického faktoru.

TCF_vliv_i je stupeň vlivu i -tého technického faktoru na projekt.

$EF_váha_i$ je váha i -tého faktoru prostředí.

EF_vliv_i je stupeň vlivu i -tého faktoru prostředí na projekt.

Dalším krokem je zjištění počtu *use case points* (UCP). Výpočet provedeme dle vzorce:

$$UCP = UUCP \cdot TCF \cdot EF$$

Výpočtem UCP jsme odhadli velikost připravovaného softwarového systému. To, co chceme znát, je pracnost potřebná k vytvoření tohoto systému. Použijeme tedy počet UCP ke zjištění pracnosti.

K výpočtu pracnosti je zapotřebí znát ještě faktor produktivity (*PF – Productivity Factor*). Faktor produktivity se uvádí v *člověkohodinách*¹ na jeden *use case point* a představuje množství práce, které je zapotřebí k realizaci jednoho *use case pointu*. Nejlepším způsobem, jak zjistit faktor produktivity, je sestavit statistiku stávajících projektů realizovaných ve vlastní společnosti. Z historických dat jsme schopni zjistit, kolik člověkohodin jsme průměrně spotřebovali a z toho odvodíme požadovaný faktor produktivity vztažený k vlastnímu vývojovému týmu.

Pokud nemáme historická data k dispozici, můžeme vyjít z doporučení Gustava Karnera (autora metody UCP) a za faktor produktivity dosadit 20 *člověkohodin/UCP*. Jiní autoři doporučují pro malý projekt (bez potřeby rozsáhlé dokumentace a s převážně stabilními požadavky) dosadit 15 *člověkohodin/UCP*. Pro střední (ne příliš komplexní) systémy doporučují 20 *člověkohodin/UCP* a pro komplexní projekty (s potřebou rozsáhlé dokumentace) pak doporučují 36 *člověkohodin/UCP*. (16)

Podle autorů Schneidera a Winterse (25) lze vypočítat faktor produktivity z faktorů prostředí EF. Vycházíme ze stupně vlivu, který jsme přiřadili faktorů E1 až E8. Sečteme stupně faktorů E1 až E6, které jsou nižší než 3. K získanému výsledku přičteme stupně faktorů E7 a E8, které jsou vyšší než 3.

¹ Pojmem *člověkohodina* je označena hodina lidské práce. Zkráceně též jako *čh*.(26)

Nechť X je výsledná hodnota předchozího součtu, pak určení faktoru produktivity provedeme dle následujících pravidel:

- jestliže $0 \leq X \leq 2$, pak $PF = 20 \text{ člověkohodin} / UCP$
- jestliže $3 \leq X \leq 4$, pak $PF = 28 \text{ člověkohodin} / UCP$
- jestliže $4 < X$, pak $PF = 36 \text{ člověkohodin} / UCP$

Pokud je X větší než 4, pak se doporučuje provést změny v projektu nebo v týmu (22).

Jakmile máme faktor produktivity určený, můžeme vypočítat pracnost projektu podle vztahu:

$$\text{Pracnost} = UCP \cdot PF$$

Tímto jsme získali pracnost projektu vyjádřenou v člověkohodinách. Přesnost odhadu je dána především přesností, s jakou jsme určili množství transakcí v případech užití, stupně faktorů TCF, EF a faktor produktivity našeho týmu. Pro přesnější odhady se doporučuje provést analýzu předchozích projektů, ze kterých můžeme odvodit jednak faktor produktivity, ale také hodnoty pro kalibraci vah technických faktorů a faktorů okolí (16),(6).

Metoda Use Case Points stále není standardizována a neexistují ani formální standardy pro psaní případů užití. Problémem je především úroveň abstrakce a úroveň podrobnosti popisu scénářů (29).

Podle e-zinu The Rational Edge je metoda UCP používána s dobrými výsledky ve společnostech Sun a IBM (18). K výhodám patří jednoduchost metody, která pomáhá překonat psychologický odpor k provádění odhadů komplikovanými metodami. Dle studie (2) byl odhad metodou UCP blíže skutečné pracnosti než odhad provedený skupinou expertů.

K dalším výhodám patří možnost provádět odhady v úvodních fázích vývoje softwarového systému (například už při sjednávání smlouvy). I přes uvedené výhody je však doporučováno kombinovat metodu UCP s jinými metodami (expertní odhad, analogie s podobnými projekty, aj.) (29).

2 BLUE PIXEL SOLUTIONS, A.S.

2.1 Popis společnosti

Firma Blue Pixel Solutions, a.s. je akciovou společností, která zaměstnává přibližně 50 zaměstnanců. Firma se zabývá vývojem softwarových systémů na zakázku. Společnost s tímto typem výrobní činnosti je v oblasti IT zpravidla nazývána jako „softwarehouse“. Firma se zaměřuje především na tvorbu informačních a obchodních systémů. Díky silnému analytickému oddělení je schopna produkovat informační systémy pro různé obory podnikání (doprava, finance, elektronický obchod, telekomunikace apod.).

Dlouholetá praxe v oboru informačních technologií, provozování internetových obchodů, projektového a produktového managementu, marketingu, vedení obchodního týmu či zkušenosti ve vydavatelské oblasti umožňují Blue Pixelu nabízet také poradenské služby. Tyto se specializují na efektivní využití Internetu a informačních technologií pro podnikání.

Vedle poradenství a vývoje softwarových systémů nabízí Blue Pixel své služby také pomocí outsourcingu. Jedná se především o služby z oblasti provozu a správy informačních technologií, grafických služeb a webdesignu, správy a publikace informací pomocí webových portálů, marketingu a komunikace se zákazníky, programátorských a analytických činností.

2.1.1 Organizační struktura firmy

Vedení – Vedení firmy se zabývá strategickým řízením společnosti. Zajišťuje komunikaci s klíčovými partnery firmy a řídí významné firemní projekty.

Finanční a personální oddělení - Finanční oddělení má ve své kompetenci finanční řízení firmy, účetnictví a vztah se státní správou. Personální část oddělení má ve své kompetenci výběr a nábor nových zaměstnanců. Do oddělení spadá také personální a mzdová agenda.

Stav oddělení: 3 zaměstnanci (vzdělání středoškolské a vysokoškolské magisterské v oborech zaměřených na ekonomii a management)

Obchodní oddělení – Obchodní oddělení zajišťuje komunikaci mezi firmou a zákazníky, vyhledává nové zákazníky, prodává hotové produkty a nabízí nová řešení na zakázku. Obchodní oddělení také zodpovídá za marketing firmy. Do oddělení patří obchodní manažeři a oddělení vede obchodní ředitel.

Stav oddělení: 4 zaměstnanci (vzdělání středoškolské, vysokoškolské bakalářské nebo vysokoškolské magisterské v oborech zaměřených na ekonomii a management)

Oddělení řízení projektů – Oddělení řízení projektů plánuje a řídí výrobní projekty ve firmě. Do oddělení řízení projektů spadají projektoví manažeři a oddělení řídí projektový vedoucí. Projektoví manažeři přidělují práci pracovníkům, kteří jsou alokováni v projektech. Průběh prací je následně manažery řízen a kontrolován. Manažeři projektů také provádějí analýzu rizik, vytvářejí časové plány a odhadují pracnost projektů. Mezi kompetence manažerů patří i řízení komunikace mezi zákazníkem a firmou.

Stav oddělení: 4 zaměstnanci (vzdělání vysokoškolské magisterské v ekonomicky nebo technicky zaměřených oborech)

Analytické oddělení – Analytické oddělení zpracovává požadavky zákazníků na vyvíjený systém, poskytuje poradenské a konzultační činnosti, analýzu a modelování firemních procesů apod. Požadavky na systém analytické oddělení průběžně zpracovává do prvotních modelů systému, které slouží jako podklad pro akceptační dokumenty a jako podklad pro vývojové oddělení. V průběhu vývoje systému poskytuje analytické oddělení podporu vývojářům. Do oddělení patří analytici a řídí jej vedoucí analytického oddělení.

Stav oddělení: 5 zaměstnanců (vzdělání vysokoškolské v oboru informačních technologií)

Oddělení vývoje – Oddělení vývoje je největším oddělením firmy a proto se dále dělí do jednotlivých týmů dle hlavní technologie, kterou tým při vývoji používá.

Oddělení vývoje řídí ředitel vývoje a jednotlivé týmy pak řídí týmoví vedoucí. Do oddělení patří systémoví architekti, návrháři a vývojáři. Architekti dle zpracované analýzy navrhnu technické řešení daného systému, které je pak podle navrhnuté technologie přiděleno konkrétnímu vývojovému týmu. Vývojáři jednotlivých týmů implementují zvolené řešení a vytvářejí funkční softwarový systém.

- Tým vývoje .NET
- Tým vývoje Java/MIDP
- Tým vývoje PHP
- Tým vývoje HTML
- Tým vývoje grafiky

Stav oddělení: 24 zaměstnanců (vzdělání vysokoškolské bakalářské, vysokoškolské magisterské nebo studium vysokoškolské magisterské a doktorské v oboru informačních technologií)

Testovací oddělení – Testovací oddělení se stará o průběžné a závěrečné (akceptační) testování vyvíjených softwarových systémů. Oddělení má za úkol odhalovat chyby v softwaru a reportovat je zpět vývojářům. Testy jsou prováděny dle testovacích scénářů sestavených testovacím oddělením podle analýzy a návrhu testovaného systému. Do testovacího oddělení patří testeři a oddělení řídí vedoucí testovacího oddělení.

Stav oddělení: 6 zaměstnanců (vzdělání středoškolské a vysokoškolské bakalářské technicky zaměřených oborů)

Oddělení podpory IT - Oddělení podpory IT řeší případné technické problémy a zajišťuje servis zákazníkům. Oddělení připravuje veškeré hardwarové a základní softwarové vybavení pro provoz systémů vyvíjených firmou. Oddělení se také stará o údržbu a konfiguraci výpočetních zařízení firmy. Do oddělení spadají technici IT, které řídí vedoucí oddělení podpory IT.

Stav oddělení: 4 zaměstnanci (vzdělání středoškolské, vysokoškolské bakalářské nebo studium vysokoškolské magisterské v oboru informačních technologií)

Společnost Blue Pixel v řízení používá maticovou strukturu. Každý pracovník je vystaven dvojí podřízenosti. Nadřízeným je mu vedoucí oddělení a také projektový manažer, který řídí projekty, kterých se pracovník účastní.

2.1.2 Popis strategie a cílů firmy

Strategickým cílem společnosti je upevnění pozice na trhu informačních systémů. Dalším cílem je zisk většího podílu na zakázkách státní správy a samosprávy. Do budoucna se uvažuje také o posílení podílu outsourcingu ve vývoji a to jak v oblasti poskytování outsourcingových služeb tak i ve využití outsourcingu v podobě delegování některých pracovních procesů externím outsourcingovým společností.

Interně se firma snaží o zavedení sofistikovanějších metodik řízení a vývoje. Cílem je vybrat, pro své účely upravit a přijmout některou z osvědčených metodik používaných pro vývoj informačních systémů a pro řízení projektů v oblasti informačních technologií.

V oblasti technologické firma plánuje změnu orientace z „*desktopově*“ orientovaných aplikací, tzv. *tlustých klientů*, na aplikace webové a portálové přístupné pomocí *tenkých klientů*. V souladu s trendy v oblasti IT se pak firma bude u vhodných projektů snažit také o přechod na servisně orientovanou architekturu (SOA). S tím souvisí i nová oblast poskytování služeb a to analýza, modelování a reinženýring obchodních procesů.

Při zvýšení počtu zaměstnanců plánuje firma otevření nové pobočky v jiném městě. V úvahu přichází také pobočka v jiném státě (pravděpodobně na Slovensku). Klíčovou podmínkou pro výběr místa je levnější pracovní síla v dané oblasti a zároveň dostatek kvalifikovaných potencionálních zaměstnanců. Jako vhodná se tedy jeví ta města, ve kterých se nacházejí vysoké a střední školy se zaměřením na IT. Nová pobočka by měla být určena především pro vývojáře a testery. Pro tyto účely lze pronajmout nebo koupit i méně reprezentativní a levnější prostory na okraji města.

2.2 Hodnocení firmy

Ve společnosti Blue Pixel jsem pomocí SWOT analýzy identifikoval silné a slabé stránky této firmy a související příležitosti a hrozby. Výsledky SWOT analýzy jsou uvedeny v tabulce Tab. 2.

Tab. 2: Výsledky SWOT analýzy

Silné stránky	Slabé stránky
- Díky své velikosti je firma vcelku dynamická a je schopna rychle reagovat na změny trhu. - Firma má silné analytické oddělení a kvalitní vývojářský tým. - Poloha sídla firmy je výhodná (v místě je dostatek potencionálních kvalifikovaných zaměstnanců) a firma je dobře dostupná zákazníkům. - Firma je schopna realizovat širokou škálu softwarových řešení a při vývoji není omezena jedním programovacím jazykem nebo jednou technologií ale díky skladbě týmů dokáže vyvíjet pomocí všech nejčastěji používaných programovacích jazyků a technologií a tím se plně přizpůsobit potřebám zákazníka.	- Slabší stránku firmy tvoří obchodní a testovací oddělení. - Firma se restrukturalizuje po předchozím rychlém růstu, nejsou ještě stanoveny jasné metodiky vývoje a vytvářejí se postupně „za chodu“. - Nedostatečné využívání moderních technologií snižuje efektivitu vývoje a znesnadňuje správu realizovaných systémů.
Příležitosti	Hrozby
- Firma je schopna vytvářet stejně kvalitní softwarové systémy levněji než větší konkurenční firmy. - Pokud není sama schopna získat například státní zakázky, pak se může na jejich realizaci podílet jako subdodavatel nebo pomocí outsourcingu. - Silné analytické a vývojářské oddělení může přinášet nová a originální řešení, která by zaplnila mezery na trhu.	- V oblasti vývoje softwaru existuje silná konkurence a hrozby plynou zejména od velkých softwarových společností vyvíjejících „krabicová“ řešení nebo poskytujících software například jako SaaS (Software as a Service). - Mezi zaměstnanci v oblasti IT je vysoká fluktuace, hrozí ztráta loajality a přenos know-how do jiných společností. - Odchod některých zaměstnanců může také výrazně ohrozit průběh realizovaných projektů.

Dále jsem provedl analýzu charakteristických rysů organizační kultury podporující kvalitu. Hodnoty jednotlivých charakteristik jsou uvedeny v příloze č. 1. V následujícím textu jsou uvedeny závěry, které z této analýzy vyplývají.

Společnost působící v oblasti IT se musí rychle vyvíjet a neustále zlepšovat procesy výroby. Tohoto rysu firma Blue Pixel dosahuje, ale na druhou stranu některé změny (rychlý růst firmy) jsou zvládány s jistými obtížemi. Týmová spolupráce je na vysoké úrovni, ale i zde jsou patrné dopady problémů plynoucích z rychlého růstu. Analytické přístupy jsou v Blue Pixelu na vysoké úrovni.

Pracovníci jsou vesměs samostatní a odpovědní. Někdy se tato samostatnost negativně projevuje ve vztahu k týmové spolupráci. Nízké hodnocení má loajalita pracovníků. Souvisí s vysokou fluktuací zaměstnanců a tedy nedostatečnou dobou k vytvoření silnějších pout mezi zaměstnanci a firmou. Vysoká fluktuace má také vliv na vysoké hodnocení odpovědnosti pracovníka k vlastnímu profesnímu růstu a nižší hodnocení odpovědnosti pracovníka k společnosti a organizaci.

Účinné jednání a komunikační politika managementu má také nižší hodnocení. Bodovou ztrátu způsobily zejména nedostatky v informačních systémech řízení, komunikace a spolupráce na projektech. Více než v oblasti software jsou tyto nedostatky patrné v oblasti orgware (v pravidlech, kdo a jak má dané systémy používat).

Ve firmě také není vytvořena strategie soustavného a uceleného dalšího vzdělávání zaměstnanců. Vysoce je hodnocena schopnost organizace dosahovat kvality a orientace na spokojenost zákazníka. Na druhou stranu je zejména u starších projektů zřetelná nekvalita poškozující výkonnost organizace. Vysoce jsou dále hodnoceny systémy formální a osobní kontroly.

V hodnocení některých rysů jsou patrné důsledky rychlého růstu firmy a nutnosti neustálých změn. Zaměstnanci jsou vesměs vysoce kvalifikovaní a motivovaní, ale jejich motivace spočívá hlavně v osobním profesním růstu, což se projevuje i ve vztahu nižší loajality k firmě. Rostoucí firma také vyžaduje změny v komunikační politice a v řízení projektů.

3 ANALÝZA PROCESŮ

Pod pojmem procesní analýza chápeme komplexní vyhodnocení procesů probíhajících v organizaci. Součástí procesní analýzy je vytvoření procesního modelu. Procesní model umožňuje posoudit stav procesů, navrhnout nové, zlepšit stávající a nebo odstranit nepotřebné procesy. Úpravy mohou mít dopad i na informační systémy a aplikace, které příslušné procesy podporují. Analýza a úpravy procesů by měly vést ke zlepšenému fungování firmy.

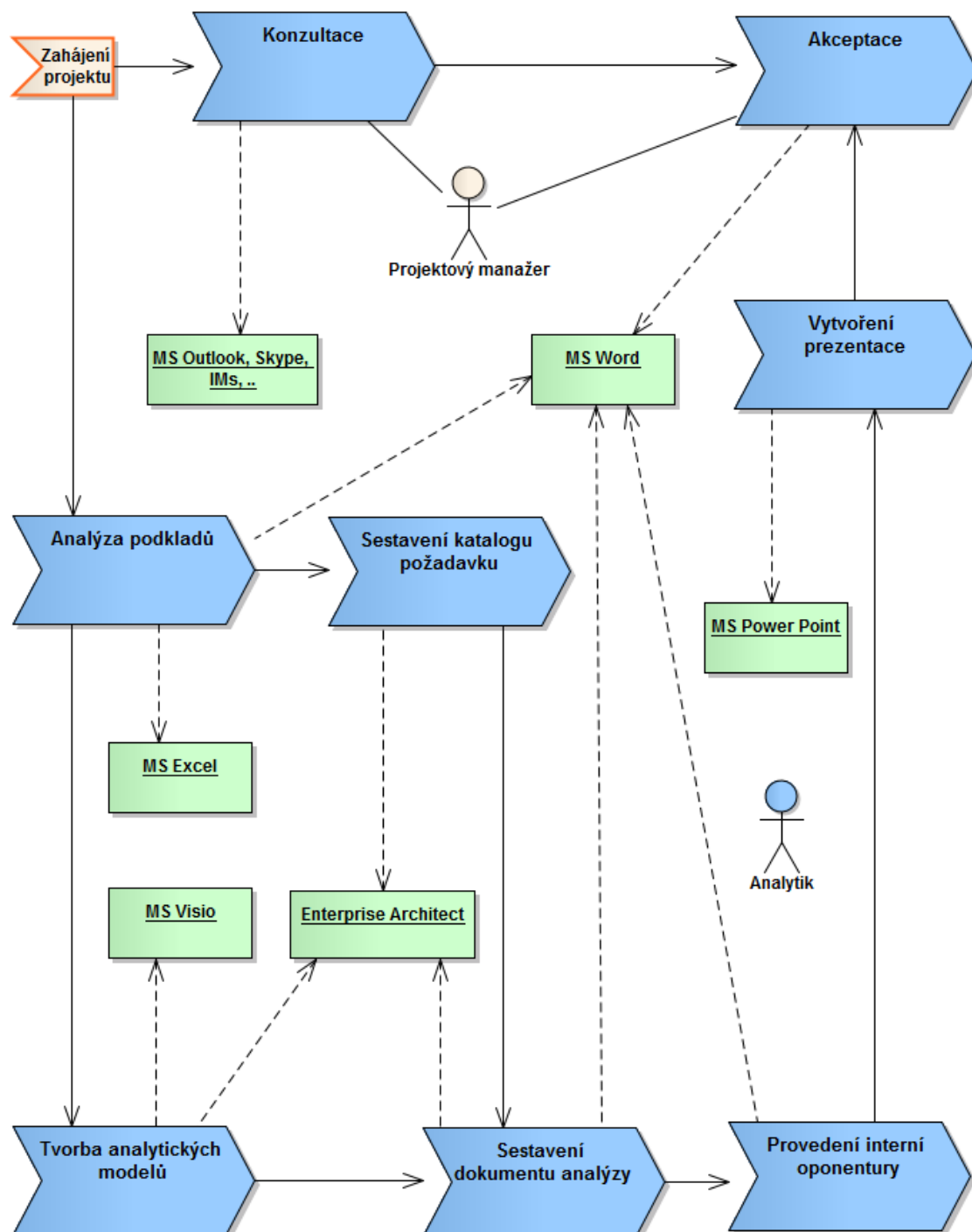
V následujícím textu jsou popsány jednotlivé procesy probíhající uvnitř firmy Blue Pixel. Obrázky Obr. 1 a Obr. 2 ukazují vzájemné závislosti mezi těmito procesy. U každého z procesů jsou znázorněny výrobní prostředky, které daný proces využívá. S procesem je také spojena osoba – pracovník, který daný proces realizuje. Pro realizaci procesů je nutno uvedené výrobní prostředky a pracovníky alokovat. U výrobních prostředků se jedná převážně o softwarové programy, které jsou nasazeny na osobních počítačích pracovníků a nebo na serverech, které jsou mezi pracovníky sdíleny.

3.1 Výrobní procesy

Následující výčet obsahuje procesy, které probíhají v rámci vývoje softwarového produktu. Jedná se o procesy, které přímo souvisejí s výrobou softwaru (produktově orientované procesy). Procesy řízení projektu jsou pak obsahem další kapitoly.

Zahájení – Procesem zahájení je označen milník, ve kterém začíná probíhat vlastní vývoj systému. Před tento proces jsou vyčleněny činnosti obchodního rázu (tvorba vize, získání zakázky, tvorba obchodního plánu apod.). Tento proces zahajuje projektový manažer.

Analýza podkladů – V tomto procesu studuje analytik podklady pro návrh systému. Jedná se o dokumenty předané zákazníkem (faktury, obchodní plány, pracovní řady, směrnice, výrobní plány apod. - vždy dle druhu vyvíjeného systému). Analytik také v závislosti na druhu projektu studuje související legislativu, konkurenční řešení, staré verze systému atd.



Obr. 1: Procesy při vývoji softwarových systémů

Konzultace – Proces konzultace v sobě zahrnuje komunikaci mezi zákazníkem a vývojovým týmem. Do procesu konzultace řadíme také část komunikace (porady, brainstorming, ...) mezi členy vývojového týmu. Zákazník při konzultacích může navrhovat změnové požadavky, které zpracovává analytik.

Katalog požadavků – V tomto procesu probíhá tvorba katalogu požadavků. Katalog požadavků zpracovává analytik na základě konzultací se zákazníkem.

Tvorba analytických modelů – V tomto procesu vznikají jednotlivé analytické modely, které popisují budoucí vlastnosti softwarového systému. Pro modelování je použit jazyk UML. Mezi zpracovávané modely patří model procesů (*Business Proces Model*), diagramy aktivit (*Activity Diagrams*), model případu užití (*Use Case Model*), doménový model (*Domain Model*) a model uživatelského rozhraní (*User Interface Model*). Modely vytváří analytik.

Dokument analýza – V tomto procesu dochází k vytvoření dokumentu analýzy. Dokument má stanovenou formu a obsahuje modely vytvořené v předchozích procesech vývoje systému. Všechny entity z uvedených modelů mají slovní popis. Dokument vytváří analytik.

Interní oponentura – Proces, ve kterém je prováděna oponentura vytvořené analýzy. Oponentura je provedena některým dalším analytikem z analytického oddělení firmy, který se nepodílí na tvorbě dané analýzy.

Prezentace – V tomto procesu dochází k tvorbě prezentace, která je určena k předvedení zákazníkovi. Návrh prezentace provádí analytik a konečnou podobu prezentace vytváří zpravidla obchodník, který prezentaci předvádí.

Akceptace – V procesu akceptace je zákazníkem potvrzena a podepsána analýza.

Konzultace – Proces konzultace v sobě zahrnuje komunikaci mezi zákazníkem a vývojovým týmem. Do procesu konzultace řadíme také část komunikace (porady, brainstorming, ...) mezi členy vývojového týmu. Tento proces se od předchozího procesu Konzultace liší tematicky, soustředí se na oblast implementace.



41

Návrh architektury – Proces, ve kterém vzniká návrh architektury. Tento návrh vytváří architekt (návrhář) a vychází z analýzy. Součástí návrhu jsou implementačně závislé modely tříd, datový model, sekvenční diagramy, diagramy komponent, diagramy nasazení systému, (případně diagram balíku, komunikační diagram apod.).

Implementace – V procesu implementace dochází k vytváření zdrojového kódu systému. Zdrojový kód vytvářejí programátoři (vývojáři), kteří vycházejí z návrhu architektury.

Testování – V procesu testování je vytvářený systém postupně testován. Testy probíhají souběžně s implementací. Tvorba, spouštění a vyhodnocení testů je úkolem testerů. Testeři vycházejí z analýzy a z návrhu architektury.

Test akceptační – Proces, který je spouštěn po dokončení implementace systému a který slouží k závěrečnému provedení testů. Testy provádějí testeři.

Dokumentace – Cílem tohoto procesu je tvorba dokumentace. Dokumentace vzniká v průběhu implementace a je dokončena před předáním systému. Dokumentaci vytvářejí analytici, vývojáři nebo testeři.

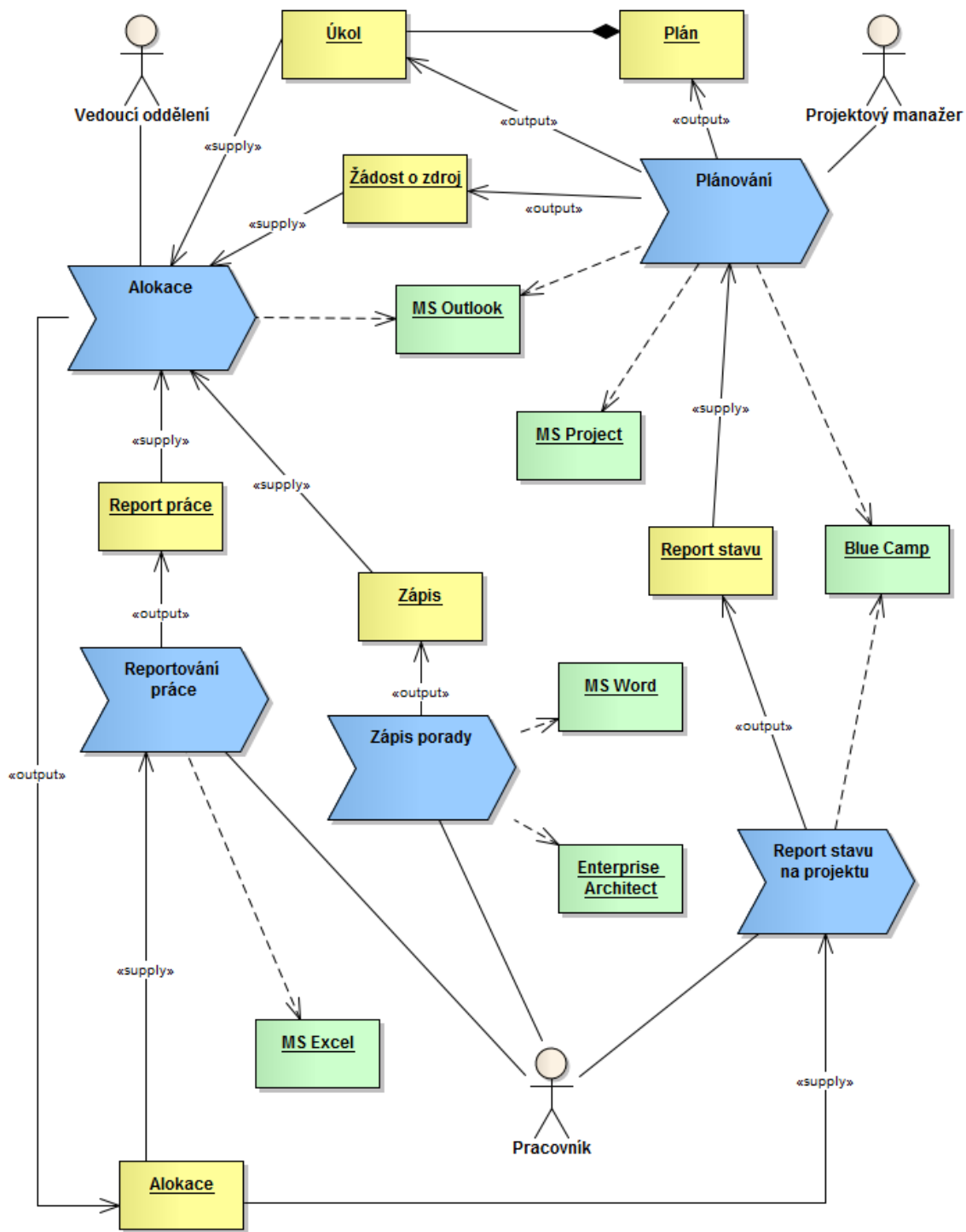
Předání – Procesem předání je označen milník, ve kterém je systém předán zákazníkovi. Do tohoto procesu patří také nasazení systému u zákazníka. Tento proces je následován procesem správy a provozu systému. Také případný import provozních dat je chápán jako navazující činnost na proces předání systému.

3.2 Řídící procesy

Následující výčet osahuje procesy, které probíhají v rámci řízení a správy projektů. Jedná se především o procesy komunikace mezi jednotlivými aktéry, kteří se podílejí na realizaci projektu.

Alokace – V tomto procesu vedoucí oddělení vybere své podřízené, které na žádost projektového manažera přidělí ke zvolenému projektu nebo úkolu. Vedoucí určí čas, který na daném projektu jeho podřízený stráví. Součástí procesu je také vyhodnocení reportů (výkazů), které podřízený odevzdá na základě provedené práce na

přidělení projektu. V procesu alokace vedoucí hodnotí také fakta, která vzešla z porady a ovlivňují jeho následná rozhodnutí.



Obr. 3: Procesy pro řízení projektu

Plánování – Jde o proces, ve kterém projektový manažer sestavuje plán projektu, žádá o přidělení zdrojů, rozděluje úkoly přiděleným zdrojům a na základě reportů stavu (výkazů o stavu provedené práce) provádí další úpravy plánu a celkové řízení projektu.

Reportování práce – V tomto procesu odevzdává pracovník svému nadřízenému výkaz o provedené práci na všech projektech, kterých se účastní.

Report stavu na projektu – V tomto procesu odevzdává pracovník projektovému manažerovi výkaz o stavu práce provedené na projektu. Pracovník tomuto projektovému manažerovi odevzdává výkaz obsahující pouze práce provedené na projektu, který daný projektový manažer řídí.

Zápis porady – V tomto procesu vytváří pracovník, který je určen jako zapisovatel porady, zápis o uskutečněné poradě. Zápisy se archivují a jsou přístupné všem zúčastněným na dané poradě a slouží jako zdroj informací o závěrech usnesených na poradě.

V diagramu Obr. 3 jsou znázorněny role, které dané procesy realizují, dále pak objekty (zápisy, žádosti, úkoly apod.), které slouží jako vstupy nebo výstupy jednotlivých procesů. Přerušovanou čarou se šipkou je znázorněn vztah mezi procesem a použitým softwarovým nástrojem.

3.3 Reinženýring procesů

Z diagramu Obr. 3 je patrné, že procesy řízení a komunikace ve společnosti Blue Pixel vyžadují ke svému fungování velké množství softwarových nástrojů. Reportování vedoucímu probíhá pomocí programu Excel a Outlook, reportování projektovému manažerovi pak pomocí systému Blue Camp. Zápis z porady je realizován dokonce použitím tří aplikací (vytvořen v Enterprise Architectu, exportován do MS Wordu a odeslán pomocí Outlooku). Používání velkého množství aplikací k v zásadě příbuzným činnostem snižuje produktivitu práce a způsobuje nárůst času spotřebovaného administrativní prací. Dokumenty se objevují v různých formátech a různých úložištích. Není také definována jasná politika přístupových práv, archivování nebo zálohování daných dokumentů.

Obr. 4: Upravené procesy pro řízení projektu



Na diagramu Obr. 4 je znázorněn upravený procesní model řízení a komunikace ve společnosti Blue Pixel. Procesy zůstaly až na výjimky stejné, ale změnily se prostředky, které jsou těmito procesy využívány. Většina komunikace, plánování a reportování je nyní postavena na systému Blue Camp. Změna procesů má vliv na efektivitu práce. Zaměstnanci používají jeden nástroj, který je určen právě a jen k požadovanému účelu. Procesy reportování práce a reportování stavu na projektech jsou realizovány společně. Pracovník vypisuje pouze jeden výkaz, ze kterého systém automaticky generuje různé reporty pro různé adresáty. Před změnou procesů musel pracovník pro svého vedoucího vytvořit jeden report práce. Pro projektové manažery pak musel za každý projekt, kterého se účastnil, vytvořit ještě další report o stavu prací.

4 ANALÝZA IS/IT FIRMY

4.1 Analýza McFarlan

Pomocí analýzy McFarlan byly identifikovány klíčově důležité aplikace, které jsou nepostradatelné pro současný chod firmy. Jedná se o nástroje, které vytvářejí vlastní produkt firmy. Patří mezi ně nástroje návrhářské, vývojářské, testovací a také infrastruktura umožňující chod těchto nástrojů. Původně byly do této skupiny nástrojů zařazeny také účetní a bankovní programy. Krátkodobý výpadek těchto programů však lze překlenout pomocí záložních řešení (např. bankovní platby lze provést pomocí internetového bankovníctví nebo platebním příkazem přímo v bance). Pro srovnání výpadek serveru zajišťujícího vývoj znamená okamžité zastavení výrobního procesu firmy.

Další skupinu systémů tvoří programy podpůrné. Programy této skupiny lze poměrně snadno nahradit alternativními nástroji. Náhrada programů je snadná a rychlá (např. programy sady MS Office lze nahradit nástroji OpenOffice, v případě výpadku Exchange Serveru lze krátkodobě použít jiný komunikační kanál apod.). Bankovní a účetní programy leží někde na pomezí těchto dvou skupin. Snadnost a rychlost náhrady daných programů je relativní, lze ale použít například srovnání s náhradou vývojového prostředí, která je jednak vysoce náročná a u probíhajících projektů i téměř nemožná.

Pro budoucnost je důležité udržování a aktualizování klíčových systémů. Pořízení nových verzí těchto systémů připadá v úvahu v momentě, kdy se nové systémy osvědčí a na Hype křivce zaujmou pozici tzv. osvětleného nástupu (Slope of Enlightenment - bod, kdy se technologie začíná postupně prosazovat do reálné praxe). Dřívější nasazení nové technologie je možné, pokud by na něm trval zákazník a bylo by v technických možnostech firmy této žádosti vyhovět.

Čtvrtou skupinu tvoří podpůrné systémy s budoucím potenciálem. Z nástrojů firmy Blue Pixel sem lze zařadit systém řízení projektů Blue Camp. Po určitých úpravách tohoto softwaru a některých firemních procesů existuje reálná příležitost na zrychlení vnitropodnikových procesů a zvýšení efektivity práce plynoucí z lepší organizace administrativních úkonů prováděných běžnými zaměstnanci firmy.

Taxativní výčet aplikací a jejich přiřazení do jednotlivých skupin je obsahem tabulky Tab. 3.

Tab. 3: Rozdělení aplikací dle analýzy McFarlan

budoucnost	STRATEGICKÉ	POTENCIÁLNÍ
	(nové verze současných)	Blue Camp
	<i>Restrukturalizace procesů</i>	<i>Inovace procesů</i>
současnost	KLÍČOVÉ	PODPŮRNÉ
	Operační systémy, servery SQL Servery, databáze Enterprise Architect Visual Studio NetBeans Testovací nástroje	MS Project MS Visio MS Office Exchange Server Tortoise SVN Mantis Elvis Grafické a Flash nástroje Účetní a bankovní programy
	<i>Koordinace procesů</i>	<i>Úspora nákladů</i>
	nutnost	možnost

4.2 Analýza HOS 8

Tato kapitola se zabývá zhodnocením informačního systému Blue Camp pomocí metody HOS 8. Metoda nám poskytne ucelený pohled na informační systém s důrazem na zhodnocení vyváženosti celého systému. Systém je hodnocen v osmi oblastech: hardware, software, orgware, peopleware, dataware, customers, suppliers a management IS.

Dotazníky, podle kterých bylo hodnocení vytvořeno, jsou k nahlédnutí v příloze č. 5. Obodování jednotlivých odpovědí je uvedeno (dle zkoumaných oblastí) v následujících tabulkách. U každé tabulky je uveden také výsledek výpočtu stavu dané oblasti.

Oblast hardware: celkový stav je 4

Otázka:	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.
Body:	3	4	3	4	2	5	4	3	3	5

Oblast software: celkový stav je 3

Otázka:	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.
Body:	1	3	3	4	2	1	3	4	2	2

Oblast orgware: celkový stav je 3

Otázka:	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.
Body:	1	2	3	2	4	5	4	2	1	3

Oblast peopleware: celkový stav je 3

Otázka:	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.
Body:	4	3	2	3	2	3	3	2	2	2

Oblast dataware: celkový stav je 4

Otázka:	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.
Body:	5	5	1	4	4	4	1	3	4	5

Oblast customers: celkový stav je 3

Otázka:	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.
Body:	5	3	2	4	5	3	3	3	2	3

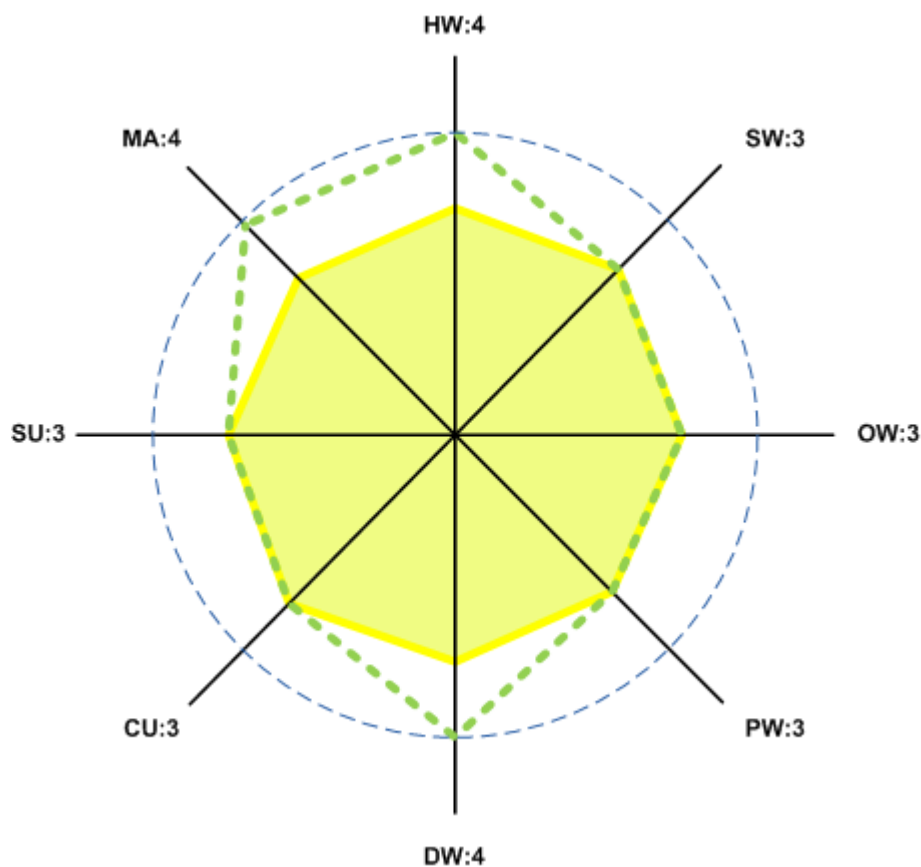
Oblast suppliers: celkový stav je 3

Otázka:	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.
Body:	3	3	4	2	4	4	4	3	3	2

Oblast management IS: celkový stav je 4

Otázka:	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.
Body:	5	5	2	3	2	3	5	3	4	4

Stav zkoumaných oblastí je zanesen do diagramu Obr. 5. Oblasti společně s hodnotou jejich stavu jsou vyneseny na příslušné poloosy diagramu. Tyto body jsou spojeny přerušovanou zelenou čarou.



Obr. 5: Stav zkoumaných oblastí dle analýzy HOS 8

Souhrnný stav informačního systému je pak v diagramu zaznačen pomocí žlutého osmiúhelníku se světle žlutou výplní. Souhrnný stav se rovná stavu nejnižší složky informačního systému.

Posuzovaný informační systém lze považovat za vyvážený. Splňuje totiž podmínku vyváženosti:

Pro všechna u_i platí: $(u_i - u) \leq 1$ a dále platí $\sum_{i=1}^8 (u_i - u) \leq 3$,

kde u_i je stav i -té oblasti a u je souhrnný stav informačního systému.

Podle významu informačního systému pro firmu byla určena úroveň doporučeného souhrnného stavu informačního systému. Význam informačního systému byl ohodnocen číslem **1**. To znamená, že zkoumaný informační systém je pro chod

firmy klíčově důležitý a jeho výpadek výrazně ovlivní fungování firmy a potažmo i zisk a spokojenost zákazníků. V systému jsou zaneseny všechny alokace, plány projektů, reporty a další řídicí informace. Bez těchto údajů by bylo ohroženo řízení projektů a v důsledku toho i samotný chod prací na projektech. Při hodnocení významu informačního systému uvažujeme stav, kdy je stávající verze systému Blue Camp nasazena na podporu procesů vzniklých po provedení reinženýringu procesů. Tzn., že pro podporu procesů již nejsou používány předchozí nástroje, které systém Blue Camp nahrazuje. Viz kapitola Reinženýring procesů.

Dle významu systému byla tedy určena úroveň doporučeného souhrnného stavu informačního systému na hodnotu 4. Tato úroveň je zaznačena v diagramu pomocí přerušované kružnice modré barvy.

Opatření dle doporučení HOS 8

Metoda HOS 8 pro každou kombinaci doporučeného souhrnného stavu systému a dosaženého souhrnného stavu systému uvádí výčet opatření, která je vhodné provést za účelem dosažení vyváženosti systému. Uvedená opatření by také měla zajistit úroveň systému odpovídající jeho významu (důležitosti) pro chod firmy. Pro jednotlivé oblasti systému je pak definována příslušná strategie.

Souhrnný stav informačního systému je nižší než doporučený stav odvozený od významu informačního systému, proto je třeba souhrnný stav zvýšit. Pro zvýšení souhrnného stavu vyváženého systému je ale zapotřebí zvýšit stav většiny oblastí. Jako vhodná se tedy jeví strategie expanze. Zvýšení výdajů je doporučeno pro oblasti Software, Orgware, Peopleware, Customers a Suppliers.

5 INFORMAČNÍ SYSTÉM SPRÁVY PROJEKTŮ

5.1 Výběr vhodného systému

Většina z pracovních procesů firmy je podporována softwarovými nástroji ve vyhovující míře. Byla však nalezena oblast, která není zcela uspokojivě pokryta pomocí softwarových informačních systémů. Jedná se o oblast řízení projektů a to zejména o procesy alokace zdrojů, zadávání úkolů a reportování práce. Tyto činnosti jsou prováděny za pomoci několika různých nástrojů, které plně nepokrývají požadavky projektového řízení. Pro alokaci zdrojů se používá aplikace MS Project a MS Outlook. Reportování práce probíhá za pomoci MS Excelu a samotné reporty jsou zasílány prostřednictvím emailové pošty. Pro zpracování výsledků a tvorbu statistik pak slouží proprietární nástroj s omezenými možnostmi.

Současné řešení s sebou nese řadu nevýhod:

- Absence jednotné databáze zaměstnanců. Pro tvorbu a plánování alokací používají projektiví manažeři databázi aplikace MS Project. Jednotliví vedoucí oddělení pak přidělují úkoly svým podřízeným pomocí aplikace MS Outlook, která využívá další nezávislou databázi. Také nástroj pro vyhodnocování reportů obsahuje svůj vlastní seznam zaměstnanců.
- Reporty nejsou vázané na zadané úkoly. Pracovník zapíše podle předepsaného vzoru název projektu, ale konkrétní činnost a úkol již specifikuje vlastními slovy. Neexistuje tedy jednoznačná vazba na zadaný úkol či projekt.
- Reporty nemusejí obsahovat validní data, jejich automatické zpracování je obtížné. Pokud provede zaměstnanec v reportu chybu (např. v názvu projektu nebo ve formátu času), pak nejsou reporty jeho práce správně zpracovány.
- Reporty se provádějí pouze jednou týdně. Informace o stavu projektu tedy nejsou aktuální a projektový manažer na základě nich nemůže dostatečně rychle reagovat.

- Pokud je pracovník zároveň nasazen na více projektech, které mají různé projektové vedení, pak musí vytvářet a zasílat reporty zvlášť pro každého z projektových manažerů. Další souhrnný report veškeré provedené práce musí podřízený zaslat vedoucímu svého oddělení.
- Velké množství používaných nástrojů s různým uživatelským rozhraním a způsobem ovládání. Jednotlivé úkony jsou prováděny v dílčích programech, což vyžaduje více času potřebného k provedení těchto úkonů, než by vyžadovalo řešení s centrálním softwarovým systémem jednotného rozhraní a ovládání.

Oblast projektového řízení je tedy místem, které lze zefektivnit nasazením vhodnějšího softwarového systému. Nový systém by měl zahrnovat funkcionalitu pro vytváření alokací, přidělování úkolů, reportování provedené práce a vyhodnocování stavu projektu. Hlavním cílem by mělo být zrychlení cyklu, který začíná alokací zdroje a končí vyhodnocením průběžného stavu projektu. Pokud získá projektový manažer informace o stavu projektu dříve, je schopen rychleji reagovat na vzniklé události. Přesnější a lépe zpracované informace pak podporují správná rozhodnutí manažera.

Vytyčené cíle je možné splnit buď nasazením některého ze softwarových řešení dostupných na trhu, nebo realizací vlastního systému „na míru“. Společnost Blue Pixel je autorem softwarového systému Blue Camp, který částečně cílovou funkcionalitu pokrývá. K úplnému pokrytí všech požadavků by však bylo potřeba funkcionalitu tohoto systému ještě rozšířit.

5.1.1 Project management software

Hledaný software spadá do skupiny tzv. systémů pro podporu projektů (Project management software). Tato skupina zahrnuje aplikace specializující se na plánování, kontrolu a rozpočet výdajů, alokaci zdrojů, týmovou spolupráci a komunikaci.

Systémy podpory projektů jsou realizovány buď jako *desktopové* nebo *webové* aplikace.

Desktopové aplikace jsou nativní lokální aplikace, které jsou spouštěny v operačním systému uživatele. Výhodou bývá rychlá odezva aplikace, komplexnost

grafického rozhraní, uživatelsky přívětivé chování a komfort funkcí. Data jsou většinou ukládána lokálně a aplikace vyžaduje připojení k Internetu jen za účelem synchronizace. Pracovat s aplikací lze tedy po určitou dobu i offline. Nevýhodou desktopových aplikací bývá nutnost instalace a závislost na daném operačním systému.

Webové aplikace běží na serverech v lokálních sítích nebo v síti Internet. Přístupné jsou uživatelům pomocí webového prohlížeče. Zpravidla tyto aplikace nevyžadují žádné další instalace. Data jsou ukládána centrálně na serveru a snižují tak náklady na lokální úložný prostor, ale je požadováno kvalitní a rychlé internetové připojení. Pokud je uživatel offline, pak s aplikací nemůže pracovat. Aplikace není závislá na operačním systému a data jsou přístupná v libovolnou dobu z jakéhokoliv počítače připojeného do sítě. Současné možnosti internetových webových aplikací jsou z hlediska uživatelského komfortu a funkcí limitovány omezeným rozsahem běžně rozšířených technologií internetu. Existují však nové přístupy (například tzv. *Rich Internet Application*), které přinášejí výhody uživatelského komfortu i pro webové aplikace. Webové aplikace se také vyznačují snazší údržbou, která je prováděna centrálně na aplikačním serveru bez zásahu do klientských počítačů.

Svým charakterem firmě Blue Pixel více vyhovují systémy fungující na modelu webové aplikace. S reporty pracuje velké množství uživatelů z různých míst a operačních systémů. Také není nutné mít data k dispozici offline, ale naopak je potřeba co nejčastěji provádět reporty, což znamená pracovat s aplikací online, aby mohly být informace v reálném čase synchronizovány s aplikačním serverem systému.

Systémy podpory projektů lze dále rozdělit na hostované (SaaS, ASP) a na vlastněné (spravované) uživatelem.

5.1.2 Hostované systémy

Hostované systémy běží na zařízeních poskytovatele a zpravidla jsou dostupné vzdáleně přes síť Internet. O zabezpečení dat, update softwaru a zálohování se stará poskytovatel. Uživatel neplatí za licenci softwaru, ale od poskytovatele si jej pronajímá.

Hostování je nabízeno buď jako SaaS (Software as a Service) nebo jako ASP (*Application Service Provider*). Liší se dle umístění dat uživatelů. V prvním případě

jsou data uložena na jednom serveru, který sdílí více uživatelů. V druhém má každý uživatel svůj vlastní server. Aberdeen Group u hostovaných systémů definuje ještě jemnější rozdělení. (1)

Hostování přináší jistá pozitiva ale i negativa. Mezi přínosy lze řadit pružnější licenční politiku, kdy cena za pronájem může být určena podle toho, jak zákazník software využívá. Dále se mezi výhody řadí možnost okamžitého využití aplikace (ihned k dispozici), predikovatelnost a linearita nákladů (většina nákladů je rozložena do průběžných poplatků), vzdálený přístup odkudkoliv apod. Jako nevýhodu mohou zákazníci vnímat správu vlastních dat jiným subjektem. Pro některé společnosti mají zpracovávaná data existenční význam a jejich správa jiným subjektem je neakceptovatelná. (17)

Management společnosti Blue Pixel se rozhodl u systému pro řízení projektů nevyužívat hostovaná řešení.

5.1.3 Volba vhodného systému pro podporu řízení projektů

Na trhu je k dispozici velké množství aplikací, které se řadí do skupiny systémů pro podporu projektů. Některé z nich se snaží být univerzální a pokrývat všechny aspekty řízení projektů. Jiné se v rámci podpory projektů specializují pouze na určité oblasti. Jedná se například o oblast plánování, kontroly výdajů, alokace zdrojů, reportování práce, týmové spolupráce, sdílení informací a komunikace.

Před samotným výběrem nejvhodnějšího systému podpory projektů je třeba nejprve definovat vlastnosti, které firma Blue Pixel od systému očekává. Požadované vlastnosti vycházejí z rozboru procesů, které nejsou dostatečně a uspokojivě podporovány použitými softwarovými nástroji. Tento nevyhovující stav by měl být vyřešen právě nasazením nového systému, který dostatečnou podporu procesů zajistí.

Požadované vlastnosti seřazené dle priority:

- Alokace zdrojů a přidělování úkolů
- Reportování práce, míra dokončení úkolu
- Zpracování reportů a statistiky

- Orientace na projekty vývoje softwaru
- Sofistikovanější podpora plánování vývoje softwaru

Přehled dostupných systémů, které by přicházely v úvahu, je k dispozici v příloze č. 2. V přehledu jsou uvedeny názvy systémů, jejich verze, výrobci, ceny a stručné srovnání vlastností.

Pokud bychom chtěli vyhovět všem stanoveným požadavkům firmy, ať už se jedná o požadavky definované managementem firmy nebo o požadavky vyplývající z procesní analýzy, pak by nejlépe vyhovoval systém Blue Camp. Tento systém sice svou stávající funkcionalitou všechny vznesené požadavky neuspokojuje, ale jelikož se jedná o produkt firmy, existuje možnost úpravy stávajících funkcí a implementace funkcí nových, které by již dané požadavky splňovaly.

5.2 Postup návrhu nových funkcí

Navržené funkce systému Blue Camp vycházejí z požadavků definovaných v analytické části práce. Vedle přizpůsobení systému novému procesnímu uspořádání byl návrh funkcí zaměřen také na reálné zvýšení hodnoty pro zákazníka. Zákazníkem je v tomto případě zaměstnanec firmy - projektový manažer, vedoucí týmu a všichni pracovníci, kteří se účastní hlavních výrobních procesů firmy. Hodnota pro zákazníka je definována jako vztah mezi uspokojením potřeby a zdroji použitými pro dosažení tohoto uspokojení (4).

Z pěti možných cest vedoucích ke zvýšení hodnoty byla vybrána ***cesta rychlého růstu užítku při pomalejším růstu celkových nákladů*** (4). Systém Blue Camp přináší uživateli vysoký užitek, protože zefektivňuje některé procesy při plánování a řízení projektů. Vyžaduje však dodatečné finanční náklady. Procesy, které systém Blue Camp zajišťuje, lze realizovat i pomocí skupiny jiných kancelářských softwarových nástrojů, které jsou levnější. Systém Blue Camp však přináší ucelené řešení, které poskytuje funkce navržené přesně na míru podporovaným procesům. Jednoduchost, cílenost a integrita tohoto řešení zajišťuje vysoký užitek pro zákazníka za cenu mírně zvýšených nákladů.

5.2.1 Varianty řešení

Při vytváření nápadů a představ o funkcích systému byly použity některé techniky tzv. tvořivého myšlení (4). Cílem těchto technik je nalezení nových způsobů řešení, které přinesou zvýšení hodnoty pro zákazníka.

5.2.1.1 Porovnávání podobností a funkcí

Hlavní funkce systému vznikaly na základě podobných konkurenčních řešení. Přehled konkurenčních aplikací je uveden v příloze č. 2. Návrh funkcí vycházel také z vlastností kancelářských softwarových nástrojů, které jsou používány k podpoře stejných procesů, jaké mají být podporovány i nově vyvíjeným systémem. V průběhu návrhu byly redukovány funkce, které jsou v kancelářských nástrojích navíc a které tedy nejsou využívány k podpoře požadovaných procesů. Tímto byly sníženy náklady na vývoj základních funkcí. Odbouráním nepotřebných funkcí byla zvýšena rychlost a přehlednost aplikace.

5.2.1.2 Konfrontační techniky

Další nové funkce systému, zejména funkce vedlejší a podpůrné, byly identifikovány pomocí technik „*Brainstormingu*“ a techniky „*Advocatus Dei et Advocatus Diabli*“ (neboli techniky „*Pro a proti*“) (4). Ve fázi brainstormingu byly nalezeny nové nápady a nové varianty řešení. Společně s náměty a připomínkami jednotlivých pracovníků (budoucích uživatelů systému) byly nové návrhy oponovány za pomoci techniky „*Pro a proti*“. Návrhy, které obstály kritickým argumentům, byly schváleny k realizaci. Tímto se podařilo vytvořit nové funkce, které obohatí systém složený ze základních funkcí o nové vlastnosti zajišťující další zefektivnění práce a vyšší užitečnost pro uživatele.

5.3 Funkce systému Blue Camp

5.3.1 Stávající funkcionalita

Následující přehled obsahuje balíky funkcionalit, které jsou v systému již úspěšně implementovány a jsou použitelné v reálném provozu. Uvedené funkce nepotřebují

další úpravy. Jsou zde uvedeny, aby doplnily kompletní výčet budoucích funkcí celého systému. Balíky stávajících funkcionalit jsou označeny čísly od 1 do 6. Označení bude dále použito v hodnotové analýze, ze které bude na balíky funkcí odkazováno.

1. Správa systému

Funkce správy systému slouží k základnímu nastavení systému, určování uživatelských rolí, evidenci uživatelů, evidenci zdrojů, zálohování a k exportům dat. Nastavení jsou většinou ve formě uživatelsky modifikovatelných číselníků. Nastavení jdou napříč celou aplikací, od věcí projektově závislých, přes fond pracovní doby včetně kalendáře svátků, až po definici legislativních prostorů, která umožňuje oddělit pobočky působící v různých zemích. Systém podporuje zavedení tzv. „fixních typů práce“, které zastupují kategorie pracovních úkonů. Typy práce mají hierarchickou strukturu s rozpadem na podtypy podporující snadnou orientaci a jednoduchou správu činností vykonávaných v podniku.

2. Založení projektu

Typický scénář projektu začíná založením projektu. Právo založit projekt má typicky jen projektový manažer. Ten v první fázi stanoví hrubý rozsah projektu, jak z hlediska časového, tak z hlediska finančního objemu.

3. Sestavení týmu

Projektový manažer na základě předběžného plánu zadá požadavek o alokaci zdrojů (lidí, serverů či pracovišť). Správce daného zdroje (u lidí je to vedoucí týmu) uvolní požadované množství zdrojů. Díky variabilnímu nastavování uživatelských rolí podporuje systém jak liniiovou organizační strukturu, tak i organizační struktury s pružnými prvky (maticovou strukturu). Členy týmu lze průběžně přidávat a odebírat nebo měnit jejich pozice.

4. Sestavení harmonogramu

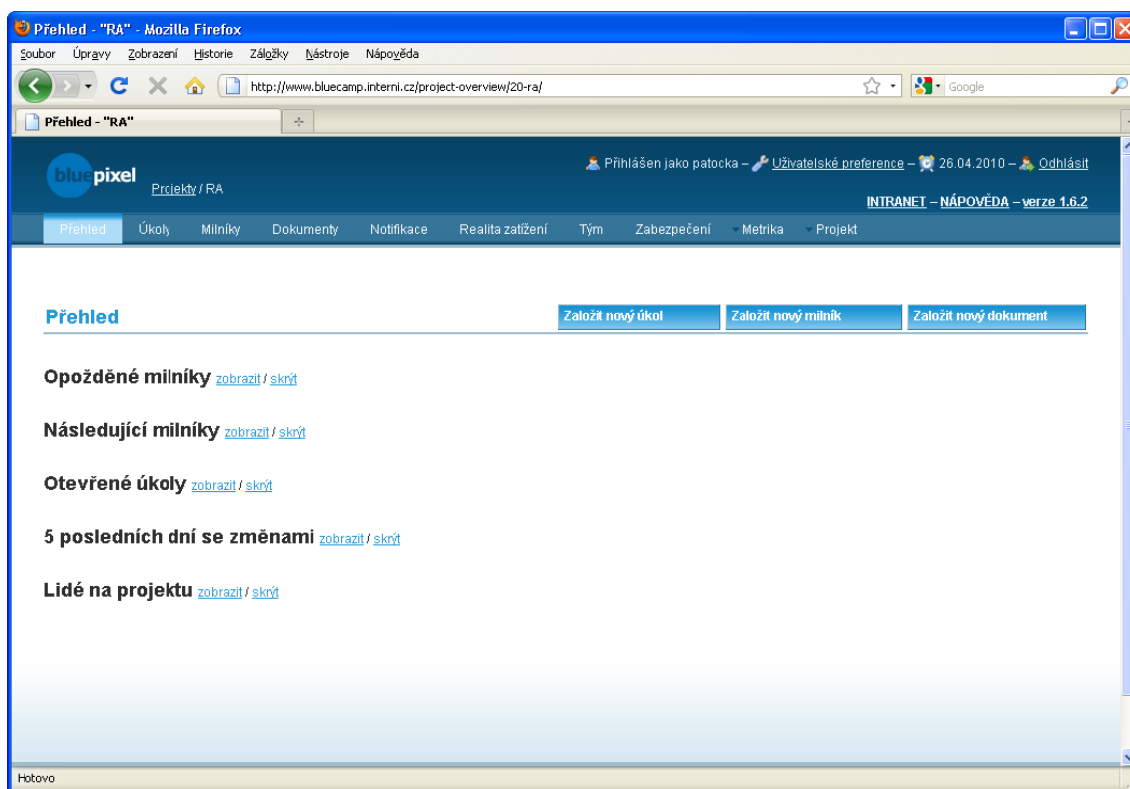
Dále je v systému zpracován hrubý harmonogram projektu, který je upřesněn po provedení analýzy. Jsou naplánovány milníky, tedy termíny, které vytyčují hranice jednotlivých fází projektu. Milníky jsou typicky události jako schválení analýzy, ukončení vývoje, ukončení testů, dokončení nasazení systému, ukončení podpory apod.

5. Naplánování fází projektu

Pomocí milníků jsou nadefinovány fáze, do kterých zanášíme alespoň hrubý popis činností probíhajících v daných fázích. Upřesnění obsahu činností zpravidla proběhne po ukončení analýzy (s výjimkou milníků a činností samotné analýzy, které je třeba podrobněji specifikovat hned po založení projektu).

6. Úkoly a Todo listy

Projektový manažer na základě harmonogramu přidělí podřízeným pracovníkům na konkrétním projektu úkoly s odpovídajícím časovým prostorem pro jejich splnění. Pracovníci úkoly průběžně plní a zaznamenávají průběh prací ve formě pracovních výkazů. Zadané úkoly jsou součástí tzv. „todo listů“. Jedná se o seznam přidělených úkolů a termínů jejich splnění. Úkol se vždy nachází v jednom z předdefinovaných stavů (založený, přidělený, realizovaný, ukončený, zrušený, ...).



Obr. 6: Náhled na uživatelské rozhraní systému Blue Camp

5.3.2 Nově navržená funkcionalita

Následující přehled obsahuje nové funkcionality, které jsou navrženy tak, aby systém odstranil nevýhody původního řešení a splňoval všechny požadavky vycházející ze zadání managementu firmy a z procesní analýzy. Balíky nových funkcionalit jsou značeny čísla od 7 do 18. Označení bude dále použito v hodnotové analýze, ze které bude na balíky funkcí odkazováno.

7. Zobrazení celkového přehledu

Celkový přehled je vlastně domácí stránka uživatele v systému. Je to obrazovka, která obsahuje vše důležité pro uživatele. Pro běžného člena týmu jsou zobrazeny bližící se úkoly a pro vedoucího aktuální milníky a projekty. Jednotlivé přehledy jsou tedy závislé na rolích uživatele. Postranní panel by měl upozorňovat na poslední založené projekty a úkoly. Na téže stránce by měl mít uživatel možnost pohodlně založit nový úkol nebo výkaz práce. K tomu slouží tzv. „inline formulář“. Jedná se o jednořádkový formulář, který se používá k rychlému zadání daného záznamu, aniž by se otvírala další editační stránka nebo dialogové okno. Většinou je tento řádek umístěn nad tabulkou záznamů, které vkládá. Všechny možné hodnoty (jako datum, čas apod.) jsou předvyplněny. Ostatní položky je možné vybrat z číselníku nebo vepsat jako text či číslo. Celkový přehled agreguje důležité informace do úvodní stránky, což by mělo výrazně zlepšit orientaci uživatele v aplikaci a tím i celkové zrychlení práce s aplikací.

8. Srovnání reality a plánu

Tato funkcionalita umožňuje zaměstnanci konfrontovat svoje vykázané úkony s plánovaným časovým rozsahem. Stejně tak může srovnání reality a plánu kontrolovat nadřízený týmový vedoucí nebo projektový manažer přidělený k danému projektu. Zaměstnanec je zejména upozorňován, pokud realizace jeho úkolu přesáhne plánovaný časový rozsah.

9. Hodnocení efektivity zdrojů

Pomocí nástroje pro vyhodnocení efektivity použitých zdrojů je možné sledovat realitu zatížení a poměr plánovaného k reálnému zatížení. Hodnocení efektivity je přístupné pro vedoucí týmů a pro projektové manažery. Nástroj by měl obsahovat grafické znázornění výsledků pomocí grafů a export výsledků do formátu xls.

10. Podrobný přehled

Podrobný přehled obsahuje totéž, co přehled celkový, avšak sestava má pouze informace za zvolený projekt. Jsou zde také zobrazeny detailnější informace jako historie založení milníků, úkolů atd. Projektový manažer pomocí této funkce vidí rekapitulaci za všechny zdroje podílející se na projektu. Důležité časové okamžiky projektu (milníky, začátky a konce úkolů, ...) jsou zobrazeny v kalendáři na postraním panelu. Také podrobný přehled obsahuje *inline* vkládání záznamů. Přehled agreguje důležité informace do úvodní stránky, což by mělo výrazně zlepšit orientaci uživatele v aplikaci a tím i celkové zrychlení práce s aplikací.

11. Správa zabezpečení

Správa zabezpečení slouží k přidělování a odepírání oprávnění skupinám a jednotlivcům. Stávající řešení je neintuitivní a těžkopádné. Zejména neexistuje možnost hromadného přidělení práv a obecně možnost práce se skupinou uživatelů. Také je nutné uživatelské role rozšířit a přesně definovat příslušná práva ve vztahu k funkcionalitám a informacím v systému.

12. Určení pracnosti a ceny

Funkce, která napomáhá projektovému manažerovi v určení pracnosti a cen práce, což následně využije při plánování projektu. Vedle zobrazení předpočítaných tabulek, které obsahují pomocné hodnoty (vyčíslení nákladů na den práce zdroje, kvantifikace typových projektů apod.), lze použít také funkce, které pomáhají určit pracnost a cenu. Příkladem může být metodika UCP (Use Case Points), která slouží k odhadu pracnosti projektu na základě Use Case modelu vytvořeného ve fázi analýzy. Další možností je funkce založená na metodě FP (Function Points), která provádí odhad pomocí tzv. funkčních bodů vytvářeného softwaru.

13. Vytváření a aplikace šablon

Velmi užitečnou funkcí, která urychluje a zpřesňuje plán vývoje, je vytváření a aplikování šablon. Typickým příkladem šablony může být typ projektu "Webové stránky" nebo "ERP systém". Šablony umožňují v nových projektech aplikovat zkušenosti z uskutečněných projektů. Tato technika je (navzdory množství metodik určujících pracnost softwarového vývoje) v současné době pořád nejpoužívanější a

obecně nejpřesnější. Projektoví manažeři odhadují pracnost a cenu projektů nejčastěji na základě zkušeností s podobnými uskutečněnými projekty. Každá šablona obsahuje popis projektu, původně odhadovanou pracnost a reálnou pracnost zadanou při dokončení projektu. Aplikováním šablony na nový projekt nastavíme milníky a činnosti jednotlivých fází. Pomocí šablon není realizováno alokování konkrétních zdrojů, součástí šablony je pouze vytvoření požadavku na určité množství zdrojů (např. na počet testerů). Konkrétní zdroje pak uvolní vedoucí daného týmu.

14. Úprava milníků

Milníky lze posouvat podle aktuálního stavu projektu. Milníky posouvá projektový manažer nebo jeho nadřízený. Posun jednoho milníku může znamenat posunutí všech následujících milníků a nebo jen zkrácení následující fáze bez vlivu na ostatní milníky. Ve stávající verzi neměl posun jednoho milníku vliv na ostatní milníky.

15. Porady

Funkce pro podporu konání porad umožňuje vytvořit osnovu pro průběh porady, podle které moderátor porady postupuje v projednávání jednotlivých bodů. Zapisovatel porady provádí zápis přímo do osnovy. Po dokončení zápisu se tento dokument zveřejní ostatním zúčastněným a jejich vedoucím. Zapisovatele, moderátora a účastníky porady určí vyhlašovatel porady.

16. Výkazy

Uživatel, který je v roli projektového manažera nebo vedoucího týmu, vidí seznamy výkazů na všech svých projektech nebo seznamy výkazů všech svých podřízených. Zaměstnanec, který není v roli vedoucího ani projektového manažera, vidí jen svoje pracovní výkazy. Vykazuje se projekt, na němž byla práce vykonána, popis práce a odpracovaná doba. Základní idea výkazu je taková, že se vykazuje pravidelně v kratších úsecích (typicky např. půl hodinové úseky), maximálně po dnech. Zaměstnanec vytvoří denní výkaz své práce bez ohledu na to, na jakých projektech pracuje. Podle předmětu jednotlivých položek výkazu jsou pak automatizovaně sestaveny výkazy určené projektovému manažerovi (jen položky relevantní projektu) a pak výkazy určené vedoucímu (obsahují všechny položky denního výkazu).

17. Dokumenty

Funkce pro sdílení dokumentů mezi uživateli. Funkce je vhodná pro sdílení menších souborů (např. schválená analýza, výsledky testů, ...). Není určena pro sdílení zdrojových kódů programu apod. U sdílení dokumentů je podporováno verzování.

18. Notifikace

Nastavení upozornění pro jednotlivé role na projektu. Uživatel si může nastavit upozornění na libovolnou událost. Pokud tato událost nastane, pak je uživateli zaslán mail, nebo je upozornění zobrazeno na uživatelské hlavní stránce v systému.

5.4 Hodnotová analýza

Pomocí hodnotové analýzy byla určena hodnota jednotlivých funkcí ve vztahu k uživateli. Podle pořadí funkce v posloupnosti dané hodnotou či užitečností můžeme plánovat, které funkce implementovat přednostně a které neimplementovat vůbec. Funkce označené jako základní bude potřeba implementovat všechny. V opačném případě by došlo ke změně hlavní funkce systému (případně až k praktické nefunkčnosti celého systému). Naopak funkce vedlejší a podpůrné s nízkou hodnotou pro uživatele je možné v případě potřeby odložit na pozdější realizaci.

5.4.1 Párové srovnávání funkcí

Princip metody párového srovnávání spočívá v tom, že funkce jsou srovnávány každá s každou (po dvojicích) v tabulce Tab. 4, kde v řádcích i sloupcích jsou funkce ve stejném pořadí. Číselné označení funkcí v tabulce koresponduje s označením funkcí v kapitole Funkce systému Blue Camp.

Je-li funkce v řádku důležitější (užitečnější) než ve sloupci, zapíšeme 1, v opačném případě 0. Pokud pokládáme funkce za stejně důležité, přiřadíme každé funkci půl bodu. Součtem hodnot v *i*-tém řádku tabulky dostaneme číslo, které udává, kolikrát je daná funkce důležitější než ostatní funkce.

Tab. 4: Párové srovnání funkcí v hodnotové analýze

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	Σ
1	0,5	0,5	0,5	0,5	0,5	0,5	1	1	1	1	1	1	1	1	1	1	1	1	15
2	0,5	0,5	0,5	0,5	0,5	0,5	1	1	1	1	1	1	1	1	1	1	1	1	15
3	0,5	0,5	0,5	0,5	0,5	0,5	1	1	1	1	1	1	1	1	1	1	1	1	15
4	0,5	0,5	0,5	0,5	0,5	0,5	1	1	1	1	1	1	1	1	1	1	1	1	15
5	0,5	0,5	0,5	0,5	0,5	0,5	1	1	1	1	1	1	1	1	1	1	1	1	15
6	0,5	0,5	0,5	0,5	0,5	0,5	1	1	1	1	1	1	1	1	1	1	1	1	15
7	0	0	0	0	0	0	0,5	1	1	0,5	0	1	1	0,5	1	1	0,5	1	8
8	0	0	0	0	0	0	0	0,5	1	0,5	0	0,5	0,5	0	0	0	0	1	4
9	0	0	0	0	0	0	0	0	0,5	0	0	0	0	0	0	0	0	0,5	1
10	0	0	0	0	0	0	0,5	0,5	1	0,5	0	0	0,5	0	0	0	0	0	3
11	0	0	0	0	0	0	1	1	1	1	0,5	1	1	1	1	1	1	1	11,5
12	0	0	0	0	0	0	0	0,5	1	1	0	0,5	0,5	0	0,5	0,5	0,5	1	6
13	0	0	0	0	0	0	0	0,5	1	0,5	0	0,5	0,5	0	0	0	0	1	4
14	0	0	0	0	0	0	0,5	1	1	1	0	1	1	0,5	1	1	1	1	10
15	0	0	0	0	0	0	0	1	1	1	0	0,5	1	0	0,5	0,5	1	1	7,5
16	0	0	0	0	0	0	0	1	1	1	0	0,5	1	0	0,5	0,5	1	1	7,5
17	0	0	0	0	0	0	0,5	1	1	1	0	0,5	1	0	0	0	0,5	1	6,5
18	0	0	0	0	0	0	0	0	0,5	1	0	0	0	0	0	0	0	0,5	2
Σ	Kontrolní součet:																		162

5.4.2 Ocenění funkcí a výpočet hodnoty pro zákazníka

V tabulce Tab. 5 jsou uvedeny jednotlivé funkce, jejich užitečnost (z párového porovnání), odhadnuté náklady na realizaci funkce, hodnota funkce pro zákazníka (jako poměr užitečnosti a nákladů) a druh funkce.

Druh funkce je označen třemi hodnotami: *základní*, *vedlejší* a *podpůrná*.

Základní funkce jsou klíčovými funkcemi systému. Odebrání nebo přidání základní funkce změní hlavní funkci celého systému. Změna v souboru základních funkcí znamená, že je dále řešen zcela jiný systém.

Vedlejší funkce svojí funkcionalitou doplňují základní funkce systému. Změna v souboru vedlejších funkcí ovlivní celkovou koncepci řešení.

Podpůrné funkce svojí funkcionalitou vylepšují hlavní funkci systému. Změna v souboru podpůrných funkcí již zpětně neovlivňuje celkovou koncepci řešení.

Tab. 5: Ohodnocení funkcí systému Blue Camp

Funkce:	Užitečnost:	Cena: [tis. Kč]	Hodnota:	Druh funkce:
1. Správa systému	15,0	60	0,25	základní
2. Založení projektu	15,0	30	0,50	základní
3. Sestavení týmu	15,0	50	0,30	základní
4. Sestavení harmonogramu	15,0	65	0,23	základní
5. Naplánování fází projektu	15,0	30	0,50	základní
6. Úkoly a Todo listy	15,0	30	0,50	základní
7. Zobrazení celkového přehledu	8,0	10	0,80	vedlejší
8. Srovnání reality a plánu	4,0	25	0,16	podpůrná
9. Hodnocení efektivity zdrojů	1,0	20	0,05	podpůrná
10. Podrobný přehled	3,0	12	0,25	vedlejší
11. Správa zabezpečení	11,5	17	0,68	vedlejší
12. Určení pracnosti a ceny	6,0	65	0,09	podpůrná
13. Vytváření a aplikace šablon	4,0	30	0,13	podpůrná
14. Úprava milníků	10,0	12	0,83	vedlejší
15. Porady	7,5	20	0,38	podpůrná
16. Výkazy	7,5	20	0,38	podpůrná
17. Dokumenty	6,5	25	0,26	podpůrná
18. Notifikace	2,0	6	0,33	podpůrná

Odhad ceny jednotlivých funkcionalit byl proveden na základě odhadu pracnosti a vyčíslení nákladů potřebných na práci vývojářů. Odhad pracnosti provedl metodou expertního odhadu projektový manažer. Náklady na práci vývojáře poskytl finanční manažer. Tyto náklady u softwarových projektů obvykle představují hlavní položku celkových nákladů. Při výpočtu se vychází z hrubé mzdy vývojářů včetně položek, jako je zdravotní a sociální pojištění, odměny a nepravidelné složky mzdy. Dále se zahrnuje

určitá část z pořizovací ceny prostředků, které vývojář používá (odpisy). Jde zejména o výpočetní techniku a software. Mezi všechny projekty jsou ještě rozpočítány náklady provozní režie.

Pro naše účely počítáme 625 Kč za hodinu práce vývojáře. Tuto cenu násobíme odhadem pracnosti v člověkohodinách. Výsledek je uveden v tabulce Tab. 5 ve sloupci Cena.

Pokud sečteme ceny za ohodnocené funkce, dostaneme následující souhrnné výsledky:

- Celkové náklady za všechny funkce činí: 527.000,- Kč.
- Náklady za základní funkce činí: 265.000,- Kč.
- Náklady za vedlejší funkce činí: 51.000,- Kč.
- Náklady za podpůrné funkce činí: 211.000,- Kč.

Tabulka Tab. 6 obsahuje výčet funkcí seřazených podle užitečnosti, ceny a hodnoty. Ohodnocení a pořadí funkce je jedním ze vstupních kritérií při rozhodování o její implementaci.

Tab. 6: Seřazení funkcí podle jednotlivých kritérií

Seřazení podle užitečnosti:		Seřazení podle ceny: [tis. Kč]		Seřazení podle hodnoty:	
1. Správa systému	15,0	4. Sestavení harmonogramu	65	14. Úprava milníků	0,83
2. Založení projektu	15,0	12. Určení pracnosti a ceny	65	7. Zobrazení celkového přehledu	0,80
3. Sestavení týmu	15,0	1. Správa systému	60	11. Správa zabezpečení	0,68
4. Sestavení harmonogramu	15,0	3. Sestavení týmu	50	2. Založení projektu	0,50
5. Naplánování fází projektu	15,0	2. Založení projektu	30	5. Naplánování fází projektu	0,50
6. Úkoly a Todo listy	15,0	5. Naplánování fází projektu	30	6. Úkoly a Todo listy	0,80
11. Správa zabezpečení	11,5	6. Úkoly a Todo listy	30	15. Porady	0,38
14. Úprava milníků	10,0	13. Vytváření a aplikace šablon	30	16. Výkazy	0,38
7. Zobrazení celkového přehledu	8,0	8. Srovnání reality a plánu	25	18. Notifikace	0,33
15. Porady	7,5	17. Dokumenty	25	3. Sestavení týmu	0,30
16. Výkazy	7,5	9. Hodnocení efektivity zdrojů	20	17. Správa zabezpečení	0,26
17. Dokumenty	6,5	15. Porady	20	1. Správa systému	0,25
12. Určení pracnosti a ceny	6,0	16. Výkazy	20	10. Podrobný přehled	0,25
8. Srovnání reality a plánu	4,0	11. Správa zabezpečení	17	4. Sestavení harmonogramu	0,23
13. Vytváření a aplikace šablon	4,0	10. Podrobný přehled	12	8. Srovnání reality a plánu	0,16
10. Podrobný přehled	3,0	14. Úprava milníků	12	13. Vytváření a aplikace šablon	0,13
18. Notifikace	2,0	7. Zobrazení celkového přehledu	10	12. Určení pracnosti a ceny	0,09
9. Hodnocení efektivity zdrojů	1,0	18. Notifikace	6	9. Hodnocení efektivity zdrojů	0,05

5.5 Inovativní přínos

Snahou celého projektu je přinést inovaci do oblasti procesů firmy. Úspěšné využití inovačních příležitostí je spojeno s určitými kritérii, která je nutno splnit.

„Inovace neznamenaají laboratoře plné vědců, ale tvůrčí činnost ve veškerých aktivitách podniku, včetně personalistiky, obchodní strategie a všech pracovních postupů podniku. Je to styl podnikového chování, který je slučitelný s neustálým uplatňováním nových myšlenek a odpovídajících změn, i když znamenají určité riziko a nemusejí se vždy podařit. Aby inovace skutečně padala na váhu, musí proniknout celou strukturou a myšlením organizace.“ (5)

Změny, které jsou zdroji inovačních příležitostí, dělíme dle oblastí, odkud pocházejí, na:

- změny uvnitř oboru, změny uvnitř firmy,
- změny mimo obor. (7)

Ke změnám uvnitř oboru (firmy) patří nečekané události, rozpory, potřeby procesu a oborové a tržní struktury (5). Inovační příležitost nového systému podpory řízení projektů vychází z potřeb procesu. Jedná se o zdroj inovací, který zdokonaluje již existující proces a nahrazuje nějaký jeho slabý článek.

Aby byla inovace založená na změně procesu úspěšná, musí splnit následující kritéria:

- Jedná se o samostatný proces.
- Existuje slabý článek procesu.
- Existuje jasné stanovení cíle.
- Existuje možnost jasné definice řešení (existence znalostí pro úspěšné řešení).
- Obecné uvědomění si potřeby změny (účastníci procesu musejí chápat, proč je dobré proces změnit). (7)

Proces tvorby alokací a reportování práce můžeme chápat jako samostatný proces.

V procesu tvorby alokací a reportování práce je slabým článkem podpora ze strany informačního systému. Současné řešení neumožňuje reportování stavu prací na projektu v kratších intervalech a je uživatelsky nepohodlné, což vede k nechuti se systémem pracovat a prodlužuje dobu potřebnou k vytvoření a odevzdání reportů. Dlouhé intervaly mezi jednotlivými reporty znemožňují projektovému manažerovi včas reagovat na vývoj situace.

Stanoveným cílem je tedy zlepšení informační podpory řízení projektů nasazením nového systému, který by umožňoval vytvářet reporty a alokace v kratších intervalech a kde by práce se systémem byla uživatelsky přijatelnější.

Řešení spočívá v nasazení systému Blue Camp, který nejvíce odpovídal vytyčeným požadavkům.

Účastníci jsou ztotožnění s potřebou plánované změny. Zaměstnanci vytvářející reporty sami požadují přívětivější způsob práce se systémem. Zájmem projektových manažerů je pak získávání aktuálních informací včas. Očekávají také přehlednější výstupy a automatizovanou tvorbu statistik a přehledů.

5.6 Analýza rizik

Existují hrozby, které mohou mít nežádoucí vliv na nasazení a chod našeho systému pro podporu řízení projektů. Působením hrozby může vzniknout škoda. Je tedy potřeba identifikovat potencionální hrozby a odhalit slabiny systému.

5.6.1 Model změny

O nezbytnosti provedení změn bylo rozhodnuto na základě předchozích analýz. Před provedením plánovaných změn byli ještě identifikováni agent změny a intervenční oblasti. Jejich analýza je součástí procesu řízené změny a odpovídá úvodním krokům tzv. Lewinova modelu (21).

5.6.2 Agent změny

Agent změny je jednotlivec nebo skupina zaměstnanců, kteří jsou nositeli a realizátory celého procesu změny ve firmě. V našem případě je touto osobou projektový

ředitel a jako jeho zástupce jeden z projektových manažerů. Agent změny je podporován tzv. sponzorem změny, kterým je majitel firmy.

Jedním ze základních úkolů agenta změny je zjistit ochotu zaměstnanců akceptovat proces změny. Vztah zaměstnance k očekávaným změnám je dán dvěma hledisky. Prvním z nich je stupeň očekávaného osobního rizika a druhým je stupeň nespokojenosti se současným stavem. Do skupiny zaměstnanců, u kterých převládá obava z osobního rizika, patří zejména ti zaměstnanci, kteří již nyní neprovádějí své reporty v souladu s firemní směrnicí. Jejich reporty práce nebývají včasné a úplné. Do druhé skupiny zaměstnanců, u kterých převládá nespokojenost se současným stavem, patří ti, kteří reporty odevzdávají včas a správně, avšak časová režie spojená s tvorbou těchto dokumentů je nepřiměřeně vysoká. Zatímco od pracovníků druhé skupiny můžeme očekávat podporu v průběhu zavádění nového systému, u první skupiny je nejprve nutné zvýšit úroveň akceptace procesu (např. formou vhodné motivace), než začneme s realizací vlastních změn.

5.6.3 Intervence

Intervence je zásah do stávajících firemních systémů. V našem případě je směřována do oblasti lidských zdrojů a řízení a do oblasti komunikačních a organizačních toků a procesů firmy. Naopak se netýká oblasti organizační struktury firmy nebo oblasti firemních technologií.

5.6.4 Rizika spojená se změnou

Míru ohrožení aktiv, nebezpečí uplatnění hrozby a nežádoucí výsledky vedoucí ke vzniku škody vyjadřuje riziko. Rizikem chápeme situaci, v níž existuje možnost nepříznivé odchylky od žádoucího výsledku, který očekáváme. Riziko můžeme snížit navržením vhodných protiopatření, která zmírňují působení hrozby, snižují zranitelnost systému a dopad hrozeb.

Při navrhování protiopatření se používá pravidlo, které stanovuje, že náklady vynaložené na snížení rizika musí být přiměřené hodnotě chráněných aktiv (nebo hodnotě škod vzniklých dopadem hrozby). Pomocí tohoto pravidla se určuje referenční úroveň rizika, která vyjadřuje hranici míry rizika. Pokud je velikost rizika menší než

referenční úroveň, pak riziko označíme za zbytkové. Takové riziko je již přijatelné a není nutné podnikat žádné další protiopatření pro jeho snížení (21).

Následující přehled obsahuje nalezené hrozby, zhodnocení míry rizika a navržená protiopatření. Pro vyjádření rizik a určení jejich závažnosti byla použita kvalitativní metoda analýzy rizik. Tato metoda vyjadřuje rizika v intervalu, pravděpodobností nebo slovně. Úroveň rizik je určována kvalifikovaným odhadem.

Odmítnutí systému – Zaměstnanci systém odmítnou používat a nebudou vytvářet reporty.

Tato hrozba vychází z reálných zkušeností s předchozím systémem, kdy někteří zaměstnanci odevzdávali své reporty pozdě, nebo je neodevzdávali vůbec.

Pravděpodobnost výskytu: Nízká (4 – 7 % pracovníků odmítalo pracovat se systémem nebo pracovalo nesprávným způsobem).

Závažnost dopadu: Vysoká (odmítnutí systému může vedle chybějících dat způsobovat také demotivaci ostatních pracovníků).

Pro snížení této hrozby je nutné zaměstnance k práci se systémem motivovat. Zaměstnanci mohou být motivováni tím, že jim nový systém usnadní práci, kterou museli dříve provádět prostřednictvím nevhodných nástrojů. Pokud by i nadále odmítali správně používat nasazený systém, pak na ně mohou být uplatněny restrikce finanční (např. v podobě snížení prémie).

Nepravdivé reporty práce – Zaměstnanci ve snaze „vylepšit“ své pracovní nasazení mohou odevzdávat nepravdivé reporty.

Pravděpodobnost výskytu: Vysoká (reporty provedené práce se lišily od skutečného stavu přibližně o 10 – 30 %).

Závažnost dopadu: Nízká (nepřesné údaje zkreslují výsledky, ale na rozdíl od reportování míry dokončení úkolu nejsou kritické pro řízení projektu).

Je potřebné oddělit reportování práce od reportování stavu projektu. Pro řízení projektu totiž není tak důležité, kolik hodin daný pracovník pracoval, ale spíše nás zajímá míra dokončení úkolu. Nepřesné reporty práce by neměly ovlivňovat zprávy o stavu projektu. Oddělení reportů práce od reportů stavu projektu však neznamena návrat k původnímu řešení, kde bylo k těmto účelům používáno rozdílných dokumentů a nástrojů. Nové řešení používá jednotné rozhraní pro společné zadávání a odesílání obou typů reportů, avšak zdůrazněn je hlavně report stavu a report práce je podružný. Vedoucí pracovníci by měli provádět namátkové kontroly skutečně provedené práce a případné nedostatky by měli promítnout do snížení premii zaměstnance.

Nepravdivé reporty stavu – Zaměstnanci mohou buď záměrně, nebo neúmyslně odevzdávat nepřesné reporty stavu úkolu.

Neúmyslně vzniklé odchylky jsou způsobeny složitostí daného oboru (vývoj softwarových systémů) a nebo špatným odhadem nezkušených pracovníků.

Úmyslné odchylky jsou buď výsledkem nadhodnocování provedené práce, kdy je také chybně navýšena míra dokončení projektu, nebo se jedná o úmyslné snižování míry dokončení projektu, což pracovníkovi může zajistit více času na dokončení úkolu.

Pravděpodobnost výskytu: Střední (reporty stavu projektu se lišily od skutečného stavu přibližně o 15 – 20 %).

Závažnost dopadu: Střední (nepřesné údaje zkreslují výsledky a ovlivňují řízení projektu).

Obě varianty vzniku chybných reportů může projektový manažer řešit pomocí statistických nástrojů, porovnáním reportovaného stavu s kvalifikovaným odhadem doby potřebné pro realizaci úkolu, nebo namátkovou kontrolou skutečného stavu. Dle příčiny problému pak může volit buď úpravu svých dat, nebo restrikcí vůči zaměstnanci.

Rozsah úkolů – Každý projektový manažer nebo vedoucí týmu může zadávat různě velké úkoly.

Úkoly s výrazně rozdílným rozsahem mohou uživateli znesnadňovat práci se systémem. Velké úkoly znesnadňují kvalifikovaný odhad míry dokončení a malé úkoly vyvolávají zbytečně vysoký podíl administrativní činnosti spojené s reportováním a evidencí úkolů.

Pravděpodobnost výskytu: Nízká (přibližně 15 % úkolů se svým rozsahem výrazněji liší).

Závažnost dopadu: Nízká (rozdílný rozsah úkolů pouze znesnadňuje práci se systémem, ale nemá významný vliv na přesnost výsledků).

Snížit riziko spojené s rozdílným rozsahem úkolů můžeme stanovením a zavedením metodiky, která by definovala optimální velikost úkolů pro danou oblast pracovního procesu.

Únik informací – Systém shromažďuje informace, které by mohly být zneužity třetí stranou.

Útočník, který se zmocní dat ze systému, by mohl získat informace o projektech firmy, informace o připravovaných novinkách ve vyvíjeném softwaru, informace o případných problémech s projekty, nebo také informace o schopnostech a profilu všech zaměstnanců společnosti.

Pravděpodobnost výskytu: Nízká (vychází ze statistiky bezpečnostních incidentů, ke kterým ve firmě v průběhu její existence došlo).

Závažnost dopadu: Vysoká (údaje o přesném stavu firemních projektů by mohly být zneužity konkurencí).

Většinu datových úniků mají obvykle na svědomí vlastní zaměstnanci (20). V případě systému pro řízení projektů má však přístup k ucelenému souboru dat jen malá skupina lidí (spíše jen jednotlivci). Významnější je tedy například hrozba ze strany provozovatele systému, zejména jedná-li se o potencionálně konkurenční firmu (z oblasti vývoje softwaru).

Snížení rizika úniku informací můžeme zajistit provozem systému na vlastní infrastruktuře, kterou spravují námi prověření administrátoři. Bezpečnostní rizika související s novým systémem by měla být zohledněna v bezpečnostních pravidlech firmy v souladu s bezpečnostní strategií firmy.

Výpadek systému – Výpadek systému může být způsoben výpadkem serveru, na kterém systém běží.

Výpadek může být spojen se ztrátou dat, nebo se může jednat pouze o dočasnou nedostupnost služby. V případě dočasné nedostupnosti nemohou uživatelé se systémem pracovat. To pozdrží vytváření alokací, zadávání úkolů a tvorbu reportů. Pokud při výpadku dojde ke ztrátě dat, pak jsou výrazně omezeny funkce, které používají uložená data. Jedná se zejména o statistiky a přehledy, ale značně omezeny jsou i funkce operativního řízení aktuálně běžících projektů.

Pravděpodobnost výskytu: Nízká (firma používá pro chod systémů robustní infrastrukturu s vysokou stabilitou).

Závažnost dopadu: Vysoká (v závislosti na délce výpadku a míře poškození dat).

Krátkodobý výpadek dostupnosti systému (v řádu hodin) lze považovat vzhledem k délce úkolů za zanedbatelný. Výpadky trvající jeden až dva dny je možné překlenout použitím dalších komunikačních kanálů, prostřednictvím kterých lze operativně zadat úkol či oznámit stav prací na projektu. Delší výpadky by již byly z hlediska řízení projektu problematické. Jeden až dva dny ale dostačují pro nasazení záložního systému nebo odstranění závady. Proti ztrátám dat způsobených výpadkem systému se můžeme bránit pravidelným zálohováním.

Komplexnost UC – Hrozba plynoucí z modulu pro výpočet doby realizace projektu na základě případů užití.

Analytici mohou různě hodnotit velikost případů užití (use cases), která je dána atributem komplexnosti. Komplexnost jako proměnná vstupuje do algoritmu, který na základě dalších vlastností systému a prostředí počítá dobu potřebnou k realizaci

projektu. Pokud analytici nepoužívají jednotné hodnocení komplexnosti, pak se mohou výsledky projektů různých analytiků výrazně lišit.

Pravděpodobnost výskytu: Nízká (5 – 10 % případů užití mívá špatně určenou komplexnost).

Závažnost dopadu: Střední (nepřesné výsledky výpočtu UCP mohou ovlivnit plán projektu, ale nejedná se o jedinou metodu plánování).

Snížit riziko spojené s rozdílným ohodnocením komplexnosti případů užití můžeme stanovením a zavedením metodiky, která by definovala jednotnou kvantifikaci. Měly by být zpracovány vzorové příklady a provedeno proškolení všech analytiků společnosti.

5.7 Harmonogram nasazení

Pro úspěšné nasazení nového systému je ještě potřeba naplánovat scénář nasazení systému, zajistit zdroje (které zrealizují potřebné operace a úkoly), připravit infrastrukturu a podpůrné systémy a naplánovat zaškolení uživatelů.

5.7.1 Alokace pracovníků

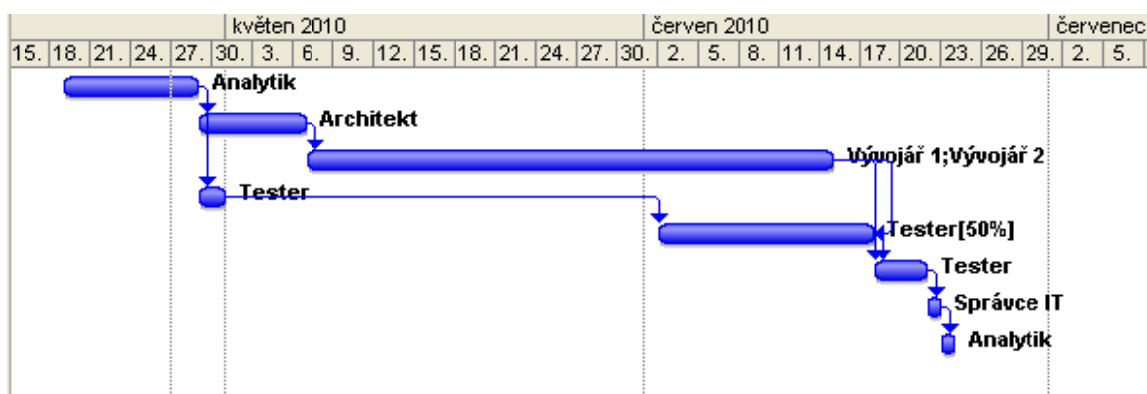
Na základě analýzy a předběžného časového plánu byl vytvořen požadavek na alokaci níže uvedeného množství zdrojů. Alokovány jsou v této fázi pouze role (analytik, architekt, tester apod.). Konkrétní zaměstnanci budou do požadovaných rolí přiděleni až dle aktuální situace v době řešení vytvářeného systému.

Jednotlivé role budou při vývoji systému potřebné v různých fázích projektu. Fáze projektu a alokace zdrojů jsou obsahem tabulky Tab. 7.

Tab. 7: Harmonogram projektu a alokované zdroje

		Název úkolu	Doba trvání	Zahájení	Dokončení	Předchůdci	Názvy zdrojů
1		Analýza	8 dny	19.4. 10	28.4. 10		Analytik
2		Návrh	6 dny	29.4. 10	6.5. 10	1	Architekt
3		Vývoj	27 dny	7.5. 10	14.6. 10	2	Vývojář 1; Vývojář 2
4		Testovací scénáře	2 dny	29.4. 10	30.4. 10	1	Tester
5		Testování průběžné	12 dny	2.6. 10	17.6. 10	4;3FF	Tester[50%]
6		Testování akceptační	2 dny	18.6. 10	21.6. 10	3;5	Tester
7		Nasazení	1 den	22.6. 10	22.6. 10	6	Správce IT
8		Školení	1 den	23.6. 10	23.6. 10	7	Analytik

Časový harmonogram odráží ideální průběh realizace systému. V našem případě se jedná o projekt interní, takže dle zkušeností z praxe lze očekávat, že požadované zdroje nebudou plně k dispozici. Uvolnění zdrojů pro tento projekt bude závislé na ostatních projektech realizovaných pro externí zákazníky. Celková doba trvání projektu se tedy s velkou pravděpodobností prodlouží. Stále by však měly platit návaznosti jednotlivých úkolů uvedených v Ganttově diagramu na obrázku Obr. 7.



Obr. 7: Harmonogram projektu v Ganttově diagramu

Časový plán odpovídá realizaci všech navržených funkcí. Pokud by některá z funkcí na základě rozhodnutí managementu nebyla realizována vůbec, pak to bude mít opět vliv na celkovou délku projektu. Funkcionality na sobě ve většině případů nejsou vzájemně závislé, takže je možné provést realizaci vybrané podmnožiny z navržených funkcí.

5.7.2 Alokace prostor a dalších zdrojů

Jednotliví pracovníci, kteří se budou podílet na vývoji systému, využívají k práci prostory svých kanceláří a osobní počítače s příslušnými softwarovými nástroji. Tyto prostředky není tedy nutné zvlášť alokovat.

Mezi zdroje, které je nutné pro vývoj systému uvolnit, patří:

Testovací server – Samostatné prostředí pro testovací běh systému je realizováno pomocí virtuálního serveru MS Windows Server 2003, databáze MS SQL Server 2005, .NET frameworku a aplikačního serveru IIS (případně frameworku JDK a aplikačního serveru GlassFish, WebSphere nebo JBoss - upřesněno bude ve fázi návrhu architektury).

Server pro nasazení a běžný provoz systému – Pro běžný provoz bude systém nasazen na serveru s podobnou konfigurací, jakou měl server testovací. Server však již nebude virtuální a budou na něm uplatněny politiky zálohování, řízení přístupu apod.

5.7.3 Školení

Pro seznámení s novými funkcemi systému Blue Camp je naplánováno jedno školení v délce trvání 3 hodin. Školení bude provedeno v rámci tří skupin vzhledem ke kapacitě místnosti zvolené pro dané školení. Vzhledem ke skutečnosti, že většina zaměstnanců firmy má informatické vzdělání, je délka školení považována za dostatečnou. Také větší část školení bude vzhledem k technickým schopnostem zaměstnanců věnována spíše pravidlům používání systému než vysvětlování jednotlivých funkcí. Popis těchto funkcí bude dostupný v manuálu a nápovědě systému. Průběh jednoho školení je rozdělen do tří částí:

1,5 hod - vysvětlení pravidel použití systému

0,5 hod - přehled funkcí systému

1 hod – dotazy a diskuze

6 MODUL UCP

Jak bylo uvedeno v předchozích kapitolách, uvažovaný systém pro správu projektů potřebuje některá rozšíření, která zajistí jeho vyšší použitelnost v procesu řízení projektu. Z množiny navržených rozšíření byl pro realizaci v rámci této práce vybrán modul zajišťující tvorbu odhadů pracnosti u projektů zaměřených na vývoj softwaru. Modul pracuje na základě metody Use Case Points, která byla popsána v teoretické části práce. Podle použité metody dostal tento modul pracovní název *Use Case Plan*. Před popisem technických aspektů je vhodné uvést příklad typového použití modulu.

6.1 Aplikace metody

Ukázkový odhad pracnosti metodou UCP provedeme na projektu, ve kterém realizujeme informační systém pro komunikaci s dodavateli. Odhad pracnosti je většinou prováděn nad celým projektem. V uvedeném typovém případě je však pro názornost zpracován pouze jeden element analytického modelu. Tímto elementem je jeden případ užití. V uvedeném systému je typickým případem užití například založení dodavatele.

Založení dodavatele provádí aktér „Správce dodavatelů“. Tento aktér k systému přistupuje přes grafické uživatelské rozhraní. Podle metody UCP je tedy ohodnocen vahou 3, spadá totiž do množiny komplexních aktérů.

Případ užití „Založení nového dodavatele“ je popsán tímto scénářem:

Hlavní scénář:

1. *Případ užití zahajuje uživatel kliknutím na volbu „Nový dodavatel“*
2. *Systém zkontroluje práva uživatele*
3. *Systém otevře formulář pro založení dodavatele*
4. *Uživatel vepíše do formuláře IČ, název dodavatele, kontaktní informace a vybere volbu „Založit nového dodavatele“*
5. *Systém ověří IČ dodavatele*
6. *Systém založí dodavatele*
7. *Systém vypíše založeného dodavatele*

Alternativní scénář:

2. *Uživatel nemá dostatečné oprávnění k provedení operace*
 - a. *Systém operaci ukončí a vypíše chybové hlášení*

Alternativní scénář:

5. *Dodavatel v systému již existuje*
 - a. *Systém operaci ukončí a vypíše chybové hlášení*

Dle počtu scénářů spadá případ užití do množiny jednoduchých případů užití a je tedy ohodnocen vahou 5. Alternativní scénáře nejsou dle rad expertů započítávány. Také se neuvažují případy užití připojené vazbou *include* nebo *extend*. Hlavní scénář pak představuje jednu transakci.

Nyní máme k dispozici hodnoty pro výpočet argumentu UUCP. Jelikož uvedený aktér figuruje v celém systému, nebudeme ho ve výpočtu jednoho případu užití uvažovat. Argument UUCP je tedy roven hodnotě 5.

Pro realizovaný systém bychom měli mít připravenou tabulku technických faktorů a faktorů prostředí. Váhy jsou dané metodou UCP a vliv faktorů nastaví expert (architekt, vedoucí technolog nebo vedoucí programátor). Hodnoty pro naši typovou úlohu máme uvedené v tabulce Tab. 8.

Tab. 8: Ohodnocení technických faktorů a faktorů prostředí

Faktor	Vliv	Váha	Vážený vliv T	Faktor	Vliv	Váha	Vážený vliv E
<i>T1</i>	2	2,0	4,0	<i>E1</i>	3	1,5	4,5
<i>T2</i>	1	2,0	1,0	<i>E2</i>	4	0,5	2,0
<i>T3</i>	4	1,0	4,0	<i>E3</i>	4	1,0	4,0
<i>T4</i>	1	1,0	1,0	<i>E4</i>	5	0,5	2,5
<i>T5</i>	2	1,0	2,0	<i>E5</i>	2	1,0	2,0
<i>T6</i>	5	0,5	2,5	<i>E6</i>	4	2,0	8,0
<i>T7</i>	5	0,5	2,5	<i>E7</i>	0	-1,0	0,0
<i>T8</i>	0	2,0	0,0	<i>E8</i>	3	-1,0	-3,0
<i>T9</i>	2	1,0	2,0				
<i>T10</i>	4	1,0	4,0				
<i>T11</i>	3	1,0	3,0				
<i>T12</i>	2	1,0	2,0				
<i>T13</i>	1	1,0	1,0				
Celkem:			29,0				20,0

Na základě hodnot z tabulky Tab. 8 jsme schopni vypočítat argumenty TCF a EF:

$$TCF = 0,6 + (0,01 \cdot 29) = 0,89$$

$$EF = 1,4 + (-0,03 \cdot 20) = 0,8$$

Z hodnot argumentů TCF, EF a UUCP již vypočítáme počet bodů UCP:

$$UCP = UUCP \cdot TCF \cdot EF = 5 \cdot 0,89 \cdot 0,8 = 3,56 \text{ bodů}$$

Případ užití tedy představuje 3,65 *bodů UCP*. Vypočítaný počet bodů nám ještě nic neříká o očekávané pracnosti. Pro výpočet pracnosti musíme ještě určit faktor produktivity PF. Zde většinou vycházíme z vlastních již zrealizovaných projektů nebo doporučení expertů. Jelikož se jedná o malý systém, dosadíme podle (16) za faktor produktivity 15 *člověkohodin / UCP*.

$$Pracnost = UCP \cdot PF = 3,56 \cdot 15 = 53,4 \text{ člověkohodin}$$

Odhadnutá pracnost na realizaci funkcionality pro založení dodavatele je tedy 53,4 *člověkohodin*. Do uvedené pracnosti jsou zahrnuty všechny činnosti počínaje návrhem architektury přes implementaci až po testování a nasazení.

Faktor produktivity nemusíme získávat pouze ze starých projektů nebo odhadů expertů, ale lze použít i výpočet založený na technických faktorech a faktorech prostředí. Algoritmus výpočtu navrhli autoři Schneider a Winters (25). Po aplikaci tohoto algoritmu byla zjištěna hodnota faktoru produktivity ve výši 20 *člověkohodin / UCP*. Po dosazení do vzorce pro výpočet pracnosti pak získáváme výsledek 71,2 *člověkohodin*.

Provedené odhady se od sebe liší o 25 %. Vliv na tento rozdíl měl pouze faktor produktivity. Určení faktoru produktivity se podle (16) jeví jako nejméně přesné, protože vychází jen z vágní definice velikosti systému. Faktor produktivity vypočítaný podle algoritmu Schneider Winters vychází z technických faktorů a z faktorů prostředí. Tyto faktory jsou expertem nastaveny pro konkrétní systém. Určený faktor produktivity by tedy měl lépe odpovídat realitě. Pokud však není algoritmus nijak modifikován, pohybujeme se neustále v rozmezí pouhých tří hodnot, které je možné dosadit za faktor produktivity. Třetí možností je využití historických dat z předchozích projektů. Faktor produktivity určený z těchto dat pak nejlépe odpovídá schopnostem konkrétního realizačního týmu.

Každý z uvedených přístupů určení faktoru produktivity přináší určitou výhodu, avšak sám o sobě ještě nezajišťuje přesný odhad pracnosti. Doporučována je tedy jejich kombinace - zvážení výsledků všech tří možných přístupů.

6.2 Popis modulu UCP

Tato kapitola obsahuje technický popis realizovaného modulu Use Case Plan pro odhad pracnosti projektu. Součástí popisu jsou diagramy v jazyce UML (*Unified Modeling Language*).

K zachycení činností prováděných uživatelem nebo externím systémem byly použity diagramy případů užití (*Use Case Diagrams*). V diagramu je zachycen aktér a související případy užití, které aktér vyvolal. Vztah mezi aktérem a případem užití je

vyjádřen vazbou. Mezi případy užití jsou pak používány vazby *include* a *extend*. Rozšíření případu užití (stereotyp <<extend>>) umožňuje postupně rozšiřovat obecně definované případy užití o detaily a provádět jejich specializaci. Slouží pro modelování volitelných částí. Rozšiřující případ užití nemusí být spuštěn ani jednou. Šipka směřuje od rozšiřujícího (speciálního) případu užití k obecnějšímu rozšiřovanému případu užití. Vložení případu užití (stereotyp <<include>>, <<uses>>) znamená zahrnutí případu užití do jiného případu užití. Umožňuje použít kroky jednoho případu užití v jiném případě užití. Vkládaný případ užití musí být spuštěn alespoň jednou. Šipka směřuje od případu užití, který vkládá, k případu užití, který je vkládán. Diagramy případů užití jsou použity na obrázcích Obr. 8 a Obr. 9.

Pro zachycení algoritmu výpočtu byly použity diagramy aktivit (*Activity Diagrams*). Tyto diagramy popisují jednotlivé kroky procesů. Obsahují jeden výchozí bod a zpravidla jeden i více bodů koncových. Hlavním cílem diagramu je zachytit souslednosti jednotlivých činností. Diagram obsahuje podmíněné cesty, kde rozhodovací podmínka je znázorněna symbolem kosočtverce a zpravidla bývá doplněna ještě textovým vyjádřením obsahu podmínky. Diagramy aktivit jsou použity na obrázcích Obr. 10, Obr. 11 a Obr. 12.

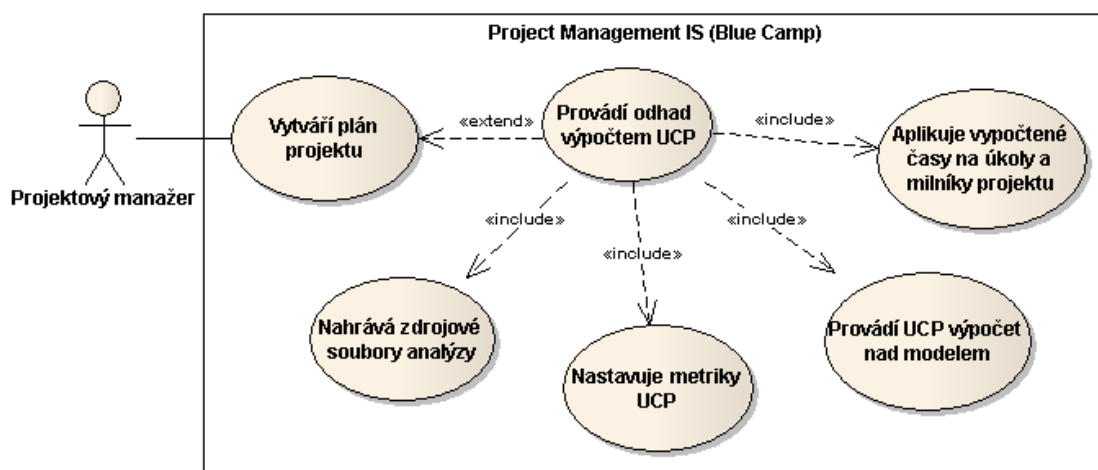
Mezi další použité diagramy patří diagram balíků a diagram tříd. Diagram tříd představuje statický pohled na entity zastoupené v systému. Popisuje vlastnosti (atributy) jednotlivých entit (tříd) a zároveň zobrazuje vztahy mezi jednotlivými třídami. V horním poli každé třídy je zapsán její název. Střední část popisuje vlastnosti třídy a spodní pole obsahuje seznam operací, které třída umí vykonávat. Diagram balíků pak zachycuje sdružení tříd do skupin podle jejich vlastností, účelu použití, případně jiných kritérií. Uvedené diagramy jsou použity na obrázcích Obr. 13 a Obr. 14.

Rozložení jednotlivých softwarových komponent na hardwarových zdrojích a jejich spolupráci zobrazuje diagram nasazení. Diagram nasazení je použit na obrázku Obr. 15.

Interakce mezi objekty jsou zachyceny na sekvenčních diagramech. Sekvenční diagramy umožňují namodelovat interakci mezi objekty v čase. Diagram je použit na obrázku Obr. 16.

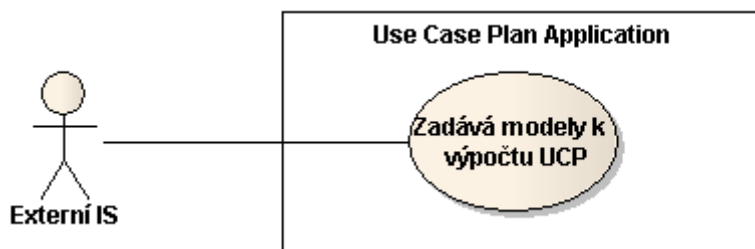
Pro názornost byl použit i rozšířený diagram uživatelského rozhraní, který zachycuje základní fragmenty budoucího rozhraní uživatele. Jsou zde schematicky zachycena okna programu a obsažené ovládací prvky. Diagram je doplněn o elementy z diagramu případů užití a z diagramu tříd, aby názorněji ukázal vazbu jednotlivých prvků na související funkcionalitu či datové pozadí. Diagram uživatelského rozhraní je použit na obrázku Obr. 17.

Jednou z činností projektového manažera je sestavení plánu projektu. Tvorba plánu projektu v systému Blue Camp je zachycena na obrázku Obr. 8. Při tvorbě projektového plánu je možné použít odhad pracnosti na základě výpočtu bodů UCP. Pokud se rozhodne projektový manažer využít tuto možnost, nahraje do systému soubor s analýzou, nastaví potřebné metriky, nechá provést výpočet a zpracuje výsledky. Zpracováním výsledků se rozumí jejich aplikace na úkoly a milníky projektu.



Obr. 8 Činnosti manažera v systému Blue Camp

Veškerou funkcionalitu potřebnou pro tvorbu plánu realizuje systém Blue Camp. Samotný výpočet bodů UCP je pak realizován modulem Use Case Plan. Modul Use Case Plan funguje jako relativně samostatná aplikace. Obsahuje rozhraní, přes které lze tento modul použít v libovolném externím systému. Schematicky je spolupráce mezi modulem a externím systémem zachycena na obrázku Obr. 9.

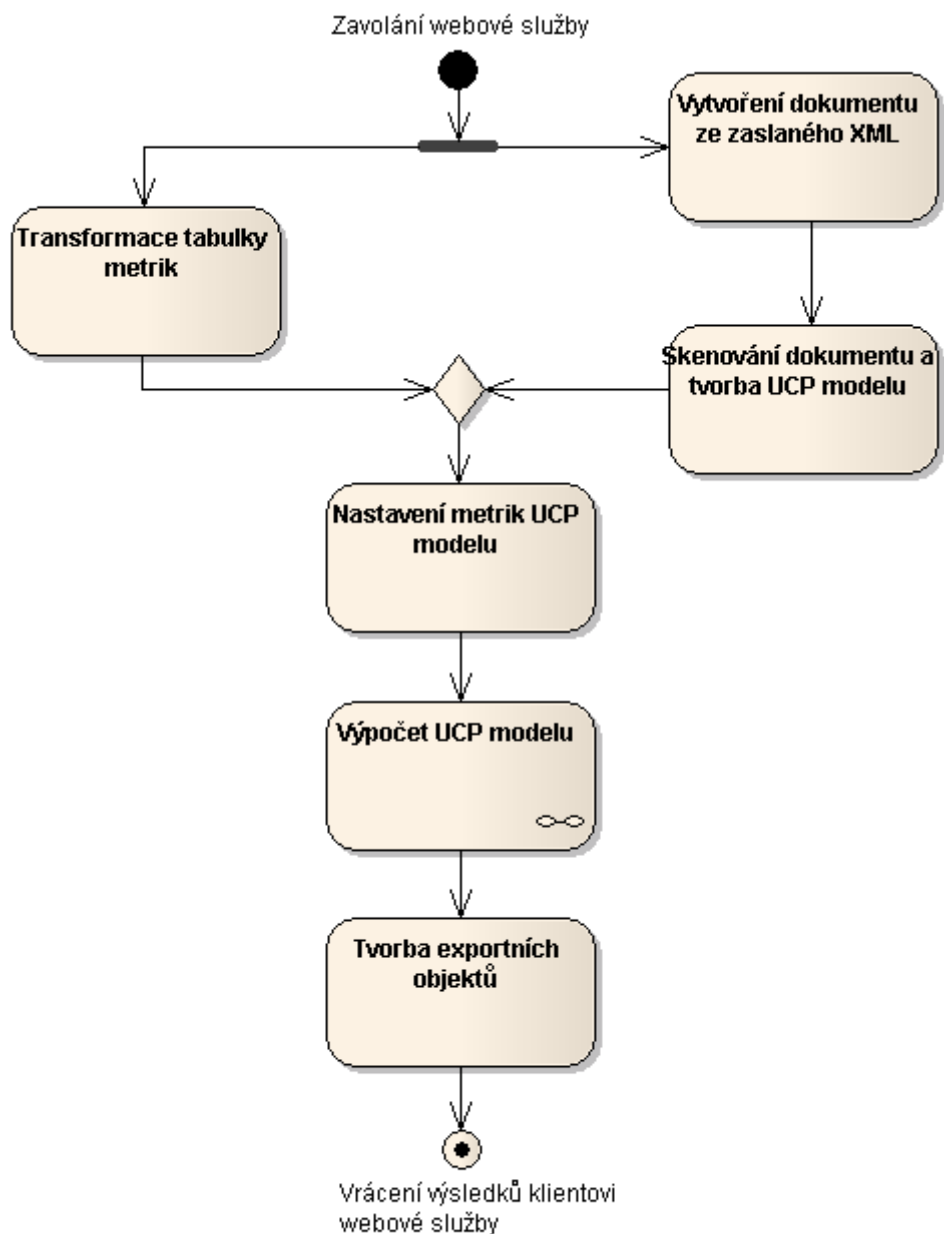


Obr. 9 Schematické znázornění použití modulu Use Case Plan

Modul Use Case Plan byl záměrně realizován tak, aby jej bylo možné používat různými systémy vytvořenými na různých platformách. Z toho důvodu byla pro komunikaci mezi modulem a externími systémy zvolena technologie webových služeb. Technologické aspekty budou popsány dále v textu. Nyní se však zaměříme na algoritmus, který realizuje vlastní výpočet.

Užití modulu Use Case Plan je zahájeno zasláním dotazu na webovou službu a končí vrácením výsledků. Vstupem pro modul je XML soubor, který obsahuje analýzu vytvářeného softwarového systému. Analýza musí obsahovat zejména model případů užití. Společně s analýzou je předávána také tabulka s nastavenými metrikami potřebnými pro výpočet bodů UCP. Výsledek vrácený webovou službou obsahuje tabulku elementů s vypočítanými hodnotami. Základní schéma algoritmu pro výpočet je uvedeno na obrázku Obr. 10.

Jakmile obdrží webová služba požadovaná vstupní data, zahájí transformaci tabulky metrik a transformaci XML souboru s analýzou. Transformovaný soubor analýzy je následně skenován a v průběhu skenování je postupně vytvářen tzv. UCP model. UCP model tvoří hlavní datovou základnu pro výpočet UCP bodů. Jakmile je UCP model vytvořen, jsou mu nastaveny metriky z tabulky metrik. Po nastavení metrik je spuštěn proces vlastního výpočtu. Výsledky výpočtu jsou transformovány do exportních objektů a jsou zaslány zpět externímu systému.



Obr. 10 Základní schéma výpočtu bodů UCP

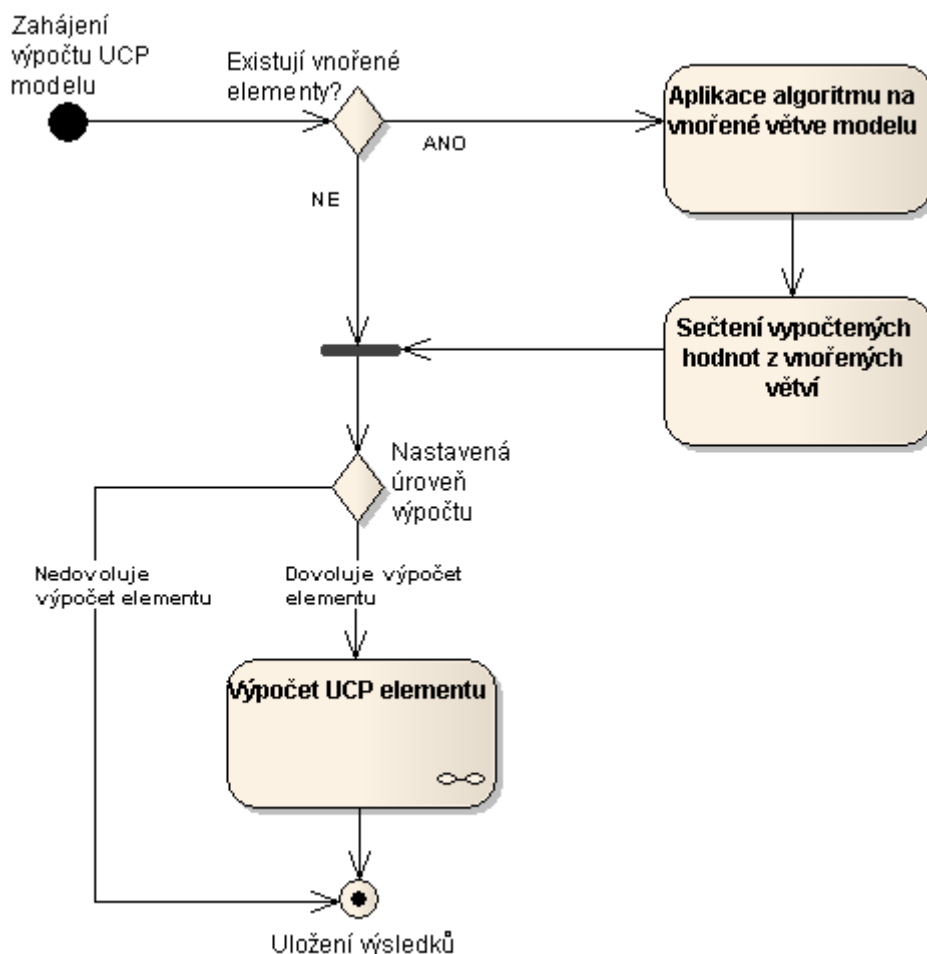
Vlastní výpočet UCP modelu je zachycen na obrázku Obr. 11. UCP model odpovídá struktuře modelu případů užití. Tento model může být složen z balíků, případů užití a aktérů. Balíky mohou obsahovat další vnořené balíky, případy užití a aktéry. Případy užití pak mohou být definovány jako tzv. *kompozitní objekty* a mohou obsahovat další vnořené případy užití. Dle teorie grafů odpovídá struktura modelu stromu. Elementy UCP modelu jsou zachyceny na obrázku Obr. 14.

Jakmile je UCP model sestaven, můžeme přejít k jeho výpočtu. Výpočtem UCP modelu rozumíme aplikaci algoritmu metody UCP na jednotlivé elementy modelu. Každý element obsahuje tabulku metrik, ve které jsou uloženy technické faktory, faktory prostředí, faktor produktivity a další hodnoty potřebné k výpočtu pracnosti. Vedle těchto hodnot se do tabulky ukládají i průběžné výsledky výpočtů. Tabulka v daném elementu obsahuje také sumární výsledky za elementy vnořené. Při výpočtu jsou nejprve zpracovány vnořené elementy. Výsledky získané z vnořených elementů jsou pak použity ve výpočtu elementu nadřazeného (kořenového).

Tento přístup přináší vyšší granularitu výpočtu. Neprovádíme pouze jeden výpočet pro model jako celek, ale zvlášť jsou prováděny výpočty nad všemi elementy modelu. Projektovému manažerovi to umožňuje zjistit odhad pracnosti pro libovolnou část projektu.

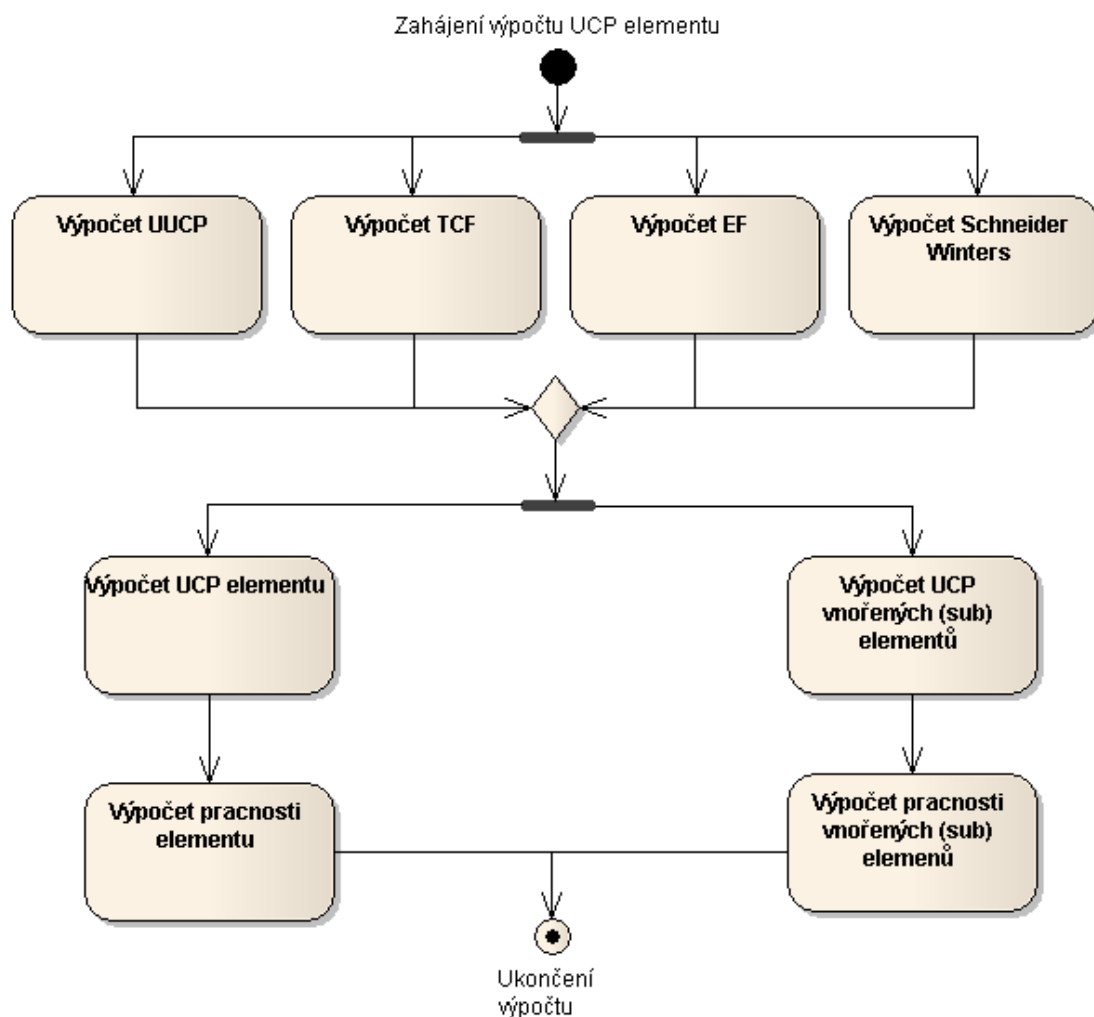
Modul je také implementován tak, aby mohl být v budoucnu rozšířen a tabulky metrik bylo možné definovat zvlášť pro libovolný element. Každá část systému tak může být ohodnocena jinými faktory, které lépe odpovídají dané části systému či realizačnímu týmu, který tuto část bude zpracovávat.

V modulu lze nastavit i hloubku výpočtu. Pokud by byl model moc rozsáhlý, lze za účelem snížení nároků na výpočetní výkon nastavit nižší hloubku výpočtu. Nastavení se provádí na straně poskytovatele webové služby. Zatímco projektového manažera nároky na výkon nezajímají, u provozovatele služby je tomu naopak. Pokud by byl modul poskytován i jiným externím systémům, pak může být nastavení hloubky výpočtu jedním z prodejních parametrů služby.



Obr. 11: Schéma výpočtu UCP modelu

Algoritmus výpočtu jednoho elementu UCP modelu je znázorněn na obrázku Obr. 12. Nejprve jsou vypočteny hodnoty UUCP, TCF, EF a faktor produktivity dle algoritmu Schneider Winters. Poté následuje výpočet UCP bodů a výpočet pracnosti pro daný element. Paralelně lze zpracovat hodnoty za vnořené elementy. Pokud nastavená hloubka nedovolila provést výpočet pracností již na vnořených elementech, je tento výpočet proveden až zde při zpracování elementu nadřazeného. Přitom se jako vstup výpočtu pro vnořené elementy bere tabulka metrik (faktory) elementu nadřazeného.



Obr. 12: Schéma výpočtu UCP elementu

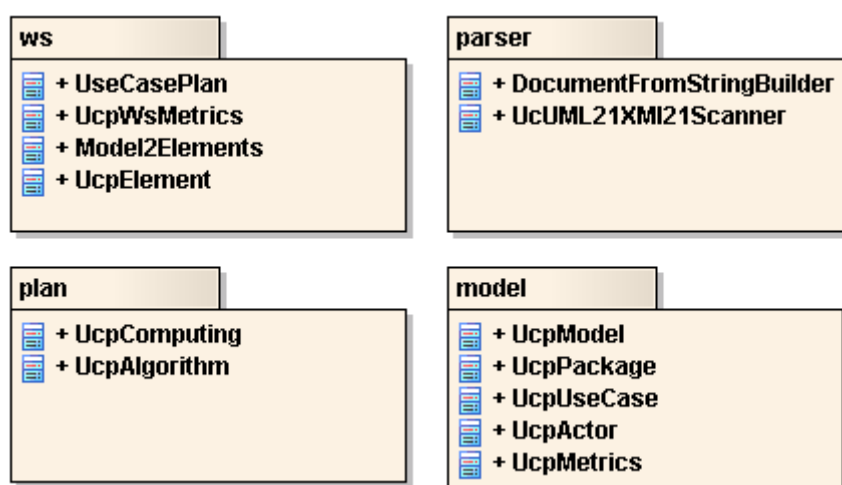
Jakmile je výpočet modelu dokončen, získáme jednak souhrnné výsledky za celý projekt, ale také pro každý element (v závislosti na nastavené hloubce výpočtu) získáme podrobné výsledky s počty UCP bodů a odpovídající pracností.

Třídy realizující funkčnost modulu, jsou rozděleny do čtyř balíků. Rozdělení je zachyceno na obrázku Obr. 13. První balík s názvem *ws* obsahuje třídy, které zajišťují chod webové služby. Třída *UseCasePlan* definuje operace poskytnuté webovou službou. Třídy *UcpWsMetrics* a *UcpElement* reprezentují přenášená data. Třída *Model2Element* zajišťuje transformaci UCP modelu do exportních objektů typu *UcpElement*.

Třídy z balíku *parser* zajišťují zpracování vstupního souboru s analýzou realizovaného systému.

Třídy z balíku *plan* zajišťují výpočty UCP modelu. Třída *UcpComputing* řídí celý proces výpočtu. Třída *UcpAlgorithm* obsahuje vlastní algoritmus výpočtu UCP bodů a pracnosti. Poskytuje metody pro zpracování tabulek metrik.

Balík *model* obsahuje třídy reprezentující elementy UCP modelu. Třída *UcpModel* reprezentuje vlastní model UCP. Třídy *UcpPackage*, *UcpUseCase* a *UcpActor* reprezentují jednotlivé elementy modelu případů užití, který tvoří hlavní část UCP modelu. Třída *UcpMetrics* reprezentuje tabulku metrik.



Obr. 13: Rozdělení tříd do balíků

Jádro aplikace tvoří třídy balíku *model*. Představují datovou základnu, nad kterou jsou prováděny veškeré výpočty. Detaily těchto tříd a jejich vzájemné závislosti jsou zachyceny na diagramu Obr. 14.

Třída *UcpMetrics* reprezentuje tabulku metrik. Tato tabulka obsahuje vstupní data pro výpočet, mezivýsledky a výsledky výpočtů. Mezi data označená jako vstupní můžeme zařadit technické faktory, faktory prostředí, faktor produktivity a komplexnost elementu. Ostatní položky tabulky slouží jako mezivýsledky nebo konečné výsledky.

Atributy *t* a *e* značí pole technických faktorů a faktorů prostředí. Atribut *schneiderwinters* představuje faktor produktivity vypočtený podle metody Schneider Winters. Faktor produktivity zadaný projektovým manažerem je uložen v atributu *hoursPerUc*. Komplexnost elementu je uložena v atributu *complexity* a vypočtené

množství UCP bodů v atributu *ucp*. Atributy *hours* a *hoursWS* nesou výsledek pracnosti. První uvedená pracnost je vypočtena s použitím faktoru produktivity zadaného projektovým manažerem. Druhá uvedená pracnost je vypočtena s použitím faktoru produktivity získaného dle metody Schneider Winters.

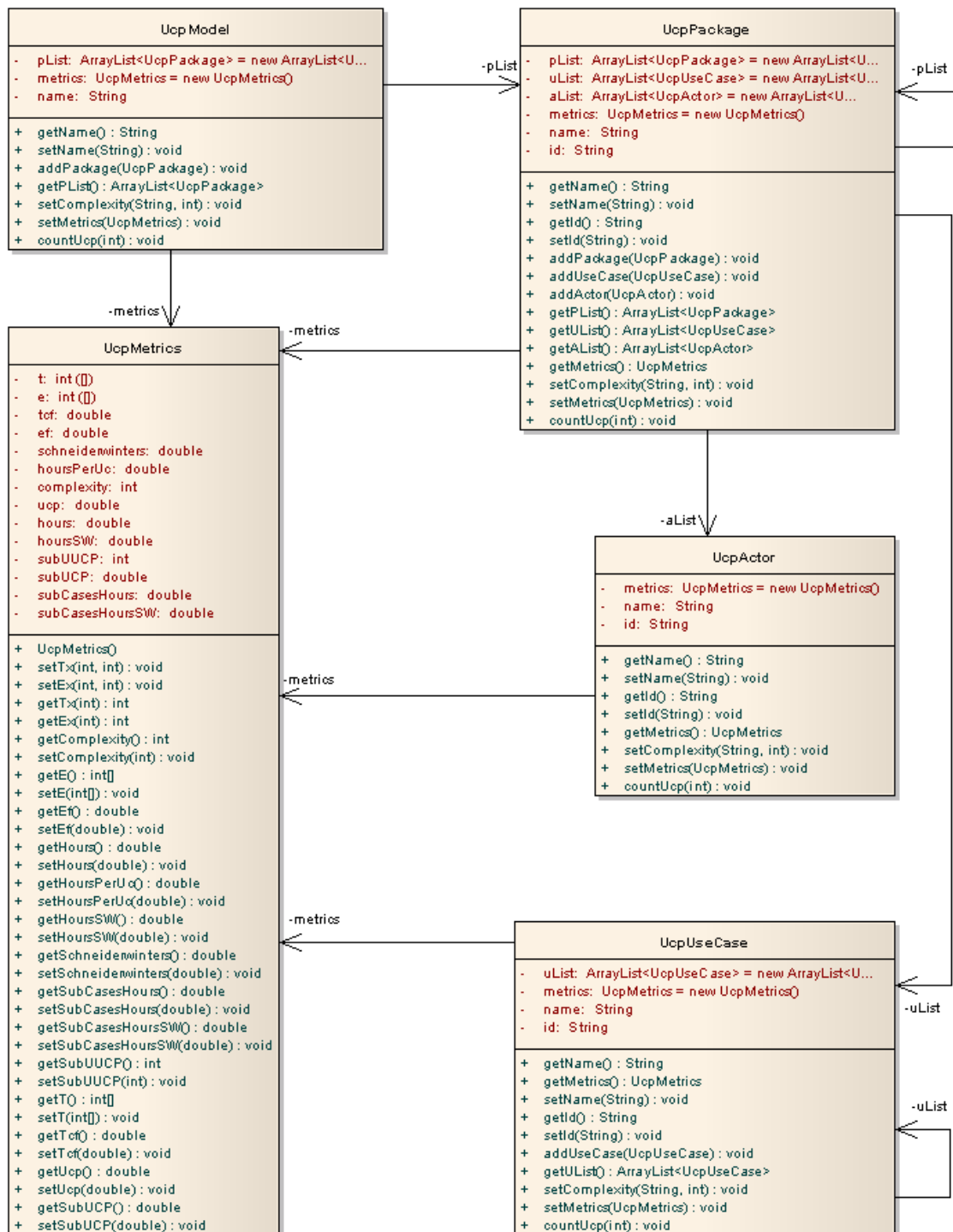
Zatímco všechny uvedené atributy se vztahovaly vždy k danému elementu UCP modelu, ostatní atributy s prefixem *sub* se vztahují k elementům vnořeným. Každý element modelu tedy ve své tabulce metrik uchovává své vlastní výsledky i výsledky svých vnořených elementů. Pro vnořené elementy se však do tabulky ukládají pouze sumární výsledky. Tyto sumární výsledky lze z určitého pohledu brát jako redundantní data, která je vždy možno vypočítat z vnořených elementů, avšak kvůli rychlosti dalšího zpracování výsledků je účelné udržovat si v tabulce tato data předpočítaná.

Třída *UcpModel* tvoří základ modelu UCP. Obsahuje seznam objektů typu *UcpPackage* a tabulku metrik, která nese výsledky za celý projekt.

Třída *UcpPackage* reprezentuje balík z modelu případů užití. Obsahuje tabulku metrik, která po provedení výpočtů nese výsledky za všechny vnořené elementy daného balíku. Třída *UcpPackage* obsahuje seznam dalších balíků (objektů typu *UcpPackage*). Vedle balíků ještě může obsahovat seznam případů užití (objektů typu *UcpUseCase*) a seznam aktérů (objektů typu *UcpActor*).

Třída *UcpUseCase* reprezentuje případ užití z modelu případů užití. Obsahuje seznam dalších vnořených případů užití (objektů typu *UcpUseCase*). Také tato třída obsahuje tabulku metrik, která v sobě nese výsledky za daný případ užití a výsledky za ostatní vnořené elementy.

Třída *UcpActor* reprezentuje aktéry z modelu případů užití. Neobsahuje žádné další vnořené elementy. Tabulka metrik u objektu typu *UcpActor* nese pouze hodnoty za daného aktéra.



Obr. 14: Třídy reprezentující UCP model

Rozložení jednotlivých softwarových komponent na hardwarových zdrojích a jejich spolupráci zobrazuje diagram nasazení na obrázku Obr. 15.

Na klientském zařízení (server nebo osobní počítač) je nasazena aplikace Blue Camp nebo libovolný jiný informační systém pro řízení projektů. Tento systém komunikuje s modulem Use Case Plan pomocí webové služby. Klient webové služby je součástí systému pro řízení projektů. Vlastní webová služba je pak součástí modulu Use Case Plan. Modul Use Case Plan může být nasazen na jiném fyzickém (případně virtuálním) serveru, než na kterém je nasazen klientský systém pro řízení projektů.

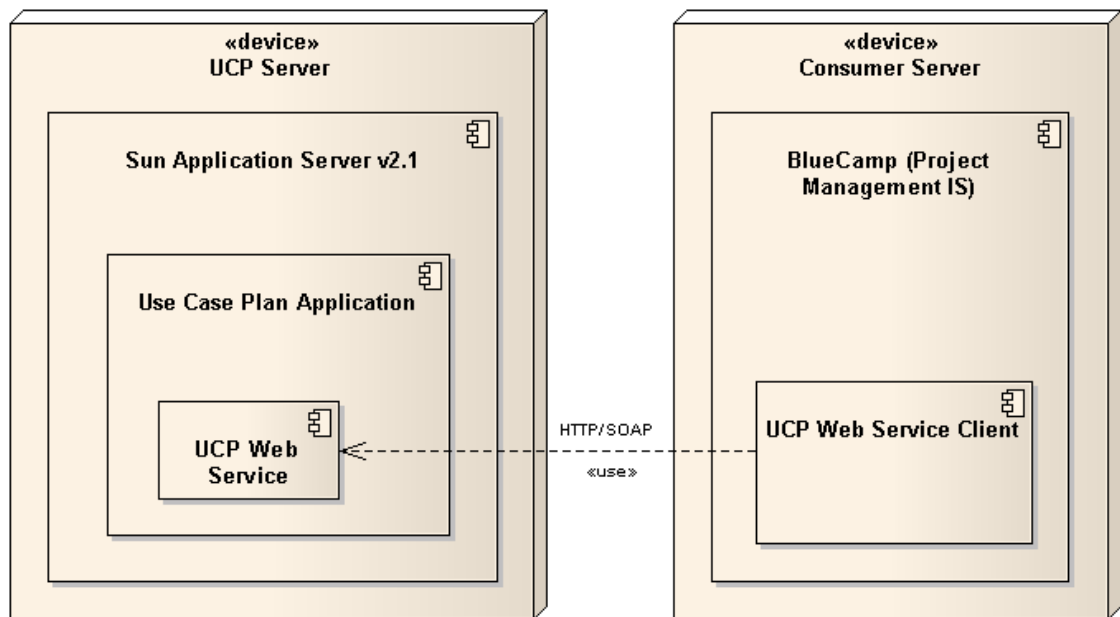
Důvodem pro realizaci webové služby bylo zpřístupnění modulu Use Case Plan libovolným systémům pro řízení projektů. Pod pojmem webové služby (*Web Service*) rozumíme systém navržený pro podporu přenosu informací mezi počítači různých platforem. Při komunikaci pomocí webových služeb hraje jeden z počítačů roli poskytovatele webové služby a druhý počítač je klientem webové služby.

Pro realizaci webové služby modulu UCP byla vybrána webová služba založená na protokolu SOAP. Přenášené SOAP zprávy mají formát XML. Formát XML má výhodu v tom, že se předávaná data nemusí omezovat na text, je možné předávat si složité objekty nebo kolekce objektů. Formát XML umožňuje také popsat strukturu a typy předávaných dat.

Abychom mohli webovou službu používat, musíme znát, jaké funkce nám webová služba poskytuje, jaké mají tyto funkce parametry a jaké hodnoty vracejí. K tomuto účelu slouží WSDL (*Web Service Definition Language*) dokument, který definuje služby jako kolekci koncových bodů zpracovávajících zprávy. Popis realizované webové služby (soubor WSDL) je k dispozici v příloze č. 4.

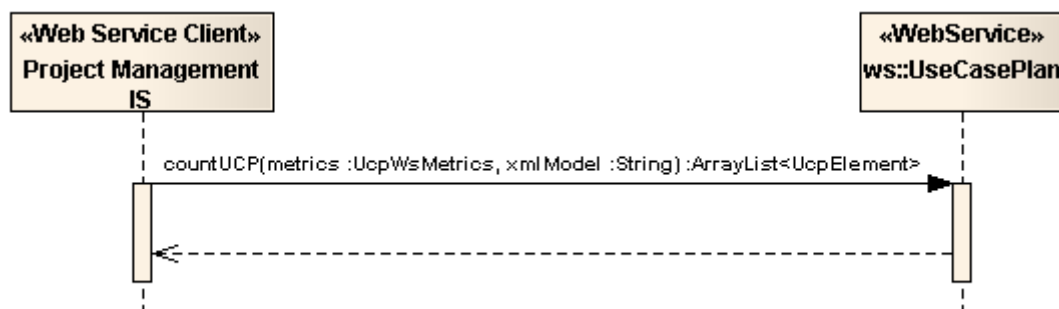
Jako transportní protokol je použit protokol HTTP. Protokol HTTP je prakticky základem dnešní internetové infrastruktury a není blokován firewally, což je jedna z hlavních výhod oproti jiným protokolům.

Modul UCP byl realizován ve vývojovém prostředí *NetBeans 6.8* v programovacím jazyce Java. Webová služba je založena na technologii JAX-WS. Aplikace byla nasazena na aplikačním serveru *Sun Application Server v2.1* a vyzkoušena byla také na aplikačním serveru *GlassFish v3*.



Obr. 15: Diagram nasazení jednotlivých komponent systému

Komunikace mezi klientem webové služby a webovou službou je znázorněna na sekvenčním diagramu Obr. 16. Klient webové služby volá operaci *countUCP*, která v parametru předává nastavené metriky *metrics:UcpMetrics* a exportovaný model analýzy realizovaného softwaru *xmlModel:String*. Jako návratovou hodnotu obdrží klientská aplikace seznam objektů typu *UcpElement*. Tyto objekty obsahují výsledky výpočtu za jednotlivé elementy UCP modelu.



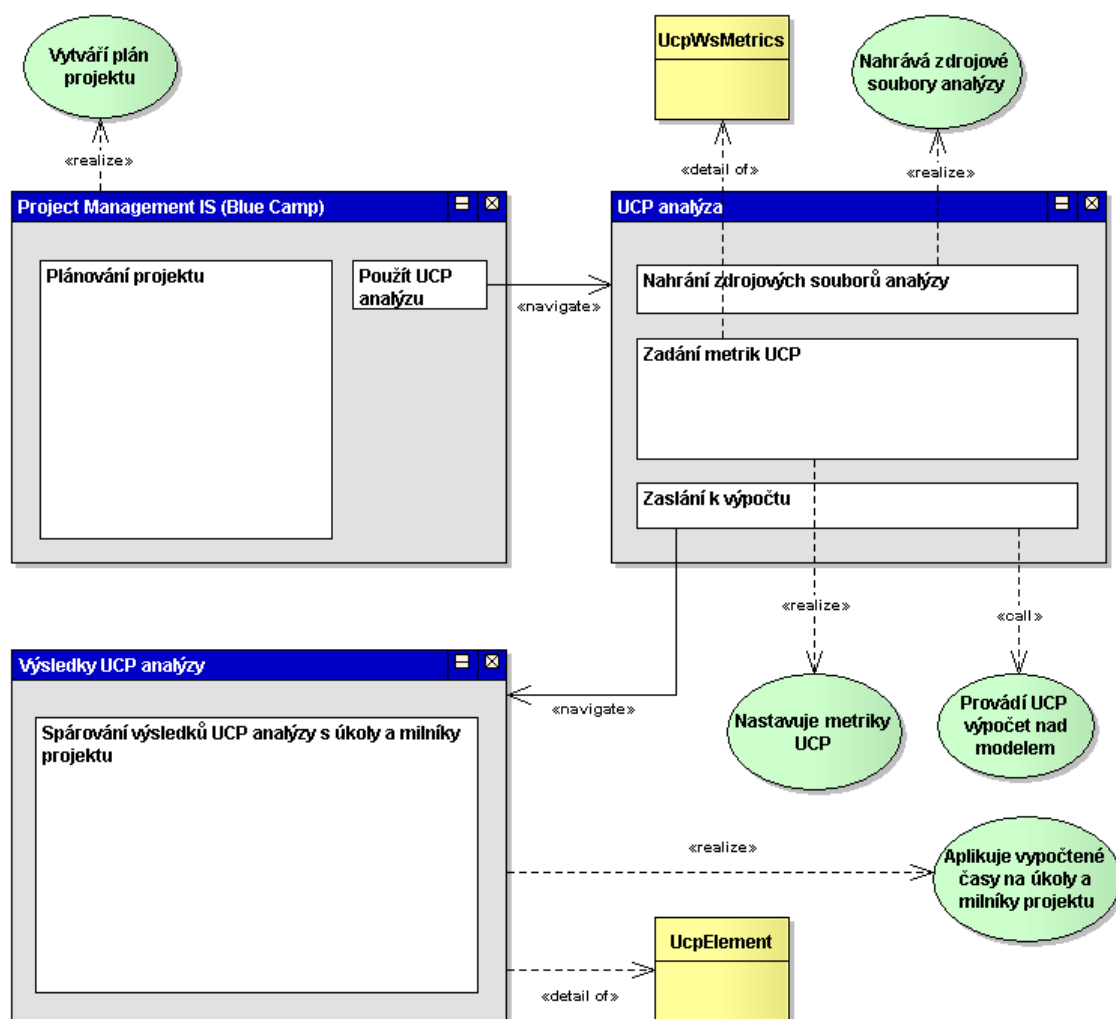
Obr. 16: Komunikace mezi klientem webové služby a webovou službou

Rozšířený diagram uživatelského rozhraní (Obr. 17) zobrazuje základní fragmenty budoucího rozhraní uživatele společně se související funkcionalitou a datovým pozadím. Diagram znázorňuje navrženou podobu rozhraní klientské části aplikace. Jde

tedy o formuláře, které by se měly zobrazovat uživateli informačního systému pro správu projektů.

Formulář nazvaný „*Project Management IS (Blue Camp)*“ obsahuje výpisy a ovládací prvky, které slouží k sestavení plánu projektu. Podoba tohoto formuláře je dána použitým systémem pro správu projektů. Jediným nově navrženým prvkem je tlačítko „*Použít UCP analýzu*“. Pokud bude chtít projektový manažer aplikovat na plán projektu odhady pracnosti založené na metodě UCP, zvolí toto tlačítko. Kliknutím na tlačítko se otevře nový formulář „*UCP analýza*“. Na tomto formuláři jsou obsaženy ovládací prvky, pomocí kterých může uživatel nahrát do aplikace zdrojový soubor analýzy a nastavit hodnoty pro všechny požadované metriky projektu. Zadání faktoru produktivity je řešeno vložení číselné hodnoty představující počet člověkohodin potřebných k realizaci jednoho bodu UCP. Tuto hodnotu určí projektový manažer. Pokud by se v rámci rozšíření funkcionality realizoval výpočet faktoru produktivity z báze historických dat, pak by na tomto formuláři místo klasického pole pro vložení číselné hodnoty byly ovládací prvky pro výběr podobných dokončených projektů, na základě kterých by se následně provedlo určení faktoru pracnosti.

Jakmile jsou metriky vyplněny a je nahrán soubor analýzy, může být proveden výpočet pracnosti. Aplikace zašle vstupní data do modulu Use Case Plan, který provede výpočet pracnosti a vrátí výsledná data. Výsledky jsou pak zobrazeny v tabulce na formuláři „*Výsledky UCP analýzy*“. Výpis výsledků obsahuje hodnoty pracnosti za všechny uvedené elementy modelu. Balíky případů užití a kompozitní případy užití obsahují i agregovaná data z vnořených elementů. Projektový manažer může aplikovat vypočtené hodnoty na tvorbu úkolů a milníků projektu.



Obr. 17: Rozšířený diagram uživatelského rozhraní

Na obrázku Obr. 18 je uveden snímek obrazovky testovací aplikace, která byla vytvořena na základě výše uvedeného návrhu. Aplikace slouží k nahrání zdrojového souboru analýzy, zadání požadovaných metrik a zaslání vstupních dat k výpočtu. Po výpočtu se výsledky zobrazí na záložce „Results“ (viz obrázek Obr. 19).

Vstupní soubor analýzy musí být ve formátu XML 2.1 a použitá verze UML jazyka musí být 2.0 nebo 2.1. Modul byl testován na exporty z modelovacího prostředí Enterprise Architect verze 7.0 a 7.5.

1. Load project source file (Enterprise Architect - UML 2.1 XMI 2.1):

Load file D:\DATA\export.xml

2. Set Technical factors, Enviromental factors and Hours per UCP:

Distributed system	2	Easy to change	2	Familiar with process	3	Hours per UCP	15
Performance objectives	1	Concurrent use	4	Application experience	4		
End-user efficiency	4	Security	3	Object-oriented experience	4		
Complex processing	1	Access for third parties	2	Lead analyst capability	5		
Reusable code	2	Training needs	1	Motivation	2		
Easy to install	5			Stable requirements	4		
Easy to use	5			Part-time staff	0		
Portable	0			Difficult prog. language	3		

3. Count UCP model:

Count model

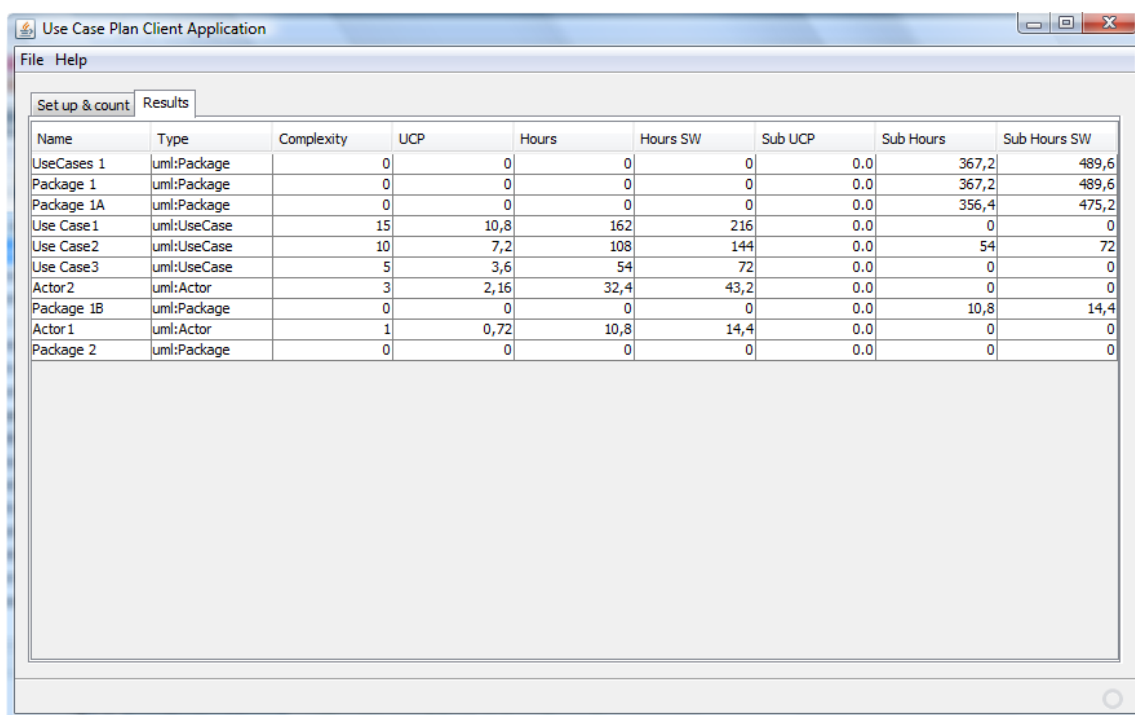
Obr. 18: Náhled obrazovky testovací aplikace

Tabulka výsledků obsahuje v řádcích jednotlivé elementy a ve sloupcích vypočtené hodnoty:

- Sloupec „Name“ obsahuje názvy elementů.
- Sloupec „Type“ obsahuje typy elementů UCP modelu, které odpovídají typům elementů z jazyka UML.
- Sloupec „Complexity“ obsahuje komplexnost elementů. U elementů typu balík „uml:Package“ se komplexnost neuvádí.
- Sloupec „UCP“ obsahuje počet bodů UCP pro daný element.
- Sloupec „Hours“ obsahuje počet člověkohodin potřebných k realizaci daného elementu.
- Sloupec „Hours SW“ obsahuje počet člověkohodin vypočtených s použitím algoritmu Schneider Winters.
- Sloupec „Sub UCP“ obsahuje počet UCP bodů vnořených elementů. Do výsledku jsou zahrnuty pouze vnořené elementy, pro které ještě nebyla vypočtena pracnost v člověkohodinách.

- Sloupec „*Sub Hours*“ obsahuje počet člověkohodin potřebných k realizaci vnořených elementů.
- Sloupec „*Sub Hours SW*“ obsahuje počet člověkohodin vypočtených s použitím algoritmu Schneider Winters. Vztahuje se k vnořeným elementům.

Výsledné hodnoty se mohou lišit dle nastavené hloubky výpočtu. Hloubka výpočtu definuje, u kterých elementů se provede plný výpočet pracnosti a které elementy budou postupovat svá data k výpočtům do nadřazených úrovní. Proto mohou být některá pole v tabulce výsledků bez vyplněných hodnot, zatímco při jiné nastavené hloubce by tato pole výsledné hodnoty obsahovala.



Name	Type	Complexity	UCP	Hours	Hours SW	Sub UCP	Sub Hours	Sub Hours SW
UseCases 1	uml:Package	0	0	0	0	0.0	367,2	489,6
Package 1	uml:Package	0	0	0	0	0.0	367,2	489,6
Package 1A	uml:Package	0	0	0	0	0.0	356,4	475,2
Use Case1	uml:UseCase	15	10,8	162	216	0.0	0	0
Use Case2	uml:UseCase	10	7,2	108	144	0.0	54	72
Use Case3	uml:UseCase	5	3,6	54	72	0.0	0	0
Actor2	uml:Actor	3	2,16	32,4	43,2	0.0	0	0
Package 1B	uml:Package	0	0	0	0	0.0	10,8	14,4
Actor1	uml:Actor	1	0,72	10,8	14,4	0.0	0	0
Package 2	uml:Package	0	0	0	0	0.0	0	0

Obr. 19: Náhled obrazovky testovací aplikace s výsledky

6.3 Praktické použití modulu

Metoda UCP byla vyzkoušena na reálném softwarovém produktu. Byl vybrán již zrealizovaný projekt s dobře provedenou analýzou (zejména z hlediska popisu případů užití).

Analýza zkoumaného systému byla vyexportována z modelovacího nástroje Enterprise Architect 7.5. Následně byla nahrána do testovací aplikace modulu Use Case Plan. V testovací aplikaci byly nastaveny technické faktory a faktory prostředí. Faktory byly definovány dle posudků analytika a architekta projektu.

Faktor produktivity byl určen s použitím starších projektů, u kterých byla k dispozici analýza s popisem případů užití a celková doba realizace. Výpočet UCP bodů pro tyto projekty byl proveden opět s pomocí modulu Use Case Plan. Po výpočtu bodů UCP byla určena pracnost dle vztahu:

$$PF = \frac{\textit{Aktuální pracnost}}{UCP}$$

kde

PF je faktor produktivity,

Aktuální pracnost je reálná dosažená pracnost na předchozích projektech,

UCP je počet bodů UCP z předchozích projektů.

Dosažená pracnost byla určena z výkazů práce realizačního týmu. Přesnost určení faktoru produktivity na základě starých projektů mohla být ohrožena z důvodu chybějící definice komplexnosti u případů užití a aktérů ve starých analýzách. Analytici nebyli u dřívějších projektů seznámeni s metodikou UCP, takže hodnotu komplexnosti u případů užití nezadávali. Shodou okolností však vytvářeli většinou případy užití pouze s jednou či dvěma transakcemi, takže se komplexností řadily do první třídy. To stejné platilo i pro aktéry. Díky tomu bylo možné hotové analýzy použít pro výpočet bodů UCP a následné určení faktorů pracnosti.

Jakmile byly k dispozici technické faktory, faktory okolí a faktor produktivity realizačního týmu, byl v modulu Use Case Plan proveden výpočet pracnosti zkoumaného projektu.

Výsledná pracnost byla srovnána s reálnými hodnotami, kterých bylo při realizaci projektu dosaženo. Dále byly výsledky výpočtu porovnány také s odhadem pracnosti, který provedl projektový manažer před zahájením realizace projektu.

Průběh výpočtu je uveden v následujícím textu.

Byl zjištěn rozsah tří projektů a celková dosažená pracnost:

- Celkový počet bodů UCP = 1589,3 *bodů*
- Celková aktuální pracnost = 3337,53 *člověkodnů* = 26700,24 *člověkohodin*

Dále byl vypočítán faktor produktivity:

- $PF = \frac{\text{Aktuální pracnost}}{UCP} = \frac{26700,24 \text{ člověkohodin}}{1589,3 \text{ bodů}} = 16,8 \text{ čh/UCP}$
- Technické faktory a faktory prostředí jsou uvedeny v tabulce Tab. 8.
- Faktor pracnosti vypočítaný dle algoritmu Schneider Winters = 20 *člověkohodin/UCP*

Po získání faktorů bylo možné provést výpočet nad zkoumaným projektem:

- Velikost systému, počet UCP bodů = 681,12 *bodů*
- Pracnost zkoumaného systému vypočítaná modulem Use Case Plan = 11 442,82 *člověkohodin*
- Pracnost zkoumaného systému s použitím metody Schneider Winters = 13 622,4 *člověkohodin*
- Reálná dosažená pracnost zkoumaného projektu = 1 322 *člověkodnů* = 10 576 *člověkohodin*
- Odhadnutá pracnost dle expertního odhadu = 1 130 *člověkodnů* = 9 040 *člověkohodin*

Po výpočtu bylo provedeno srovnání výsledků:

- Rozdíl mezi dosaženou pracností a odhadem pracnosti pomocí metody UCP a faktoru pracnosti získaného z historických dat = 866,82 *člověkohodin*

- Rozdíl mezi dosaženou pracností a odhadem pracnosti pomocí metody UCP a faktoru pracnosti vypočítaného dle metody Schneider Winters = 3 046,4 *člověkohodin*
- Rozdíl mezi dosaženou pracností a expertním odhadem pracnosti = 1 536 *člověkohodin*

V procentním vyjádření vyšly odhady následovně:

- Rozdíl mezi dosaženou pracností a odhadem pracnosti pomocí metody UCP a faktoru pracnosti získaného z historických dat = 8,2 %
- Rozdíl mezi dosaženou pracností a odhadem pracnosti pomocí metody UCP a faktoru pracnosti vypočítaného dle metody Schneider Winters = 28,8 %
- Rozdíl mezi dosaženou pracností a expertním odhadem pracnosti = 14,5 %

Zatímco u výsledků získaných metodou UCP došlo k nadhodnocení očekávané pracnosti, u expertního odhadu byla pracnost projektu podhodnocena. Nesprávný odhad se v obou případech projeví ve zvýšených nákladech oproti situaci, kdy byl projekt naplánován realisticky. V případě nadhodnocení pracnosti jsou podle Parkinsonova zákona spotřebovány všechny dostupné zdroje. Pokud dojde k podhodnocení, pak prudce rostou náklady způsobené nerealistickým plánem (zvýšeným výskytem chyb, nedodržením termínů apod.).

Výsledek výpočtu s použitím faktoru pracnosti získaného metodou Schneider Winters se značně liší od reálně dosažené pracnosti. Takový výsledek je ovšem možné očekávat, pokud víme, jak uvedená metoda funguje. Oborem hodnot použitého algoritmu je tříprvková množina faktorů pracnosti. Výsledek tedy nabývá jedné z těchto hodnot: 20, 28 a 36 *člověkohodin / UCP*. Pro přesnější odhady by bylo potřeba algoritmus modifikovat tak, aby nabízel širší škálu možných výsledků. Metoda Schneider Winters však poskytuje možnost provést alespoň orientační odhad pracnosti v případě, kdy nejsou k dispozici historická data a není možné zjistit faktor produktivity daného týmu.

Výsledek odhadu pracnosti provedený expertem může značně kolísat v závislosti na zkušenostech konkrétního experta. Čím více expertů provádí odhad, tím jsou výsledky odhadu objektivnější. Pro zkoumaný projekt však nebyli další experti k dispozici.

Nejpřesnějšího odhadu dosáhl výpočet založený na metodě UCP s použitím faktoru produktivity získaného na základě historických dat. Historická data pochází z projektů, které se svým rozsahem, složitostí a použitými technologiemi nejvíce podobaly zkoumanému projektu. Realizační tým byl až na drobné obměny u všech projektů stejný. Zadané faktory tedy nejlépe odpovídaly realitě a proto i výsledek získaný touto metodou byl nejpřesnější.

Pokud by tedy na začátku projektu byl použit odhad pomocí metody UCP místo odhadu provedeného expertem, byl by odklon od reality o 6,3 % nižší.

Dalšího zpřesnění výsledků by šlo docílit zavedením metodiky UCP do procesu tvorby analýzy a úpravou báze historických dat. Dodržení metodiky v procesu tvorby analýzy znamená především přesně definovat komplexnost případů užití a aktérů. U případů užití jde o dodržení jednotné úrovně abstrakce a úrovně podrobnosti popisu scénářů. Důležité je správné určení počtu transakcí a následné správné odvození komplexnosti případů užití.

Báze historických dat vznikne z údajů o dokončených a změřených projektech. Tuto bázi lze rozdělit dle charakteristik jednotlivých projektů do skupin. Kritériem pro tvorbu skupiny by mohl být rozsah projektu, složení týmu, technologické faktory nebo faktory okolí. Nově realizovaný projekt se srovná s dokončenými projekty a vybere se na základě podobnosti nejvhodnější skupina. Údaje o dokončených projektech z této skupiny se pak použijí k výpočtu faktoru produktivity. Výsledný faktor produktivity je následně aplikován v algoritmu výpočtu pracnosti nově realizovaného projektu. Faktor produktivity vypočítaný na základě dat z podobných dokončených projektů lépe odpovídá nově realizovanému projektu, což by mělo mít pozitivní vliv na přesnost konečného odhadu pracnosti. Tento postup představuje rozšíření základní metody UCP o techniku analogie, která staví právě na identifikaci rozdílů a podobností u srovnatelných projektů.

ZÁVĚR

Společnost Blue Pixel pocítuje dopady předchozího rychlého růstu. Zejména komunikační procesy a procesy řízení projektů vyžadují provedení změn. Společnosti působící v oblasti IT se musí rychle vyvíjet a neustále zlepšovat své pracovní procesy. Tohoto rysu firma Blue Pixel dosahuje, ale na druhou stranu některé změny (např. rychlý růst firmy) jsou zvládány s jistými obtížemi. Zaměstnanci firmy jsou vesměs vysoce kvalifikovaní a motivovaní, ale jejich motivace spočívá hlavně v osobním profesním růstu, což se projevuje i ve vztahu nižší loajality k firmě a vyvolává vyšší fluktuaci.

Po shrnutí cílů a strategie firmy byly analyzovány pracovní a řídicí procesy probíhající při vývoji softwarových produktů. Na základě analýzy byl vytvořen procesní model. Zatímco běžné pracovní procesy probíhají optimálně s odpovídajícím nasazením softwarových nástrojů, v oblasti řízení a komunikace existují zjevné nedostatky ve zvolených nástrojích a ve způsobu práce s těmito nástroji. Daná oblast vyžadovala reinženýring postižených procesů a navržení nové skladby souvisejících softwarových nástrojů. Vedle analýzy procesů byla provedena také analýza McFarlan, která rozdělila používané softwarové nástroje do skupin dle míry jejich důležitosti pro chod firmy.

Nejvýznamnější úpravou, která vyplynula na základě reinženýringu procesů, byl návrh nasazení systému pro správu a řízení projektů. Procesy řízení a komunikace ve společnosti Blue Pixel vyžadovaly ke svému fungování velké množství softwarových nástrojů. Používání tak velkého množství aplikací k v zásadě příbuzným úkonům snižuje produktivitu práce. Nový systém by měl eliminovat množství používaných aplikací a docílit vyšší produktivity.

Společnost Blue Pixel už vlastní jeden informační systém, který je částečně používán pro řízení a správu projektů. Nese název Blue Camp a jedná se o vlastní interní produkt firmy. Produkt však nebyl v takovém stavu, aby mohl plně pokrývat všechny požadavky na systém správy a řízení projektů definované dle nového procesního schématu.

Vyváženost systému byla posouzena v rámci analýzy HOS 8. Zkoumaný informační systém byl ohodnocen jako důležitý pro chod firmy. Jeho výpadek by fungování firmy výrazně ovlivnil. V systému jsou totiž zaneseny všechny alokace, plány projektů, reporty a další řídicí informace. Bez těchto údajů by bylo ohroženo řízení projektů a v důsledku toho i samotný chod prací na projektech. Souhrnný stav informačního systému je nižší než doporučený stav odvozený od významu tohoto informačního systému. Proto bylo třeba souhrnný stav zvýšit. Pro zvýšení souhrnného stavu vyváženého systému je ale zapotřebí zvýšit stav většiny oblastí. Jako vhodná se tedy jevila strategie expanze.

V rámci práce byl také zpracován přehled konkurenčních systémů pro správu a řízení projektů. Stanoveným požadavkům firmy, ať už se jednalo o požadavky definované managementem firmy nebo o požadavky vyplývající z procesní analýzy, však nejlépe vyhovoval zmiňovaný systém Blue Camp. Bylo ale zapotřebí ještě navrhnout další funkcionality, aby mohl systém všem zadaným požadavkům plně vyhovět.

Inspirací pro nové funkce systému byly vlastnosti konkurenčních aplikací na trhu. Návrh funkcí také vycházel z vlastností kancelářských nástrojů, které byly ve firmě doposud používány k podpoře daných procesů. Zredukovány však byly funkce, které jsou v kancelářských nástrojích navíc a které nebyly k podpoře těchto procesů využívány. Odbourání nepotřebných funkcí by mělo vést ke zvýšení přehlednosti a ovladatelnosti aplikace. Uživatelé budou mít v aplikaci pouze a jen tu funkcionality, kterou přímo využijí pro činnosti prováděné v rámci daných procesů.

Navržené funkce byly podrobeny hodnotové analýze, pomocí které byla určena hodnota jednotlivých funkcí ve vztahu k uživateli. Podle pořadí funkce v posloupnosti dané hodnotou či užitečností bylo možné naplánovat, které funkce implementovat přednostně a které je případně možné odložit na pozdější realizaci nebo nerealizovat vůbec.

Existují hrozby, které mohou mít nežádoucí vliv na nasazení a chod navrženého systému. Působením hrozby může vzniknout škoda. Bylo tedy potřeba identifikovat potencionální hrozby, určit míru rizika a navrhnout nutná protipatření. Pro vyjádření

rizik a určení jejich závažnosti byla použita kvalitativní metoda analýzy rizik. Pro úspěšné nasazení bylo ještě zapotřebí naplánovat harmonogram realizace a nasazení systému, identifikovat zdroje a naplánovat zaškolení uživatelů.

Jedna z navržených funkcionalit systému byla v rámci této práce zrealizována. Vybrán byl modul pro odhad pracnosti softwarových projektů. Pro výběr tohoto modulu jsem se rozhodl ze dvou důvodů. Prvním důvodem byla relativní nezávislost modulu na systému Blue Camp. Modul lze provozovat samostatně a pomocí webové služby jej mohou využívat libovolné externí systémy. Druhým důvodem pro výběr tohoto modulu bylo jeho teoretické pozadí. Modul je postaven na metodě Use Case Points, která umožňuje provádět odhady pracnosti softwarových projektů na základě případů užití vytvořených v rámci analýzy.

Metoda Use Case Points byla popsána v teoretické části práce. V praktické části práce byla metoda aplikována na ukázkovém výpočtu pracnosti jednoho případu užití. Ukázkové zpracování případu užití prochází krok za krokem algoritmus výpočtu a podrobně čtenáře seznamuje s principy metody.

Dále byl realizovaný modul popsán po technické stránce. Samotné realizaci modulu předcházela podrobný návrh, který byl zachycen pomocí modelů UML. Vytvořené UML diagramy byly popsány a vysvětlen byl také význam jednotlivých částí systému. Samotný UCP modul je přístupný pouze přes webovou službu, proto byla v rámci diplomové práce vytvořena také testovací aplikace (klient webové služby), která obsahuje grafické uživatelské rozhraní.

Nasazený modul UCP byl vyzkoušen na reálném softwarovém projektu. Pro testování byl vybrán již zrealizovaný projekt s dobře provedenou analýzou, u kterého byla známa doba realizace. Výsledná pracnost tedy mohla být srovnána s reálnými hodnotami, kterých bylo při realizaci projektu dosaženo. Dále byly výsledky výpočtu porovnány také s odhadem pracnosti, který provedl projektový manažer před zahájením realizace projektu.

Zatímco u výsledků získaných metodou UCP došlo k nadhodnocení očekávané pracnosti, u expertního odhadu byla pracnost projektu podhodnocena. Nesprávný odhad

se však v obou případech projeví ve zvýšených nákladech oproti situaci, kdy je projekt naplánován realisticky.

Významný vliv na výsledek výpočtu má tzv. faktor produktivity, který je závislý na vývojovém týmu a na složitosti zpracovávaného softwarového produktu. Faktor produktivity lze určit buď na základě expertního odhadu, z báze historických dat nebo výpočtem dle algoritmu Schneider Winters, který vychází z technických faktorů a faktorů okolí.

Výsledek výpočtu s použitím faktoru pracnosti získaného metodou Schneider Winters se značně lišil od reálně dosažené pracnosti. Takový výsledek bylo ovšem možné očekávat, pokud víme, jak uvedená metoda funguje. Metoda Schneider Winters však poskytuje možnost provést alespoň orientační odhad pracnosti v případě, kdy nejsou k dispozici historická data a není možné zjistit faktor produktivity daného týmu.

Nejpřesnějšího odhadu dosáhl výpočet založený na metodě UCP s použitím faktoru produktivity získaného na základě historických dat. Historická data pocházela z projektů, které se svým rozsahem, složitostí a použitými technologiemi nejvíce podobaly zkoumanému projektu.

Pokud by tedy na začátku projektu byl použit odhad pomocí metody UCP místo odhadu provedeného expertem, byl by odklon od reality o 6,3 % nižší. Dalšího zpřesnění výsledků by šlo docílit zavedením metodiky UCP do procesu tvorby analýzy a lepším využitím báze historických dat.

Zde diplomová práce poskytuje prostor pro další možné rozšíření. Modul by mohl využívat techniky umělé inteligence k nalezení podobných projektů v bázi historických dat. V úvahu by přicházela například fuzzy logika, pomocí které by bylo možné identifikovat projekty, které svojí složitostí, rozsahem a složením řešitelského týmu nejlépe odpovídají nově realizovanému projektu. Faktor produktivity vypočítaný na základě dat z nalezených historických projektů by lépe odpovídal novému projektu, což by mělo pozitivní vliv na přesnost konečného odhadu pracnosti.

Modul UCP je připraven i na další uvažovaná rozšíření. Zatímco v aktuální verzi modulu je možné nastavit faktory výpočtu pouze pro celý analytický model,

v rozšířeném modulu bude možné definovat vlastní faktory i pro jednotlivé subelementy modelu. Možnost nastavení různých metrik pro libovolné elementy modelu dovolí lépe ohodnotit charakter odlišných částí systému.

Diplomová práce dosáhla vytyčených cílů. Procesní analýza odhalila slabá místa v oblasti softwarové podpory některých firemních procesů. Jednalo se o procesy správy a řízení projektů. Navrženy byly úpravy, které vedou k zefektivnění daných procesů. V rámci navržených úprav byly definovány požadavky na nový informační systém pro správu projektů. Vybraná část navrženého informačního systému byla zrealizována. K realizaci byl vybrán modul pro výpočet pracnosti softwarových projektů, který staví na hlubším teoretickém základě – na metodě Use Case Points. Teoretickým přínosem práce je podrobný popis této metody a ukázkové výpočty. Praktickým přínosem práce je úprava procesního modelu, navržení úprav informačního systému Blue Camp a kompletní implementace metody Use Case Points v rámci realizace vybraného modulu. Modul umožňuje firmě Blue Pixel provádět odhady pracnosti softwarových projektů již v úvodních fázích jejich realizace. Na rozdíl od běžně dostupných „UCP kalkulačků“ provádí modul výpočty přímo nad modelem případů užití a vypočítané výsledky poskytuje informačním systémům, které na základě nich mohou vytvářet harmonogramy, milníky a úkoly projektů. Modul lze díky zvolené architektuře a podrobné dokumentaci používat z libovolných externích systémů. Zrealizovaný UCP modul a navržené úpravy dávají systému Blue Camp potenciál k optimální podpoře procesů spadajících do oblasti řízení a správy projektů.

LITERATURA

- 1) ABERDEEN GROUP. *Software as a Service Buyer's Guide* [online]. [cit. 2009-08-20]. Dostupné z <http://www.aberdeen.com/Aberdeen-Library/3305/RA_SaaSBuyerGuide_BE_3305.aspx>.
- 2) ANDA, B. *Comparing Effort Estimates Based on Use Case Points with Expert Estimates* [online]. Department of Informatics, University of Oslo [cit. 2009-12-10]. Dostupné z <http://www.bfpug.com.br/Artigos/UCP/Anda-Comparing_Effort_Estimates_Based_on_UCP.pdf>.
- 3) BANERJEE, G. *Use Case Points – An Estimation Approach* [online]. [cit. 2010-03-20]. Dostupné z <http://www2.fiit.stuba.sk/~bielik/courses/msi-slov/reporty/use_case_points.pdf>.
- 4) BARTES, F. *Hodnotový management*. 1. vydání Brno: Fakulta podnikatelská, Vysoké učení technické v Brně, Akademické nakladatelství CERM, s.r.o., 2007. 77 s. ISBN 978-80-214-3531-5.
- 5) BARTES, F. *Inovace v podniku*. 1. vydání Brno: Fakulta podnikatelská, Vysoké učení technické v Brně, Akademické nakladatelství CERM, s.r.o., 2005. 133 s. ISBN 80-214-3086-9.
- 6) CARROLL, E. *Estimating Software Via Use Cases* [online]. Agilis Solutions, Beaverton [cit. 2010-03-09]. Dostupné z <<http://www.agilissolutions.com/WhitePapers/PDF/Estimating%20Software%20via%20Use%20Cases.pdf>>.
- 7) DRUCKER, P. F. *Inovace a podnikavost - Praxe a principy*, 1. vydání Praha: Management Press, 1993. 265 s. ISBN 80-85603-29-2.

- 8) FRANKOVÁ, E. *Manažerská psychologie I*, 1. vydání Brno: Fakulta podnikatelská, Vysoké učení technické v Brně, Akademické nakladatelství CERM, s.r.o., 2005. 53 s. ISBN 80-214-3065-6.
- 9) HORVÁTH, G. *Projektový management ve strojírenství* [online]. Katedra průmyslového inženýrství a managementu, Západočeská univerzita [cit. 2009-03-11]. Dostupné z <http://www.kpv.zcu.cz/prm/PRM_pr_07.pdf>.
- 10) HYNDRÁK, K. *Vytváříme projekty v programu Microsoft Project 2000*. Praha: Computer Press, 2000. 303 s. ISBN 80-7226-329-3.
- 11) KIMMEL, P. *UML Demystified*. McGraw-Hill Osborne Media, 2005. 235 s. ISBN 978-0072261820
- 12) KOCH M., DOVRTĚL J., HRŮZA T. *Management informačních systémů*, 2. vydání Brno: Fakulta podnikatelská, Vysoké učení technické v Brně, Akademické nakladatelství CERM, s.r.o., 2008. 193 s. ISBN 978-80-214-3735-7.
- 13) KOIRALA, S. *How to prepare software quotation* [online]. [cit. 2010-03-20]. Dostupné z <<http://k.lasphost.com/UseCasePoints/UseCasePoints.pdf>>.
- 14) LAIRD, L. M., BRENNAN, B. C. *Software Measurement and Estimation: A Practical Approach*. WileyBlackwell, 2006. 280 s. ISBN 978-0471676225.
- 15) LUKÁŠOVÁ R., NOVÝ I. a kol. *Organizační kultura. Od sdílených hodnot a cílů k vyšší výkonnosti podniku*. Praha: Grada Publishing, 2004. 174 s. ISBN 80-247-0648-2.
- 16) OCHODEK, M. *Use Case Points* [online]. Faculty of Computing Science and Management, Poznań University of Technology, 2010 [cit. 2010-02-19]. Dostupné z <<http://www.se.cs.put.poznan.pl/knowledge-base/software-engineering-blog/ucp>>.

- 17) PETERKA, J. *Fenomén ASP* [online]. eArchiv.cz, 2000 [cit. 2010-03-20].
Dostupné z <<http://www.earchiv.cz/a010s200/a010s200.php3>>.
- 18) PROBASCO, L. *Dear Dr. Use Case: What About Function Points and Use Cases?*
[online]. IBM, 2002 [cit. 2009-12-10]. Dostupné z
<<http://www.ibm.com/developerworks/rational/library/2870.html>>.
- 19) RÁČEK, J., MINISTR, J. *Odhady nákladů vývoje workflow systému* [online].
Fakulta informatiky MU v Brně, Ekonomická fakulta VŠB-TU Ostrava [cit. 2010-01-05]. Dostupné z <<http://formular-ekf.vsb.cz/formulare/F01/tsw/getfile.php?prispevekid=851>>.
- 20) RADECKÝ A., DONOGHUE A., ERBEN L. *Jak zvládnout únik informací*.
[online]. CIO Business World [cit. 2009-11-28]. Dostupné z
<<http://businessworld.cz/it-strategie/jak-zvladnout-unik-informaci-5197>>
- 21) RAIS, K. *Risk management*, 1. vydání Brno: Fakulta podnikatelská, Vysoké učení
technické v Brně, Akademické nakladatelství CERM, s.r.o., 2007. 153 s. ISBN 978-80-214-3510-0.
- 22) RIBU, K. *Estimating Object-Oriented Software Projects with Use Cases*.
(Diplomová práce) Oslo: University of Oslo, Department of Informatics, 2001.
133 s.
- 23) ROSENAU, M. D. *Řízení projektů*. Praha: Computer Press, 2000. 344 s. ISBN 80-7226-218-1.
- 24) ŘEPA, V. *Podnikové procesy - Procesní řízení a modelování*, 2. aktualizované a
rozšířené vydání Praha: Grada, 2008. 288 s. ISBN: 978-80-247-2252-8.

- 25) SCHNEIDER, G., WINTERS, J. *Applying use cases: a practical guide*. 1. vydání
Boston: Addison-Wesley Professional, 1998. 188 s. ISBN 978-0201309812.
- 26) SOCHOVÁ Z., ŠTINDLOVÁ J. *K novým ekonomickým termínům typu „člověkohodina“*. [online]. Ústav pro jazyk český AV ČR [cit. 2010-03-28].
Dostupné z <<http://nase-rec.ujc.cas.cz/archiv.php?art=4307>>
- 27) STRUSKA, Z., PERGL, R. *Srovnávání metod COCOMO a analýza Function Points* [online]. Katedra informačního inženýrství, PEF ČZU Praha [cit. 2010-03-09]. Dostupné z
<<http://www.agris.cz/etc/textforwarder.php?iType=2&iId=152840&PHPSESSID=3e>>.
- 28) TRASK SOLUTIONS. *Efektivní projektové vedení* [online]. Trask solutions s.r.o.
[cit. 2009-03-09]. Dostupné z <http://www.trask.cz/DeliverLive/Odborne_clanky-20.6.05_-_%C5%98%C3%ADzen%C3%AD_projekt%C5%AF~79~1~656>
- 29) VAHALÍK, T. *Odhadujete pracnost projektu?* [online]. Komix s.r.o, Praha [cit. 2010-03-12]. Dostupné z <<http://www.plustek.cz/upload/noviny2004c.pdf>>.

SEZNAM OBRÁZKŮ

Obr. 1: Procesy při vývoji softwarových systémů	39
Obr. 2: Procesy při vývoji softwarových systémů	41
Obr. 3: Procesy pro řízení projektu	43
Obr. 4: Upravené procesy pro řízení projektu	45
Obr. 5: Stav zkoumaných oblastí dle analýzy HOS 8	50
Obr. 6: Náhled na uživatelské rozhraní systému Blue Camp	59
Obr. 7: Harmonogram projektu v Ganttově diagramu	76
Obr. 8 Činnosti manažera v systému Blue Camp	83
Obr. 9 Schematické znázornění použití modulu Use Case Plan	84
Obr. 10 Základní schéma výpočtu bodů UCP	85
Obr. 11: Schéma výpočtu UCP modelu	87
Obr. 12: Schéma výpočtu UCP elementu	88
Obr. 13: Rozdělení tříd do balíků	89
Obr. 14: Třídy reprezentující UCP model	91
Obr. 15: Diagram nasazení jednotlivých komponent systému	93
Obr. 16: Komunikace mezi klientem webové služby a webovou službou	93
Obr. 17: Rozšířený diagram uživatelského rozhraní	95
Obr. 18: Náhled obrazovky testovací aplikace	96
Obr. 19: Náhled obrazovky testovací aplikace s výsledky	97

SEZNAM TABULEK

Tab. 1: Technické faktory a faktory prostředí	29
Tab. 2: Výsledky SWOT analýzy	36
Tab. 3: Rozdělení aplikací dle analýzy McFarlan	48
Tab. 4: Párové srovnání funkcí v hodnotové analýze	64
Tab. 5: Ohodnocení funkcí systému Blue Camp	65
Tab. 6: Seřazení funkcí podle jednotlivých kritérií	67
Tab. 7: Harmonogram projektu a alokované zdroje	76
Tab. 8: Ohodnocení technických faktorů a faktorů prostředí	80

SEZNAM PŘÍLOH

Příloha č. 1: Charakteristické rysy organizační kultury podporující kvalitu

Příloha č. 2: Konkurenční systémy pro řízení projektů a týmovou spolupráci

Příloha č. 3: Náhledy obrazovek systému Blue Camp

Příloha č. 4: Popis webové služby (WSDL soubor)

Příloha č. 5: Kontrolní otázky pro metodu HOS 8

Příloha č. 6: Obsah přiloženého CD

Příloha č. 1: Charakteristické rysy organizační kultury podporující kvalitu

Analyzováno ve společnosti Blue Pixel Solutions, a.s. Jednotlivé body jsou ohodnoceny. Bodové hodnocení se pohybuje v rozsahu 1 až 10, kde 1 znamená nejnižší úroveň a 10 nejvyšší úroveň.

Charakteristické rysy organizační kultury podporující kvalitu ve vztahu k vybraným principům TQM:

- | | |
|-----------------------------------|----|
| • Trvalé zlepšování | 9 |
| • Zvládání změn | 6 |
| • Týmová práce a vzájemný respekt | 7 |
| • Analytický přístup | 10 |

Charakteristické rysy organizační kultury podporující kvalitu ve vztahu k chování členů organizace:

- | | |
|--|----|
| • Angažovanost, aktivita a iniciativa pracovníků | 7 |
| • Samostatnost a odpovědnost pracovníků | 8 |
| • Cílovost | 7 |
| • Nalézání postupů dosahování cílů | 10 |
| • Účinné jednání ve vztazích k jiným | 6 |
| • Zajišťování zpětné vazby, sociální kontrola | 8 |
| • Specifické motivy a motivovanost pracovníka | 7 |

Zpětná vazba a systémy kontroly

- | | |
|--------------------------------|---|
| • Formální systém kontroly | 8 |
| • Neformální sociální kontrola | 6 |
| • Osobní kontrola | 8 |

Odpovědnost pracovníků

- | | |
|--|----|
| • K organizaci | 7 |
| • K profesi | 8 |
| • K společnosti jako souhrnu sociálních skupin | 6 |
| • K sobě samému (vlastní profesní růst) | 10 |

Identifikace s organizací a loajalita	8
• Identifikace pracovníka s organizací	7
• Loajalita pracovníka k organizaci	5

Charakteristické rysy obsahu kultury podporující kvalitu ve vztahu k chování managementu

• Komunikace politiky a cílů jakosti v rámci celé organizace	6
• Zavedení a aplikace principů komplexní kvality do všech subsystémů	6
• Neustálá snaha o zpětnou vazbu	8
• Vzdělávání a rozvoj pracovníků, trénink dovedností	4
• Aplikace participativního managementu, přijetí nové role managementu	6
• Zmocňování/zplnomocňování pracovníků	9
• Rozvoj týmového myšlení a týmové práce	7
• Cílené utváření organizační kultury podporující kvalitu	7

Charakteristické rysy obsahu kultury kvality na úrovni základních předpokladů a hodnot

• Kvalita klíčová pro zajištění spokojenosti zákazníků	9
• Schopnost organizace dosahovat kvality	9
• Nekvalita poškozující výkonnost organizace	5
• Kvalita „je záležitostí nás všech“	8

Čerpáno z (15)(8).

Příloha č. 2: Konkurenční systémy pro řízení projektů a týmovou spolupráci

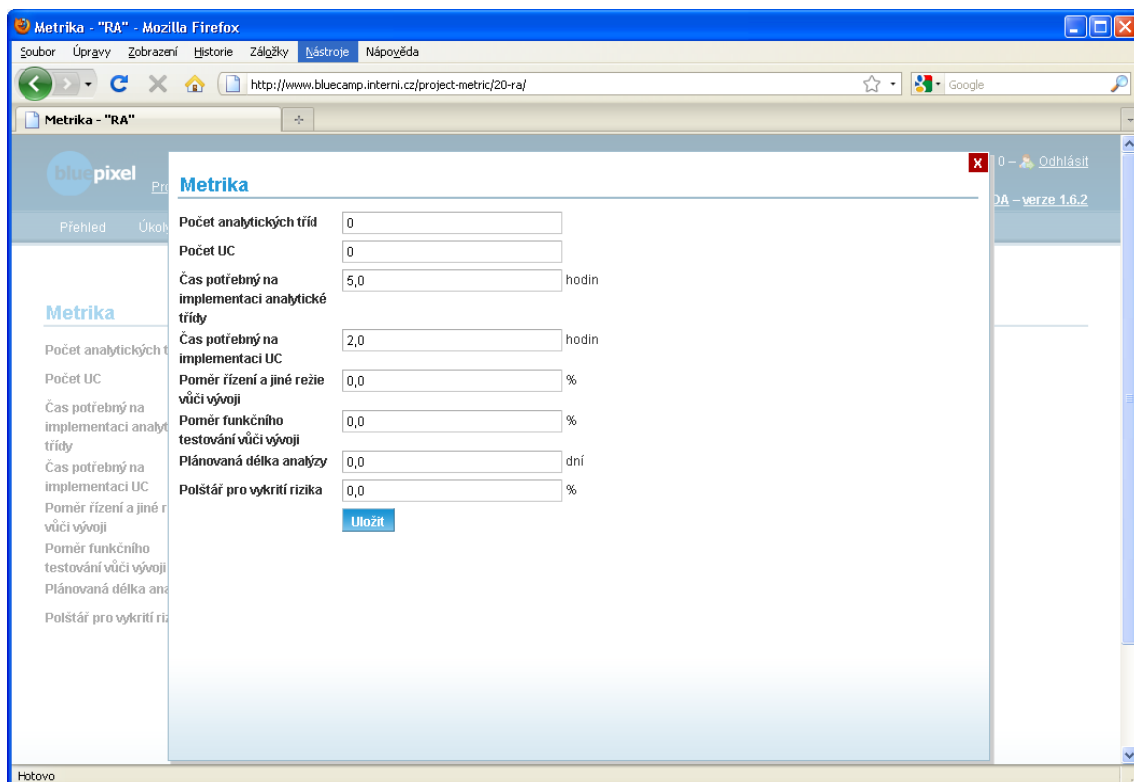
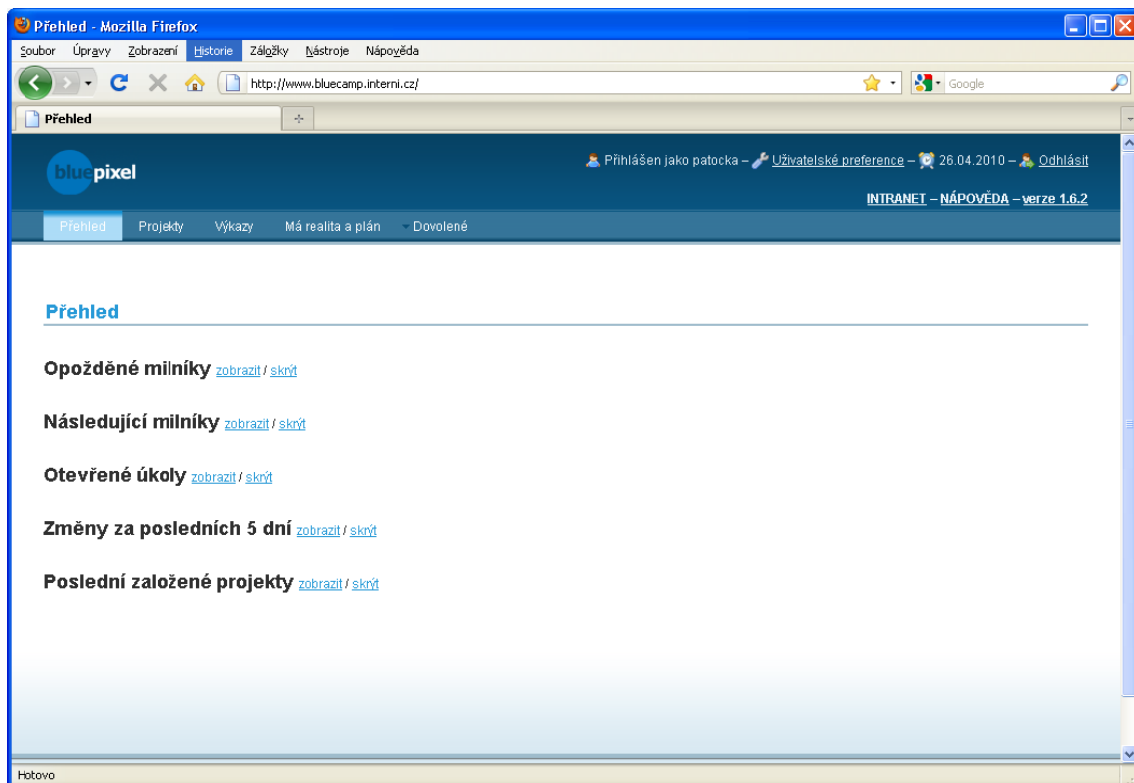
Přehled vybraných systémů pro řízení projektů a týmovou spolupráci. Přehled obsahuje jednotlivé produkty, jejich verze, výrobce, cenu, odkazy na domácí stránky a základní porovnání vlastností.

Software	Verze	Výrobce	Cena/měsíc
24SevenOffice	N/A	24SevenOffice	N/A
Aras Corp	ver 9	Aras corporation	\$0
Assembla	N/A	Assembla	\$249,00
Basecamp	ver 2.0	37 signals	\$149,00
Codendi	ver GPL v2	Xerox	\$0
Contactizer	ver 3.8.3	Objective Decision	\$119,00
Easy Projects .NET	ver 6.2	Easy Projects	\$648,00
InLoox	ver 5.6.2	Inloox	\$390,00
Planisware 5	R5 / 2008	Planisware	N/A
Project KickStart	ver 5.0	Project KickStart	\$299,00
WorkPLAN Enterprise	ver 3.0	Workplan enterprise	N/A

Software	Odkaz
24SevenOffice	http://www.24sevenoffice.com/webpage/int/index.htm
Aras Corp	http://www.aras.com/
Assembla	http://www.assembla.com/
Basecamp	http://basecamphq.com/
Codendi	http://www.codendi.com/
Contactizer	http://objective-decision.com/en/products/contactizerpro/
Easy Projects .NET	http://www.easyprompts.net/
InLoox	http://www.inloox.com/
Planisware 5	http://www.planisware.com/
Project KickStart	http://www.projectkickstart.com/
WorkPLAN Enterprise	http://www.workplan-enterprise.com/us/

Software	Týmová spolupráce	Plánování	Správa portfolia projektů	Řízení zdrojů	Web
24SevenOffice	ANO	NE	NE	NE	ANO
Aras Corp	ANO	ANO	ANO	ANO	ANO
Assembla	ANO	NE	ANO	ANO	ANO
Basecamp	ANO	NE	NE	ANO	ANO
Codendi	ANO	NE	NE	NE	ANO
Contactizer	ANO	NE	NE	ANO	NE
Easy Projects .NET	ANO	ANO	ANO	ANO	ANO
InLoox	ANO	ANO	NE	NE	NE
Planisware 5	ANO	ANO	ANO	ANO	NE
Project KickStart	NE	ANO	NE	ANO	NE
WorkPLAN Enterprise	ANO	ANO	ANO	ANO	NE

Příloha č. 3: Náhledy obrazovek systému Blue Camp



Příloha č. 4: Popis webové služby (WSDL soubor)

```
<?xml version='1.0' encoding='UTF-8'?><!-- Published by JAX-WS RI
at http://jax-ws.dev.java.net. RI's version is JAX-WS RI 2.2-hudson-
752-. --><!-- Generated by JAX-WS RI at http://jax-ws.dev.java.net.
RI's version is JAX-WS RI 2.2-hudson-752-. --><definitions
xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd" xmlns:wsp="http://www.w3.org/ns/ws-policy"
xmlns:wsp1_2="http://schemas.xmlsoap.org/ws/2004/09/policy"
xmlns:wsam="http://www.w3.org/2007/05/addressing/metadata"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:tns="http://ws.ucp/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns="http://schemas.xmlsoap.org/wsdl/"
targetNamespace="http://ws.ucp/" name="UseCasePlanService">
  <types>
    <xsd:schema>
      <xsd:import namespace="http://ws.ucp/"
schemaLocation="http://localhost:8080/UseCasePlan/UseCasePlanService?x
sd=1" />
    </xsd:schema>
  </types>
  <message name="countUCP">
    <part name="parameters" element="tns:countUCP" />
  </message>
  <message name="countUCPResponse">
    <part name="parameters" element="tns:countUCPResponse" />
  </message>
  <portType name="UseCasePlan">
    <operation name="countUCP">
      <input wsam:Action="http://ws.ucp/UseCasePlan/countUCPRequest"
message="tns:countUCP" />
      <output wsam:Action="http://ws.ucp/UseCasePlan/countUCPResponse"
message="tns:countUCPResponse" />
    </operation>
  </portType>
  <binding name="UseCasePlanPortBinding" type="tns:UseCasePlan">
    <soap:binding transport="http://schemas.xmlsoap.org/soap/http"
style="document" />
    <operation name="countUCP">
      <soap:operation soapAction="" />
      <input>
        <soap:body use="literal" />
      </input>
      <output>
        <soap:body use="literal" />
      </output>
    </operation>
  </binding>
  <service name="UseCasePlanService">
    <port name="UseCasePlanPort" binding="tns:UseCasePlanPortBinding">
      <soap:address
location="http://localhost:8080/UseCasePlan/UseCasePlanService" />
    </port>
  </service>
</definitions>
```

```

<?xml version='1.0' encoding='UTF-8'?><!-- Published by JAX-WS RI at
http://jax-ws.dev.java.net. RI's version is JAX-WS RI 2.2-hudson-752-.
--><xs:schema xmlns:tns="http://ws.ucp/"
xmlns:xs="http://www.w3.org/2001/XMLSchema" version="1.0"
targetNamespace="http://ws.ucp/">

<xs:element name="countUCP" type="tns:countUCP" />

<xs:element name="countUCPResponse" type="tns:countUCPResponse" />

<xs:complexType name="countUCP">
<xs:sequence>
<xs:element name="metrics" type="tns:ucpWsMetrics" minOccurs="0" />
<xs:element name="xmlModel" type="xs:string" minOccurs="0" />
</xs:sequence>
</xs:complexType>

<xs:complexType name="ucpWsMetrics">
<xs:sequence>
<xs:element name="e1" type="xs:int" />
<xs:element name="e2" type="xs:int" />
<xs:element name="e3" type="xs:int" />
<xs:element name="e4" type="xs:int" />
<xs:element name="e5" type="xs:int" />
<xs:element name="e6" type="xs:int" />
<xs:element name="e7" type="xs:int" />
<xs:element name="e8" type="xs:int" />
<xs:element name="hoursPerUc" type="xs:double" />
<xs:element name="t1" type="xs:int" />
<xs:element name="t10" type="xs:int" />
<xs:element name="t11" type="xs:int" />
<xs:element name="t12" type="xs:int" />
<xs:element name="t13" type="xs:int" />
<xs:element name="t2" type="xs:int" />
<xs:element name="t3" type="xs:int" />
<xs:element name="t4" type="xs:int" />
<xs:element name="t5" type="xs:int" />
<xs:element name="t6" type="xs:int" />
<xs:element name="t7" type="xs:int" />
<xs:element name="t8" type="xs:int" />
<xs:element name="t9" type="xs:int" />
</xs:sequence>
</xs:complexType>

<xs:complexType name="countUCPResponse">
<xs:sequence>
<xs:element name="return" type="tns:ucpElement" minOccurs="0"
maxOccurs="unbounded" />
</xs:sequence>
</xs:complexType>

<xs:complexType name="ucpElement">
<xs:sequence>
<xs:element name="complexity" type="xs:int" />
<xs:element name="hours" type="xs:double" />
<xs:element name="hoursPerUc" type="xs:double" />
<xs:element name="hoursSW" type="xs:double" />
<xs:element name="id" type="xs:string" minOccurs="0" />

```

```
<xs:element name="idParent" type="xs:string" minOccurs="0" />
<xs:element name="name" type="xs:string" minOccurs="0" />
<xs:element name="schneiderwinters" type="xs:double" />
<xs:element name="subCasesHours" type="xs:double" />
<xs:element name="subCasesHoursSW" type="xs:double" />
<xs:element name="subUCP" type="xs:double" />
<xs:element name="subUUCP" type="xs:int" />
<xs:element name="type" type="xs:string" minOccurs="0" />
<xs:element name="ucp" type="xs:double" />
</xs:sequence>
</xs:complexType>
</xs:schema>
```

Příloha č. 5: Kontrolní otázky pro metodu HOS 8

Sady kontrolních otázek metody HOS8 pro oblasti hardware, software, orgware, peopleware, dataware, customers, suppliers a management IS. Převzato z (12) a doplněno o odpovědi.

Oblast Hardware:

Příloha I Otázky metody HOS 8
--

1) Je možné současné HW vybavení označit za moderní a sledující současné trendy ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☒	☐	☐

2) Přispívá HW pozitivně k rychlosti a použitelnosti informačního systému ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☒	☐	☐	☐

3) Nákup nového HW je posuzován s ohledem na ergonomii pro jeho uživatele ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☒	☐	☐

4) Dá se připojení k počítačovým sítím označit za spolehlivé, dostatečně rychlé a vyhovující ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☒	☐	☐	☐

5) Jsou klíčové prvky HW dostatečně fyzicky chráněny před krádeží, požárem a povodní ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☐	☒	☐

6) Je nové HW vybavení pořizováno po zvážení jeho kompatibility s existujícím HW vybavením a softwarem, který na něm bude provozován ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☒	☐	☐	☐	☐

7) Současné HW neumožňuje účinnou výměnu dat s odběrateli či dodavateli ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☒	☐	☐	☐

8) Je rychle dostupné záložní vybavení v případě výpadku klíčových HW prvků systému ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☒	☐	☐

9) Souhlasíte s výrokem, že současné HW vybavení bude do dvou let těžko použitelné ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☒	☐	☐

10) Jsou poruchy HW vybavení na denním pořádku ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☐	☐	☒

Volná poznámka k oblasti: _____

Oblast Software:

1) Poskytuje zkoumaný software všechny funkce nezbytné pro práci uživatelů ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☐	☐	☒

2) Je grafické členění plochy pro zadávání, editaci vstupních údajů přehledné a přispívá tak ke snadnosti práce se systémem ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☒	☐	☐

3) Jsou chybová, varovná hlášení či jiné nestandardní oznámení srozumitelná a poskytují na požádání i bližší vysvětlení vzniklé situace ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☒	☐	☐

4) Rychlost zpracování úkolů jako tisky, dotazy, vyhledávání se jeví jako dostatečně rychlé ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☒	☐	☐	☐

5) Platí, že koncoví uživatelé nesmějí poskytovat podněty pro případné úpravy SW, nové nastavení nebo pořízení nových verzí software ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☒	☐	☐	☐

6) Je nápověda k softwaru srozumitelná a přehledná ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☐	☐	☒

7) Má zkoumaný informační systém jednotné ovládání obrazovek, menu, sestav a nápovědy ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☒	☐	☐

8) Jsou při pořízení nových verzí SW využívány jejich nové vlastnosti ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☒	☐	☐	☐

9) Je pravda, že snadnost používání softwaru koncovými uživateli nehraje roli při jeho pořízení nebo vývoji ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☒	☐	☐	☐

10) Existují pravidelné nebo nahodilé kontroly sloužící ke zjištění abnormalit ve využívání systému, jeho nesprávného užívání či zneužívání ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☐	☒	☐

Volná poznámka k oblasti: _____

Oblast Orgware:

1) Existují postupy či směrnice pro zotavení IS z nestandardních a havarijních situací a jsou tyto dokumenty dostatečně známy uživatelům ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☐	☐	☒

2) Existují doporučené pracovní postupy a procedury běžného provozu pro koncové uživatele a jsou udržovány v aktuálním stavu ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☐	☒	☐

3) Existují pravidla pro bezpečnost IS a obsahují i ustanovení pro nakládání s dokumenty či přílohami e-mailů získaných z Internetu ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☒	☐	☐

4) Je pravda, že management příliš nedozírá na dodržování pravidel bezpečnosti a provozu IS ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☒	☐	☐	☐

5) Má každý pracovník jasně určeno, s jakými úlohami smí pracovat a kdy ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☒	☐	☐	☐

6) Provádějí jakékoliv rozsáhlejší instalace, změny nastavení, připojení nové techniky pověřené osoby, nikoliv uživatelé?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☒	☐	☐	☐	☐

7) Jsou ošetřeny odchody zaměstnanců a ukončení platností jejich přístupových práv ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☒	☐	☐	☐

8) Existují pravidla nebo politika bezpečnosti IS a jsou tyto pravidelně aktualizovány ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☐	☒	☐

9) Umožňuje informační systém efektivní výměnu informací mezi uživateli IS v podniku?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☐	☐	☒

10) Platí, že pravidla pro provoz a bezpečnost IS jsou nejasná a nelogická ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☒	☐	☐

Volná poznámka k oblasti:

Oblast Peopleware:

1) Je každý pracovník zaškolen na úlohy, které má s informačním systémem provádět ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☒	☐	☐	☐

2) Jsou dostupná školení nových pracovníků o používaných informačních systémech, pravidlech provozu a bezpečnosti IS ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☒	☐	☐

3) Je pravda, že stávající zaměstnanci není třeba školit na nové funkce IS a že školení není dostupné

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☒	☐	☐	☐

4) Existuje zastupitelnost koncových uživatelů, kteří jsou klíčoví pro chod systému a jeho klíčové výstupy ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☒	☐	☐

5) Je dokumentace běžných postupů práce s IS jednoduše dosažitelná pro koncové uživatele ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☐	☒	☐

6) Je si management vědom vlivu firemní kultury na způsob práce koncových uživatelů s informačním systémem ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☒	☐	☐

7) Jsou dostupná místa uvnitř firmy nebo u externího dodavatele, kam se mohou uživatelé obracet se žádostí o pomoc či konzultaci ohledně IS ? (tato místa jsou označována dále jako informační centra)

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☒	☐	☐

8) Řeší informační centra z předchozího bodu podněty uživatelů obvykle v dostatečné míře a včas ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☐	☒	☐

9) Je pravda, že informační centra především „hasí“ palčivé problémy a nemají důvod se snažit o dlouhodobé zlepšení chodu IS ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☒	☐	☐	☐

10) Podporuje vedení firmy učení koncových uživatelů a jejich školení za účelem zvýšení efektivity fungování IS ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☐	☒	☐

Volná poznámka k oblasti: _____

Oblast Dataware:

1) Mají pracovníci jasně vymezenou odpovědnost za data, která spravují ? Tedy platí zásada, že určitá data smí měnit jen určitý pracovník ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☒	☐	☐	☐	☐

2) Mají pracovníci určeno, kdy musí jaká data zavést do informačního systému a kdy je musí aktualizovat ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☒	☐	☐	☐	☐

3) Platí, že uživatelům chybí z informačního systému data pro jejich rozhodování ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☒	☐	☐	☐	☐

4) Získávají koncoví uživatelé nadbytečná nebo nepřesná data ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☐	☒	☐

5) Musí pracovníci správy IS pravidelně provádět zálohování dat a dozírá management na dodržování pravidel zálohování ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☒	☐	☐	☐

6) Uznává management důležitý význam koncových uživatelů pro integritu a správnost zpracování dat ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☒	☐	☐	☐

7) Existují podrobné plány pro obnovu klíčových dat v informačním systému ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☐	☐	☒

8) Jsou média se zálohami dostatečně katalogizována a chráněna před zneužitím, krádeží či živelnou pohromou ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☒	☐	☐

9) Je bezpečnost dat zvažována a řízena i pro hrozby z Internetu nebo jiných počítačových sítí ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☒	☐	☐	☐

10) Mají pracovníci určeno, s jakými daty smí pracovat a s jakým oprávněním ? Platí tedy zásada, že nikdo nesmí získat přístup k datům, která nepotřebuje pro svou práci ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☒	☐	☐	☐	☐

Volná poznámka k oblasti:

Oblast Customers:

1) Jsou jasné stanoveny základní cíle zkoumaného informačního systému směrem k jeho zákazníkům ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☒	☐	☐	☐	☐

2) Existují metriky cílů uvedených v předchozím bodu a jsou dostatečně vyhodnocovány ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☒	☐	☐

3) Je pravidelně zkoumáno, jaké přínosy od informačního systému jeho zákazníci očekávají ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☐	☒	☐

4) Je pravda, že názory zákazníků IS na zlepšení, změnu či úpravu informačního systému nejsou pro podnik důležité ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☐	☒	☐

5) Jsou data o zákaznících IS, jejich požadavcích, operacích, atd. ukládány v informačním systému centrálně (tj. nejsou ukládány vícekrát nebo jinak nekonzistentně) ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☒	☐	☐	☐	☐

6) Přispívá současné hardwarové a softwarové vybavení k dostatečně rychlým odezvám na požadavky zákazníků IS ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☒	☐	☐

7) Je forma výstupů z informačních systémů volena tak, aby umožňovala jejich snadné využití zákazníkem IS?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☒	☐	☐

8) Ošetřují pravidla provozu nakládání s citlivými či obchodně cennými daty o zákaznících IS?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☒	☐	☐

9) Je řízena integrace zkoumaného informačního systému firmy spolu s dalšími IS podniku, které poskytují výstupy pro dané zákazníky ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☐	☒	☐

10) Mohou zákazníci získávat ze zkoumané IS výstupy pomocí různých komunikačních kanálů, které si zvolí ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☒	☐	☐

Volná poznámka k oblasti:

Oblast Suppliers:

1) Jsou jasně stanoveny základní požadavky kladené na dodavatele, které jsou nezbytné pro plnění definovaných cílů zkoumaného informačního systému ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☒	☐	☐

2) Existují metriky hodnocení výše zmíněných požadavků a jsou dostatečně vyhodnocovány ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☒	☐	☐

3) Je forma vstupů do zkoumaného IS od dodavatelů volena tak, aby umožňovala jejich snadné převzetí a využití zkoumaným IS ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☒	☐	☐	☐

4) Jsou v pravidlech provozu definovány kontroly informací od dodavatelů ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☐	☒	☐

5) Jsou požadavky na dodavatele ve vztahu ke vstupům do zkoumaného IS formulovány tak, aby byla jasně určena požadovaná podrobnost předávaných informací ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☒	☐	☐	☐

6) Jsou požadavky na dodavatele ve vztahu ke vstupům do zkoumaného IS formulovány také s jasným určením požadované včasnosti jejich dodávání ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☒	☐	☐	☐

7) Zvažuje firma možnost účelného přizpůsobení či nastavení zkoumaného IS dle návrhů dodavatelů za účelem efektivnější výměny informací ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☒	☐	☐	☐

8) Je forma výstupů ze zkoumaného IS pro dodavatele řízena s ohledem na efektivní komunikaci s dodavateli ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☒	☐	☐

9) Je pravda, že výstupy z IS pro dodavatele nejsou řízeny s ohledem na včasnost jejich předání ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☒	☐	☐

10) Přispívá zkoumaný informační systém ke snadnosti a efektivnosti komunikace s dodavateli ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☐	☒	☐

Volná poznámka k oblasti: _____

Oblast Management IS:

1) Trvají manažeři na dodržování pravidel stanovených pro informační systém ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☒	☐	☐	☐	☐

2) Provádí řízení rozvoje a provozu informačních systémů osoba, která této oblasti rozumí ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☒	☐	☐	☐	☐

3) Je rozvoj IS formulován také ve střednědobé či dlouhodobé perspektivě formou informační strategie vzhledem k cílům firmy ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☐	☒	☐

4) Je v plánech rozvoje informačních systémů zahrnut případný růst firmy a rozvoj jejích informačních potřeb ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☒	☐	☐

5) Platí, že plány rozvoje IS neexistují nebo v nich nejsou stanoveny možnosti kontroly jejich plnění ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☒	☐	☐	☐

6) Je při plánech rozvoje informačního systému, pořizování IS provedeno obhájení dané investice z ekonomického hlediska?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☒	☐	☐

7) Považuje management informačních systémů koncové uživatele za faktor s vysokou důležitostí pro úspěšný chod informačních systémů ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☒	☐	☐	☐	☐

8) Usiluje management IS soustavně o zlepšení efektivnosti chodu zkoumaného informačního systému ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☐	☒	☐	☐

9) Vnímá obecný management informační systém firmy nejen jako výdaje, ale také jako potenciál případného růstu firmy ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☒	☐	☐	☐

10) Podporuje obecný management firmy rozvoj informačních systémů, který je odůvodněný přispěním IS k dosažení podnikových cílů ?

Ano	Spíše ano	Částečně	Spíše ne	Ne
☐	☒	☐	☐	☐

Volná poznámka k oblasti:

Příloha č. 6: Obsah přiloženého CD

Složka *UCPlan/UseCasePlan*:

- Zdrojové kódy modulu *Use Case Plan*.

Složka *UCPlan/UcpDesktopClient*:

- Zdrojové kódy testovací aplikace *Use Case Plan Client Application*.

Složka *Dokumentace*:

- Dokumentace k modulu *Use Case Plan*.

Složka *Dotazníky HOS 8*:

- Naskenované a vyplněné dotazníky metody HOS 8.