

Západočeská univerzita v Plzni

Fakulta aplikovaných věd

Katedra informatiky a výpočetní techniky

Diplomová práce

Řízení a procesní mechanismy malých softwarových projektů

ZDE BUDE ORIGINAL ZADÁNÍ!

Prohlášení

Prohlašuji, že jsem diplomovou práci vypracovala samostatně a výhradně s použitím citovaných pramenů.

V Plzni dne 1. 6. 2016

Veronika Švecová

Poděkování

V první řadě bych ráda poděkovala panu doc. Ing. Přemyslu Bradovi MSc. Ph.D., vedoucímu práce, za jeho odborné rady a inspirativní způsob vedení práce.

Poděkování patří následujícím firmám, ve kterých se mi věnovali:

Firma K, která si přála být anonymizována

AIMTEC, pan Ing. Jan Brnka

CCA, pan Ing. Petr Příbyl

Unicorn Systems, pan Ing. Aleš Klein a pan Ing. Miroslav Hauser

za poskytnutí cenných informací a trpělivost při konzultacích.

Poděkování patří všem studentům a týmům, kteří na projektu pracovali a především pak Bc. Janu Kotalíkovi, bez kterého by nikdy nevznikla první funkční verze hry.

Za pomoc s korekturou bych ráda poděkovala Janu Smitkovi a Janu Kotalíkovi.

Nakonec bych ráda poděkovala svému příteli, Petru Kukrálovi, za podporu.

Abstract

This thesis is dedicated to the organisation of the Space Traffic Project; from taking it on to handing it over to the next leader. The main purpose of the project "Space Traffic" is to propagate the Department of Computer Science and Engineering University of West Bohemia into secondary schools and at the same time to teach students of secondary school programming. A university student will lead the project and verify their work within the project, gaining experience in project organization and leadership through work experience. The main of this thesis was to come up with a process to control the development of the project, document it, and create the first functional version of the project. It was vital to implement rules, instructions and tools which would serve to verify the process of work – most importantly with a view to the quality of its intermediate results. It was also necessary to create mechanisms for archiving and transferring information and experience related to the leadership of the project with the aim of ensuring the long-term quality of its output. The next task of the leader of the project was to supervise the sub-projects under way, and to ensure their successful completion. Inspiration about the organisation of the project were obtained from the companies K, AIMTEC, CCA and Unicorn Systems.

Abstrakt

Práce se zabývá řízením projektu Space Traffic od jeho přebrání po jeho předání dalším vedoucím. Hlavní myšlenkou projektu Space Traffic je propagovat Katedru informatiky a výpočetní techniky Západočeské Univerzity v Plzni na středních školách a zároveň studenty středních škol učit zábavnou formou programovat. Student vysoké školy v roli vedoucího projektu si práci na projektu ověří a prohloubí své schopnosti projektového řízení v praxi. Hlavním úkolem této práce bylo vymyslet proces kontroly vývoje projektu, dokumentace a dovést projekt do první funkční podoby. Bylo nutné zavést pravidla, postupy a nástroje, které by sloužily pro ověřování postupu prací, zejména s ohledem na kvalitu jejich mezivýsledků. Dále bylo nutné vytvořit mechanismy pro archivování a předávání znalostí a zkušeností souvisejících s vedením projektu, s cílem zajistit dlouhodobou kvalitu jeho výstupů. Dalším úkolem vedoucího projektu bylo dohlížení na probíhající podprojekty a starání se o jejich úspěšné dokončení. Informace o řízení projektů byly získávány ve firmách K, která si přála být anonymizována, AIMTEC, CCA a Unicorn Systems.

Obsah:

1.	ÚVOD	1
2.	PROJEKTOVÉ ŘÍZENÍ	3
2.1	PROGRAM.....	3
2.2	PROJEKT.....	3
2.3	PROCES.....	8
2.4	SOFTWAREOVÉ PROCESY.....	9
2.5	AGILNÍ VÝVOJ SOFTWARE.....	12
3.	ANALÝZA PROJEKTŮ VE FIRMÁCH	15
3.1	FIRMA K.....	17
3.2	AIMTEC.....	22
3.3	CCA.....	30
3.4	UNICORN SYSTEMS.....	36
3.5	SHRNUTÍ POZNATKŮ ZÍSKANÝCH VE FIRMÁCH.....	44
3.6	Hlavní odpovědi z firem.....	45
3.7	Obecné informace o schůzkách.....	46
4.	ANALÝZA PROJEKTU SPACE TRAFFIC	47
4.1	Informace ze schůzky s původními tvůrci projektu.....	47
4.2	Technická specifikace projektu.....	48
4.3	Analýza projektu Space Traffic popisem situace.....	49
4.4	SWOT analýza projektu.....	53
4.5	Popis prvního roku vedení.....	54
4.6	Další zjištěné informace v průběhu vedení prvního roku.....	57
4.7	Motivace studentů.....	58
4.8	Shrnutí problémů projektu.....	59
5.	VÝBĚR VHODNÝCH MECHANISMŮ A PROVEDENÉ ZMĚNY	60
5.1	Převedení na GitHub.....	60
5.2	Redmine projektu.....	61
5.3	Vylepšení znalostní báze.....	61
5.4	Kontrola celého projektu.....	63
5.5	Kontroly shora.....	63
5.6	Kontroly zdola.....	64
5.7	Přijímání nového člena projektu.....	65
5.8	Hodnocení splněných semestrálních prací.....	65
5.9	Nástroje pro komunikace v projektu.....	66
5.10	Zadávání semestrálních prací v projektu.....	66
5.11	Shrnutí odstranění hrozeb SWOT analýzy.....	67
5.12	Další zjištěné informace v průběhu vedení druhého roku.....	68
5.13	Popis druhého roku vedení.....	68
5.14	Propagace projektu Space Traffic.....	70
5.15	Hlavní zavedené mechanismy.....	71
5.16	Viditelná změna.....	71
6.	ZHODNOCENÍ ZKUŠENOSTÍ A DOPORUČENÍ PRO DALŠÍ VÝVOJ	73
6.1	Hodnocení vedení projektu členy týmu.....	74
6.2	Zhodnocení dosažených cílů.....	74
6.3	Hlavní doporučení pro úspěšné vedení projektu.....	75
7.	ZÁVĚR	77
	REFERENCE	III
A.	Kontrolní seznamy.....	VI
B.	Přijímání nového člena do projektu.....	IX
C.	Analýza firemních projektů.....	XI
D.	Analýza projektu Space Traffic při přebírání projektu.....	XII
E.	Analýza projektu Space Traffic při jeho ukončování.....	XV
F.	Zkrácená verze analýzy projektu:.....	XIX
G.	Komunikace: (krátký dotazník členům týmu v průběhu práce).....	XX
H.	Komunikace: (krátký dotazník členům týmu na konci práce).....	XXI
I.	Komunikace: (vyhodnocení spolupráce týmem Th3 Developers).....	XXII

1. Úvod

Projekt Space Traffic je projektem webové hry podporované Katedrou informatiky a výpočetní techniky Západočeské univerzity v Plzni. Hlavní myšlenkou tohoto projektu je propagovat Katedru informatiky a výpočetní techniky na středních školách a zároveň studenty středních škol učit nenásilnou formou programovat. Ve hře se student stane součástí vesmíru, ve kterém obchoduje se zbožím. Pro přepravu zboží mezi planetami používá vesmírné lodě. Pro pohyb lodí mezi planetami hráč musí psát vlastní skripty, aby maximálně optimalizoval své cesty.

Každý rok vedení projektu převezme student navazujícího studia na Katedře informatiky a výpočetní techniky v Plzni. V roce, kdy se stane vedoucím projektu, se stará o celý projekt od začátku do konce. Student vysoké školy v roli vedoucího projektu si práci na projektu ověří a prohloubí své schopnosti projektového řízení v praxi. Dalším úkolem vedoucího projektu je dohlížení na probíhající podprojekty a starání se o jejich úspěšné dokončení.

Projekt Space Traffic je vyvíjen již několik let v rámci studentských projektů, které studenti zpracovávají v rámci semestrálních prací z předmětů Katedry informatiky a výpočetní techniky Západočeské univerzity v Plzni. Hlavním úkolem této práce je vymyslet proces kontroly vývoje projektu a jeho dokumentace. Také je nutné dovést projekt do první funkční podoby. Bude nutné zavést pravidla, postupy a nástroje, které budou sloužit pro ověřování postupu prací, zejména s ohledem na kvalitu jejich mezivýsledků. Dále bude nutné vytvořit mechanismy pro archivování a předávání znalostí a zkušeností souvisejících s vedením projektu, s cílem zajistit dlouhodobou kvalitu jeho výstupů.

Tato diplomová práce je shrnutím poznatků z vedení projektu Space Traffic v letech 2014/2015 a 2015/2016. Vzhledem k tomu, že vedení projektu v tomto případě bude trvat dva roky, první rok se bude vycházet z původních pravidel nastíněných v diplomové práci Petra Vogla [1] z roku 2010. V druhém roce pak budou na základě zjištěných skutečností navržena a zavedena vylepšení, která povedou především k hladkému předávání projektu Space Traffic mezi studenty a ke zlepšení předávání znalostí o projektu.

První část práce je věnována projektovému řízení a bude vysvětlovat základní pojmy z této oblasti. Další část práce se bude zabývat analýzou projektového řízení v praxi v konkrétních firmách. Dále je zde bude provedena analýza studentského projektu Space Traffic a budou zde uvedeny jeho problémy. Na základě získaných informací z praxe a zjištěných problémů projektu Space Traffic pak budou vybrány a v druhém roce vedení prakticky vyzkoušeny mechanismy pro podporu projektu. V poslední kapitole budou shrnuty zkušenosti a uvedena doporučení pro další vývoj projektu Space Traffic.

2. Projektové řízení

Projektovým řízením [2] se rozumí využití všech dovedností, nástrojů, technik a poznatků o vedení projektu k tomu, aby byly splněny všechny požadavky projektu, především pak všechna přání zákazníka. Plnění požadavků projektu, nebo také projektových cílů, má na starosti projektový manažer. Projektový manažer má na starosti vedení projektu od jeho počátku (vzniku) k jeho konci (zániku). Proces řízení projektu se skládá z pěti fází [3].

- Definování
- Plánování
- Vedení
- Sledování
- Ukončení

V této kapitole bude vymezen pojem program, projekt a proces. Dále zde budou stručně popsány procesy řízení projektů. Projektové řízení je velmi rozsáhlá problematika. Tato teoretická kapitola je tak pouze shrnutím nejzákladnějších a pro tuto práci nejdůležitějších informací.

2.1 Program

Pokud projekt nemá jasně vymezený konec a skládá se z množství projektů, může se jednat o program. Podle [4] je program rozsáhlý projekt s rozsáhlým cílem. Ke splnění tohoto cíle je potřeba nejprve plnit cíle jednotlivých projektů, které postupně přispívají k celkovému cíli. Program je tedy rozsáhlá skupina věcně souvisejících řízených projektů.

2.2 Projekt

Projekt je [5] soubor činností, které zajistí přeměněnu vstupů na výstupy pomocí zdrojů. Přeměnu vstupů na výstupy v průběhu projektu zajišťují jednotlivé procesy. Projekt je jedinečný a neopakovatelný [3]. Má svůj pevně stanovený začátek a pevně stanovený konec. Projekt má trojrozměrný cíl [3], který také jinak nazýváme projektový trojimperativ (kapitola 2.2.1).

Podle systémového přístupu k řízení projektu [4] rozlišujeme v projektu produkt a proces. Produkt dává odpověď na otázku „CO?“ Procesy, což jsou vlastně činnosti a aktivity a pak dávají odpověď na otázku „JAK?“

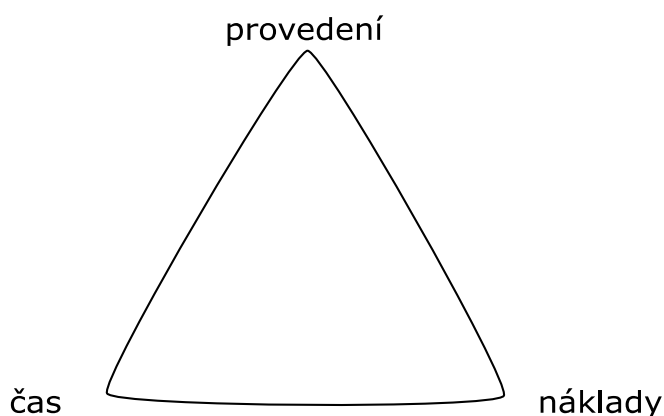
Projekty lze rozložit na části. Typické způsoby rozložení projektů [4] jsou podle práce, organizace, nákladů, toků informací, dokumentací, portfolia i programů.

2.2.1 Cíle projektu

Projekty mají trojrozměrný cíl, který je také podle [3] označován jako projektový trojimperativ. Pro úspěšné vedení projektu potřebujeme mít měřitelné tyto tři cíle:

- Náklady
- Čas (termíny)
- Provedení (kvalita)

Tyto cíle se navzájem ovlivňují, jak ukazuje Obrázek 2.1. Udržení těchto cílů v rovnováze je podmínkou úspěšného projektového řízení. Udržení cílů v rovnováze je úkolem projektového manažera. Ze strany zákazníka většinou bývá tlak na výborné provedení jeho požadavků v minimálním čase a za minimální náklady.



Obrázek 2.1: Projektový trojimperativ

George T. Doran definoval ve svém článku pro Management Review cíle projektu pomocí metody SMART [2]. Podle tohoto článku by každý nový navrhovaný cíl projektu měl splňovat následující vlastnosti:

- Specific (specifický, konkrétní)
- Measurable (měřitelný, možný řídit)
- Achievable (dosažitelný)
- Realistic (realistický)
- Time-bound (ohraničený v čase)

Pokud cíl projektu splňuje všechny uvedené vlastnosti, mělo by být možné ho dosáhnout.

2.2.2 Zahájení projektu, definování projektu

Po získání projektu realizační firmou začíná zahajovací fáze. V této fázi je hlavním úkolem vytvořit dokument studie proveditelnosti, který definuje projekt a určuje požadavky na projekt. Dokument specifikace požadavků, který navazuje na studii o proveditelnosti, se věnuje vymezení cílů projektu, předpokladů, kritérií úspěšnosti projektu, a požadavků na zdroje a na rozpočet projektu [4].

2.2.3 Plán projektu

Plán projektu je model projektu včetně simulace různých situací, které mohou v průběhu projektu nastat. Plán projektu by měl obsahovat časový plán, finanční plán a plán realizace projektu.

Časový plán projektu

Časový plán projektu slouží k uspořádání jednotlivých činností projektu do časových návazností. K vytvoření časového plánu slouží například kritická cesta, PERT (pesimistický a optimistický odhad), Ganttův diagram, nebo síťový diagram. Všechny vyjmenované varianty jsou podrobněji popsány v [4].

Plán na náklady projektu

Dále je nutné stanovit náklady projektu. Náklady na projekt mohou být vyčísleny počtem potřebných pracovních hodin [3], nebo vyjádřením času v penězích. Většinou existují hodinové sazby, které přepočít umožňují.

Plán realizace projektu

Projekt musí mít plán realizace od svého počátku do svého konce.

2.2.4 Určení rizik projektu

Rizika jsou součástí každého projektu. Můžeme je dělit na rizika hmotná a psychologická [3]. Na hmotná rizika je možné vyčlenit tzv. finanční polštář, který v případě, že riziko nastane, bude sloužit k náhradě škod. Horší situace nastává u rizik psychologických, kde například ve fázi plánování projektu dochází k příliš pesimistickým, nebo naopak optimistickým odhadům.

U rizik projektu je nutné určit aktiva, hrozby a zranitelnosti. Jejich vynásobením a vložení do tabulky je možné zjistit nejhorší rizika, pro která jsou pak sestavené rizikové plány. Menší rizika mohou být označena za akceptovatelná rizika.

2.2.5 Realizace projektu

Ve fázi realizace projektu dochází k plnění jednotlivých plánů z fáze plánování projektu. Je nutné určit výkonné procesy, možnosti kontroly procesů a způsob měření úspěšnosti projektu.

2.2.6 Kontroly procesů

Každý proces na projektu musí být měřitelný a kontrolovatelný. Díky měření mezivýsledků projektu je možné určit, jak si projekt vede v jeho průběhu i na jeho konci. Je možné včas identifikovat například odchylku nákladů, odchylku od časového plánu, efektivitu nákladů a další. Měření procesů je zpětnou vazbou pro projektového manažera.

2.2.7 Projektová znalostní báze

Znalostní báze obsahuje veškerou přehledně členěnou dokumentaci o projektu. Jsou zde specifikovány druhy projektových dokumentů a způsob jejich ukládání. Dále by měla existovat pravidla pro to, jak dokumenty uchovávat, třídit a vytvářet [6].

2.2.8 Ukončení projektu

Projekt je považován za ukončený, pokud bylo dosaženo projektových cílů specifikovaných v dokumentu specifikace požadavků. Nejdůležitějším aspektem je ale spokojenost zákazníka [3]. Může nastat situace, kdy jsou sice splněny všechny cíle projektu, ale zákazník není spokojen. V tomto případě pravděpodobně vznikla chyba už ve fázi analýzy projektu.

Ve fázi ukončení projektu vznikne závěrečná zpráva, která obsahuje [7] vyhodnocení průběhu a realizace projektu, vyhodnocení efektivnosti, zhodnocení využití zdrojů, zhodnocení úrovně kvality produktu a zhodnocení financování projektu. Dále by mělo být vytvořeno prohlášení o ukončení projektu (Práce byla ukončena, nebudou již požadovány další finanční ani jiné prostředky. Byl vytvořen speciální fond, ze kterého se budou uhrazovat případné nedostatky. Byly vyrovnány všechny závazky.).

Vyhodnocení členů týmu je motivací pro členy týmu do budoucna (děkovný dopis, změna rolí v příštích projektech).

Shromáždění a analýza dat (rozbor a analýza dat o projektu, vyhodnocení efektivity, poučení se z chyb do budoucna, počítání různých ukazatelů a jejich porovnávání s ukazateli jiných projektů) slouží k vyhodnocení projektu [8]. Tato fáze je nutná k poučení se z vlastních chyb a optimalizaci vývojového procesu pro příští projekty.

2.2.9 Lidské zdroje v projektech

Jedním z hlavních zdrojů projektu, jak bylo uvedeno v kapitole 2.2.3, je čas, který je vyjádřen odpracovanými hodinami pracovníků na projektu. Z celkového času stráveného na projektu vyplývají i náklady na projekt. Práce s lidmi je velice nepředvídatelná, ale zkušený projektový manažer by měl být na většinu situací vyškolen, případně by měl být schopný problémy řešit operativně. Při řízení lidí je důležité uvědomit si základní lidské potřeby [9], které zobrazuje Obrázek 2.2.



Obrázek 2.2: Maslowova pyramida potřeb

Jak je z obrázku patné, člověk potřebuje naplňovat různé druhy svých potřeb. Nejprve naplňuje potřeby v nejnižším patře pyramidy, protože se jedná o potřeby nejnaléhavější. Uspokojená potřeba nemotivuje a v případě uspokojení jedné potřeby se zpravidla objeví jiná.

2.3 Proces

Procesy jsou rozděleny do tří hlavních skupin [10]:

- Procesy hlavní
- Procesy řídicí
- Procesy podpůrné

Procesy hlavní jsou procesy oborově specifické. Jsou to procesy, které projekt nutně potřebuje ke svému zdárnému konci. Hlavní procesy také obvykle vytvářejí hodnoty a vedou k zisku. Procesy řídicí slouží k řízení cílů, výkonu, zdrojů a návaznosti procesů. Procesy podpůrné slouží k podpoře ostatních procesů [11].

Pro modelování procesů existuje řada nástrojů. V [10] jsou podrobněji popsány tyto způsoby modelování: znázornění toků, CASE nástroje-analýzy, simulační nástroje.

Procesy modelujeme hlavně z důvodu vyhledávání úzkých míst projektu. Pokud jsou totiž části projektu měřitelné, je možné odhalit jejich efektivitu a jejich další vlastnosti. Díky určení jednotlivých procesů a jejich měřitelnosti můžeme sbírat data o procesech a na jejich základě dělat rozhodnutí.

2.3.1 Nástrahy procesního přístupu

Nástrahy procesního přístupu jsou následující [10]:

- Soustředění se na úzká místa.
- Vedoucí pracovníci se neztotožnili se změnami.
- Nevhodnost krátkodobých řešení.
- Neměření procesů, pouze jejich zavedení.

2.3.2 Postup zavedení procesního přístupu

- Vytvoření zázemí pro změnu.
- Výběr nového procesu.
- Dokonalé zmapování situace.
- Zavedení standardů procesů.
- Změna procesů.
- Pokračování ve zdokonalování procesů.
- Hodnocení zlepšení procesů.

2.4 Softwarové procesy

Softwarový proces je [12] sada aktivit při produkci software. Softwarových procesů je celá řada, ale všechny zahrnují následující fáze:

- Specifikace (definice funkcí a omezení).
- Návrh a implementace (co musí software splňovat).
- Schválení (software musí být schválen zákazníkem).
- Možnost dalšího vývoje.

Softwarové procesy se dělí [12] na neagilní a agilní. U neagilních procesů je postupováno podle předem daného plánu. U agilních procesů je plánování inkrementální (přírůstkové) a produkt se přizpůsobuje přáním zákazníka.

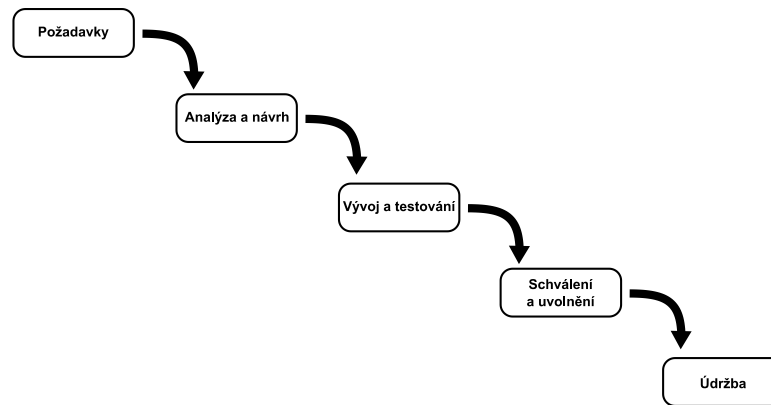
Neexistuje jednotný postup pro správný výběr procesu pro řízení projektu. Při výběru procesu je nutné zvážit velikost projektu, požadavky zákazníka, náklady, cíle projektu a mnoho dalších hledisek.

Softwarový proces lze zjednodušeně reprezentovat některým z následujících modelů [12].

2.4.1 Vodopádový model (sekvenční postup)

Vodopádový model je jedním z nejstarších modelů. Jednotlivé části vývoje se zde nepřekrývají a postupně na sebe navazují od počáteční fáze do konečné. Problém tohoto modelu spočívá v nemožnosti reagovat na zjištěné potíže během procesu. Pokud by totiž mělo dojít ke změně požadavků během vývoje, vodopádový model na tuto situaci neumí zareagovat. Dalším nedostatkem je vznik celého produktu způsobem tzv. velkého třesku, což může také přinést řadu problémů. Vodopádový model může být užitečný při implementaci projektu s malým rozsahem, u kterého je dopředu známý rozsah i obsah a změny během vývoje nejsou očekávány.

Vodopádový model je tedy sekvenční přístup, kdy na sebe jednotlivé fáze projektu navazují a nikdy se nevracíme, nebo neopakujeme předchozí fázi projektu, jak ukazuje Obrázek 2.3.



Obrázek 2.3: Vodopádový model

2.4.2 Inkrementální vývoj

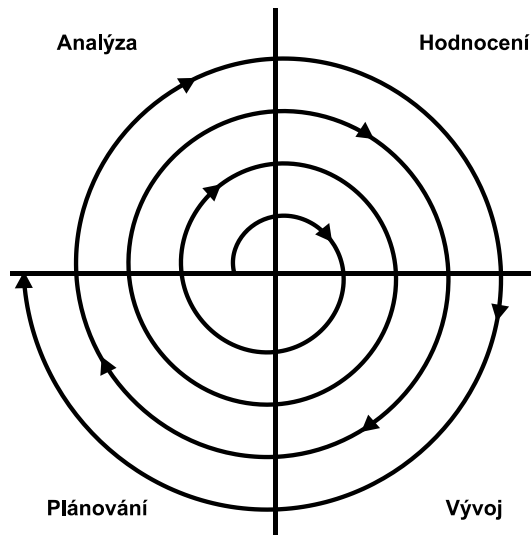
Inkrementální vývoj je založen na přírůstku. Nejprve je vytvořena počáteční implementace, která je předkládána uživatelům ke komentování. Na základě komentářů jsou prováděny změny, které jsou do systému přidávány ve formě přírůstků. Tento postup se opakuje až do fáze vzniku požadovaného systému. Inkrementální přístup kopíruje přístup, kterým lidé běžně řeší problémy. Má ale také své nedostatky. Velice těžko se kontroluje postup projektu a spolu s přidáváním nových inkrementů může docházet ke snižování kvality výsledného produktu.

2.4.3 Spirálový model (cyklický postup)

Spirálový model je model architektury softwarového procesu, který navrhl v roce 1988 Boehm. Tento model počítá s neustálým vylepšováním vývojového procesu. Je zobrazován pomocí spirály opakujících se kroků:

- Analýza (nastavení cílů)
- Hodnocení (omezení rizik)
- Vývoj
- Plánování

Postup spirálového modelu ukazuje Obrázek 2.4: Spirálový model.



Obrázek 2.4: Spirálový model

2.4.4 Proces RUP

RUP (The Rational Unified Process) je metodika vývoje software. Metodika je souhrnem praktik a postupů, které popisují a řeší situace v průběhu celého životního cyklu projektu (artefakty, milníky, role, aktivity). Metodika je tedy konkrétní proces vývoje produktu.

RUP se hodí spíše pro vývoj rozsáhlejších projektů, protože hlídá důraz na plánování, analýzu a design, řízení zdrojů a dokumentaci. RUP kombinuje části ze všech obecných modelů procesů [12].

Proces RUP vychází ze šesti Best Practices:

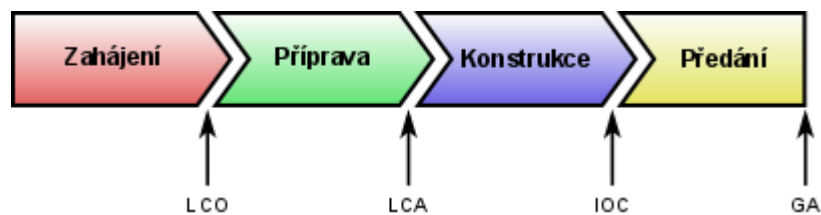
- Iterativní vývoj software.
- Aktivní správa požadavků.
- Architektura založená na komponentách.
- Vizuální modelování.
- Ověřování kvality software.
- Správa změn software.

Proces RUP se popisuje ze tří perspektiv:

- Dynamická (fáze modelu v čase).
- Statická (odehrávající se aktivity procesu).
- Postupová (optimální postupy pro nasazení v projektu).

RUP rozděluje proces vývoje na čtyři fáze. Každá z těchto fází se může skládat z jedné, nebo z více iterací. V každé fázi probíhají činnosti specifické pro tuto konkrétní fázi a také jsou zde činnosti a cíle, které se opakují v každé fázi. Fáze RUP ukazuje Obrázek 2.5 a jsou následující [13]:

- Zahájení (tzv. inception).
- Příprava (tzv. elaboration).
- Konstrukce (tzv. construction).
- Předávání (tzv. transition).



Obrázek 2.5: RUP

Milníky RUP

V procesu RUP jsou definovány čtyři milníky, které určují, v jaké fázi se projekt nachází. Každá fáze je ukončena dosažením konkrétního milníku [13].

- LCO (Lifecycle Objectivs).
- LCA (Lifecycle Architecture).
- IOC (Initial Operational Capability).
- PR (Product Release).

2.5 Agilní vývoj software

Agilní přístup umožňuje pružně reagovat na změny během vývoje software. Cíle projektu se vyvíjí spolu s projektem. Procesy jsou navrženy tak, aby byla možnost rychle reagovat na měnící se prostředí. K nejčastěji používaným agilním metodikám patří Extrémní programování a SCRUM [12]. Úspěšnost agilních metodik vedla k částečné integraci s tradičními metodami vývoje software. Výsledkem toho je například agilní verze RUP.

2.5.1 Extrémní programování

Extrémní programování, dále také XP, je způsob vyvíjení software. Je lehký, pružný, účinný a přizpůsobivý změnám. V XP se požadavky vyjadřují jako scénáře (označované jako uživatelské historie), uživatel tedy většinou plní roli člena vývojového týmu. Hlavními znaky extrémního programování jsou [14]:

- zpětná vazba vyplývající z krátkých cyklů a malých vydání
- přírůstkové plánování (vyvíjející se životní cyklus projektu)
- flexibilní, reflektující změny (zadání, či termínů)
- automatické testování, zachycení problémů včas (testování od programátorů i zákazníků)
- vypisování testů před psaním kódu
- spolupráce programátorů, propojení jejich zkušeností, kolektivní vlastnictví
- velký význam verbální komunikace
- návrh projektu se neustále vyvíjí (evoluční proces)

Základním problémem vývoje XP jsou projektová rizika [14]. XP se zabývá projektovým rizikem na všech úrovních projektu a s riziky se vypořádává přímo způsobem vývoje. Například riziko zpoždění časového plánu je vyřešeno tím, že krátké cykly nedovolí udělat velkou chybu v plánování. Míra poruchovosti je vyřešena sadami testů, které píšou jak zákazníci, tak programátoři. Ke každému riziku v XP existuje řešení, které je součástí této metodiky.

Vývoj je řízen testováním a programátoři většinou pracují ve dvojici. Také probíhají testy integrace. V projektech jsou rozeznávány a kontrolovány čtyři proměnné [14]: náklady, čas, kvalita a rozsah.

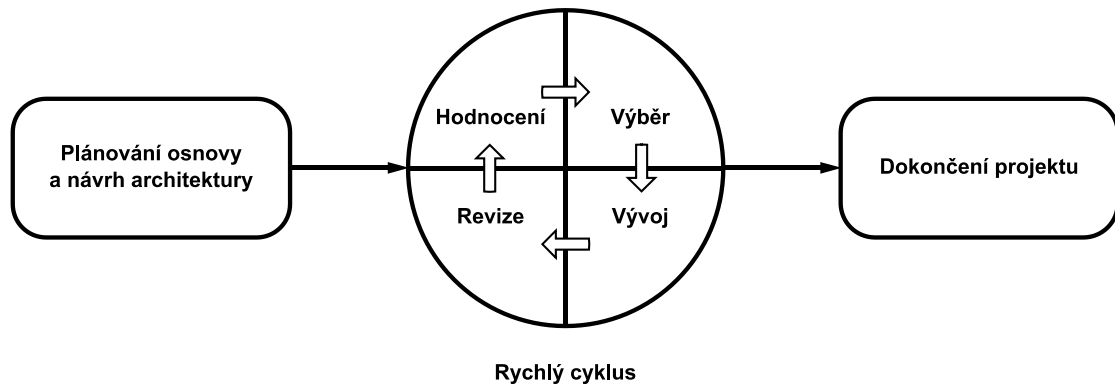
Čtyřmi hodnotami XP [14] jsou komunikace, jednoduchost (už v začátku je nutné dosáhnout největší jednoduchosti řešení), zpětná vazba a odvaha. Vycházejí ze základních principů XP, rychlé zpětné vazby, nutnosti osvojit si jednoduchost, dělat inkrementální změny a kontrolovat kvality práce.

Zavedení XP

Podle [14] je nutné najít nejhorší problém projektu a vyřešit ho pomocí XP. Ve chvíli, kdy to již není nejhorší problém, se tento postup opakuje. V již existujícím týmu lze zavést XP postupně (do testování, designu, plánování, managementu, vývoje).

2.5.2 Scrum

Scrum je obecnou agilní metodou [12]. Scrum je zaměřený na iterativní způsob vývoje.



Obrázek 2.6: Scrum

Jak ukazuje Obrázek 2.6, Scrum má tři fáze. První fází je plánování, kdy je naplánována osnova a je navržena architektura. Druhá fáze tohoto přístupu se skládá z řady rychlých cyklů, ve kterých se střídají fáze výběru, vývoje, revize a hodnocení. Každý z těchto cyklů představuje jeden přírůstek (tzv. inkrement) systému. V závěrečné fázi je projekt uzavřen a zdokumentován.

Rychlý cyklus SCRUM je plánovací jednotkou tohoto přístupu. Každý člen týmu má dostatečné pravomoci dělat rozhodnutí, takže zde není projektový manažer. SCRUM-master je osoba, která organizuje denní schůzky, zapisuje resty a sleduje vývoj projektu, měří postup projektu a komunikuje se zákazníky a managementem mimo tým.

Celý tým se účastní krátkých denních schůzek, které se pořádají ve stoje (stand up meeting). Zde se členové týmu seznamují s plánem na další den a uvádějí své překážky v práci, pokud nějaké jsou. Denním setkáním je zajištěno, že všichni členové týmu ví, co se má dělat.

3. Analýza projektů ve firmách

Tato část práce se zabývá pozorováním metod řízení a způsoby použití podpůrných nástrojů pro podporu softwarových projektů menšího rozsahu v konkrétních firmách. Zjištěné informace jsou pak v praktické části využity k podpoře projektu Space Traffic. Pro účely této analýzy byl nejprve vytvořen seznam konkrétních problémů, se kterými se projekt Space Traffic potýká. Na základě tohoto seznamu byl vytvořen seznam otázek, na které byly ve firmách hledány odpovědi. Ve čtyřech velkých firmách pak bylo zjišťováno, jak se tyto problémy řeší v praxi. Otázky byly zaměřeny především na:

- Zkoumání konkrétního projektu, či produktu (pokud to firma umožnila).
- Hlavní mechanismy, které firmy používají pro řízení a podporu projektů.
- Práci s lidskými zdroji.
- Práci se znalostní bází firmy a její udržování.

Kompletní seznam otázek je součástí přílohy C a F. Informace byly získávány konzultacemi a pozorováními prací na interních firemních projektech. V rámci této práce byly navštíveny firmy K, která si přála být anonymizována, AIMTEC, CCA a Unicorn Systems.

Firmy mají některé názory na vedení projektů společné. Artefakty společností vycházejí z RUP. Všechny společnosti uváděly, že používají Scrum. Po bližším zkoumání bylo zjištěno, že ve všech případech nejde o čistý Scrum, ale Scrum méně, nebo více upravený pro potřebám společnosti.

Ve společnostech K a AIMTEC je primárně vytvářen jeden velký produkt. V K tento produkt téměř není upravován na přání zákazníka. Ve společnosti AIMTEC zase vycházejí ze základního produktu, který pak na přání zákazníka přizpůsobí. Firmy CCA a Unicorn Systems připravují produkty na přání zákazníka, lze zde tedy narazit na různé projekty jak rozsahem, tak obsahem. Ve společnostech K a AIMTEC nebyla možnost získat informace o konkrétních produktech, ale bylo zjištěno, jak se firmy starají o svůj pilotní produkt. V CCA a Unicorn Systems bylo možné analyzovat konkrétní projekty. Zároveň při těchto konzultacích bylo zjištěno, že projekt Space Traffic podle některých definic nelze považovat za projekt, ale spíše za program, který se skládá z dílčích projektů. Srovnání projektu Space Traffic s těmito velkými produkty je z tohoto hlediska prakticky vhodné.

Zjištění vhodné metodiky vývoje je rozsáhlý problém a není prakticky možné zvážit všechny potřebné aspekty během několikahodinové konzultace. Pomocí připravených dotazníků, které jsou součástí přílohy C a F, byly převážně hledány situace během celého životního cyklu podniku, kdy dojde ke změně zdrojů, hlavně lidských. Bylo zjišťováno, jak firmy obecně řeší situace, kdy odejdou z projektu pracovníci. Obecně bylo zjištěno, že v praxi k situaci, kdy se vymění celý projektový tým, dojde zřídka. Většinou v projektu zůstává minimálně projektový manažer, který většinu informací zná, nebo ví, kde je má hledat. Situaci, kdy se lidé v projektu změní, jsem zaznamenala pouze v Unicorn Systems. Ke každému projektu, v každé z navštívených firem, je vedena rozsáhlá znalostní báze, včetně záznamů komunikace ze schůzek i telefonické komunikace. Dále byly hledány situace, kdy části projektu, nebo podprojektu, proběhne celý životní cyklus až do konce. Například K takovýto produkt zařadí k celému produktovému balíku, a pokud je v budoucnosti nutná jeho úprava, existuje natolik podrobná dokumentace, že je možné ji realizovat.

V této kapitole budou uvedeny výsledky analýz procesů firem a další zjištění. Také zde bude obsažen seznam informací o způsobech vývoje, které byly ve společnostech zjištěny. Komunikace ve firmách probíhala ve formě schůzek, kterých v každé firmě proběhl jiný počet. Také každý z manažerů, který schůzky vedl, považoval za důležitou jinou část projektového vedení a podnikové metodiky, proto bylo velmi obtížné zachovat u získaných informací jednotnou strukturu. Informace nakonec byly rozčleněny do několika stejných podkapitol, kterým je přisuzována různá důležitost, podle času, který jim byl v průběhu konzultací věnován. Také zde často přecházíme od konkrétního projektu k podnikovým metodikám obecně, protože v některých případech nebylo jednoduché metodiky od projektů oddělit. Obzvláště u firem, kde byla zpracovávána analýza konkrétního produktu, je toto přecházení patrné.

Na konci kapitoly bude provedeno porovnání, jak jednotlivé firmy přistupují k vedení projektů.

3.1 Firma K

3.1.1 Produkt

Tato firma si vyžádala anonymizaci. Firma K je mezinárodní softwarová firma o 200 zaměstnancích. Má dvě divize a zabývá se čtyřmi pilotními produkty. Nejedná se o software na zakázku, ale o produkty, které jsou neustále vyvíjeny a je do nich stále přidávána nová funkcionality (tzv. on-premises produkty). Vizi produktu dává Produktový viceprezident firmy. Ten také tvoří plán krátkodobých a dlouhodobých cílů produktu, (tzv. road map) produktu.

Vývoj produktu dále mohou ovlivnit sami zaměstnanci firmy, kteří přicházejí se svými nápady a podněty, jak produkt vylepšit. Některé z těchto nápadů jsou implementovány. Směřování produktu mohou také ovlivnit sami zákazníci s pomocí služby Uservoice. Tato služba umožňuje u každého produktu poslat zpětnou vazbu firmě. Pomocí těchto zpětných vazeb zákazník vlastně hlasuje o dalším vývoji produktu.

Zohledněním těchto aspektů se vytváří seznam požadovaných funkcí produktu (tzv. backlog) produktu. Backlog produktu obsahuje seznam funkcí podle jejich priority a základní časové odhady jednotlivých funkcí podle jejich pracnosti.

Celá tato vize projektu, která zohledňuje podněty zaměstnanců a hlasování uživatelů pomocí služby Uservoice, je pak poslána od zahraničního vedení zpět do České republiky, kde je předána manažerovi vývoje firmy, který řídí vývoj produktu.

Manažer vývoje podle konkrétní situace zpracuje dvě až tři varianty možného vývoje příští verze. Podle konkrétní situace je naplánován další postup. Je možné naplánovat celou novou funkcionality, nebo přepisování stávající funkcionality do novější technologie, nebo například naplánovat přesun aplikace do cloudu.

Naplánovaná funkcionality je přesunuta do backlogu naplánované verze. Z backlogu jsou pak plánovány jednotlivé sprinty podle metodiky Scrum. Všechna naplánovaná funkcionality by se v rámci sprintu měla stihnout nejen naprogramovat, ale i otestovat. Pokud se testování z nějakého důvodu nestihne, je naplánováno do dalšího sprintu.

3.1.2 Mechanismy a nástroje, bez kterých se nelze obejít

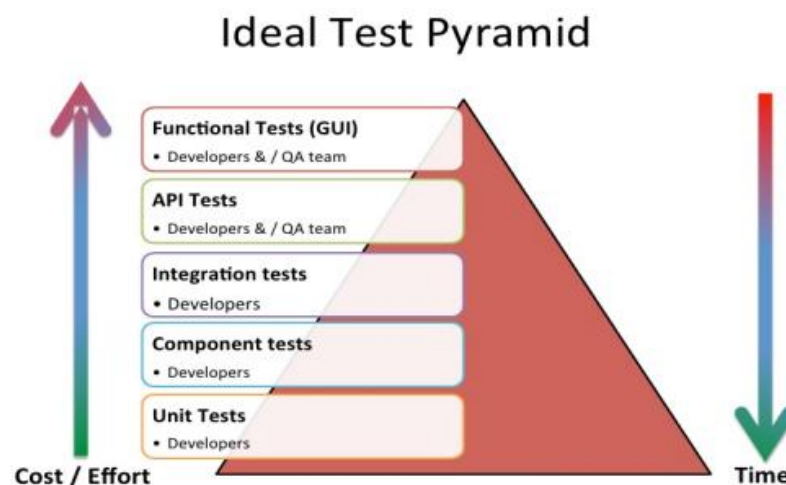
Systém pro správu verzí

Firma používá systém pro správu verzí Git. Programátoři přidávají změny a vkládají části své práce (tzv. push) přímo do hlavní vývojové větve Master. Pokud je hotová všechna funkcionalita, která byla naplánovaná do dané verze a verze je stabilní (neprobíhá mnoho změn), tak je z hlavní vývojové větve Master oddělena (tzv. branch) větev Release, větev pro vydání příslušné verze. Stabilní verze je pak novou verzí produktu. Branch je nutné udělat také v případě, že už je nutné začít pracovat na příští verzi produktu.

Ukazatele stability jsou rozebrány podrobněji v kapitole 3.1.4.

Automatická kompilace a automatické testování

Vkládání nových částí kódu přímo do větve Master je možné proto, že při každém vkládání proběhne automatické sestavení (tzv. autobuild). Autobuild probíhá pomocí systému Teamcity. Automatické sestavení probíhá pro každý nový commit. Pokud z nějakého důvodu dojde k chybě, nastane problém s testy, nebo sestavením. V této situaci je nutné kód opravit, znovu sestavit a otestovat. Na autobuild je navázané automatické testování. U funkčních testů je problém jejich dlouhé trvání, proto rozsáhlejší sady testů běží přes noc. Testy není pokrytá celá aplikace, ale momentálně roste snaha tento nedostatek zlepšit. V tomto momentě je nejvíce testů funkčních a nejméně jednotkových. Firma by si přála tento stav obrátit. Ideální stav testování ukazuje Obrázek 3.1., kde je nejvíce testů jednotkových a nejméně funkčních.



Obrázek 3.1: Ideální testovací pyramida

Automatické testování a automatická kompilace jsou mechanismy, které udržují produkt vždy funkční. Produkt se neustále vyvíjí. Je zde snaha mít co nejmenší úkoly, (tzv. task) a vkládat nové části kódu okamžitě, jak je to možné.

Software pro organizaci a řízení práce, firemních procesů a projektů

Pro organizaci a řízení práce používá firma nástroj pro řízení projektů JIRA. Zde jsou evidovány všechny úkoly (tzv. task), příběhy (tzv. story), celkový příběh (tzv. epic story) i chyby (tzv. bug). V nástroji JIRA je pak možné vidět přímo změnu v kódu související s řešenou chybou, nebo úkolem. Také je zde vidět, v jakém buildu je úkol (nebo chyba) dostupný. Nástroj JIRA je napojen na systém pro správu verzí Git. U každého úkolu je tedy možné zobrazit přímo změněné řádky.

V nástroji Teamcity je možné vidět, zda pro konkrétní build doběhl konkrétní test.

Ve firmě je využívána i agilní část nástroje JIRA, která mimo jiné nabízí možnost společného plánování úkolů. Probíhá pravidelné plánování podle story, u větších projektů například půlročních jsou pak plánovány epic story, které se rozpadají na story. Epic story jsou rozkládány metodou odshora dolů až na jednotlivé úkoly. Každý úkol musí být jasně popsán, aby přebírající programátor vždy rozuměl tomu, co má dělat. Dále je firmou používána část systému JIRA pro evidenci chyb, které jsou následně opravovány (tzv. bug fixing).

Akceptační testy

Firma má vlastní nástroj pro akceptační testy. Tester v tomto nástroji pouze zaznamenává, zda testovaná část funguje nebo ne.

3.1.3 Rozdělení týmů a lidské zdroje

Ve firmě fungují projektové týmy. V rámci produktových týmů je ve firmě několik oddělení, která se specializují na různé části produktu:

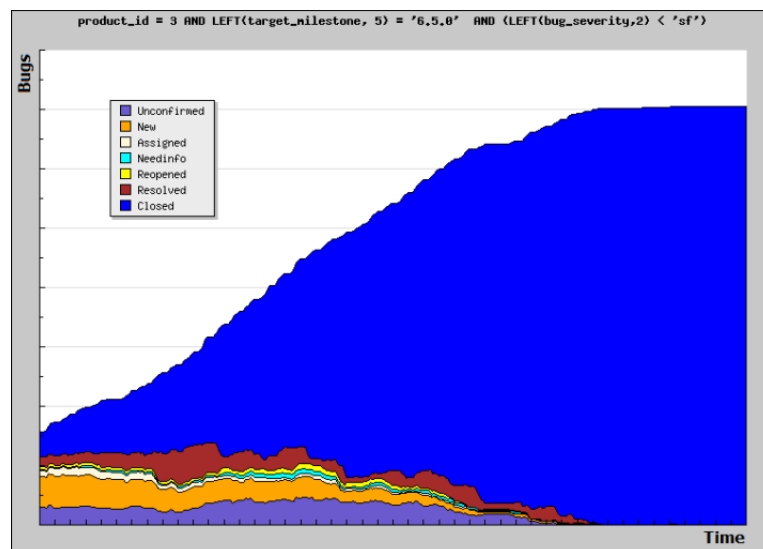
- Engine (c ++, programování nových funkcí v Javě, design)
- web frontend (různé druhy podpory, synchronizace s mobilním zařízením a další)
- ostatní klienti

Programátoři jsou zde zodpovědní za dodání určité vlastnosti (tzv. Feature). Nová verze produktu je plánována s ohledem na dostupné zdroje. Zaměstnanci mezi jednotlivými produkty nepřechází.

3.1.4 Verze k vydání

Produkt je možné vydat, pokud je dokončeno vše, co bylo v konkrétní verzi naplánováno. Vlastnosti (tzv. feature) jsou při plánování rozdělené do dvou skupin. První skupina obsahuje feature, které musí v konkrétní verzi být (tzv. must have). Druhá pak obsahuje feature, které ve verzi můžou být, ale nejsou pro vydání konkrétní verze závazně nutné (tzv. nice to have). Pokud se během obvykle půlroční práce ukáže, že některé naplánované feature nelze v konkrétní verzi stihnout, jsou tyto feature ze skupiny nice to have vyškrtnuté z plánu.

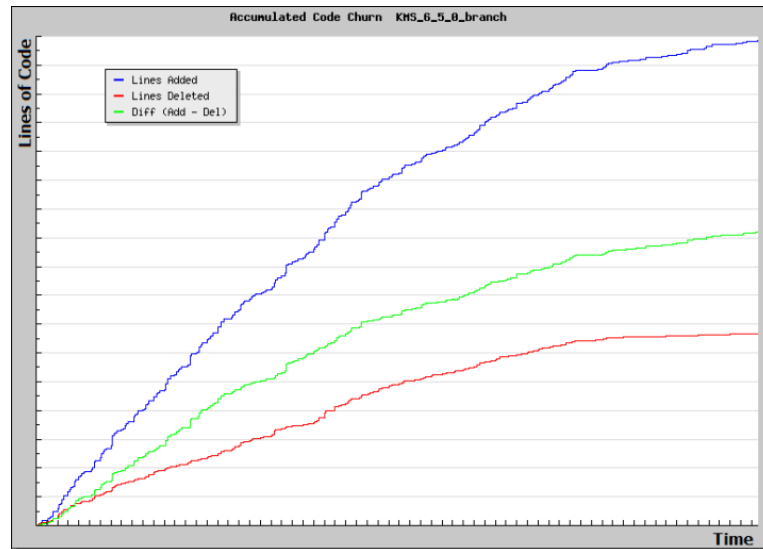
Vydána je pouze stabilní verze. Prvním signálem stability verze je počet chyb. Počet chyb se s přiblížením ke stabilní verzi produktu snižuje. Dokud jsou stále hlášeny nové chyby, které se opravují, produkt ještě není ve stabilní verzi, jak ukazuje Obrázek 3.2.



Obrázek 3.2: Počet chyb

Také je sledována závažnost chyb. Zde je používána priorita a vážnost (tzv. severita). Závažná chyba jednoho zákazníka může mít ve výsledku menší prioritu, než méně závažná chyba u tisíce zákazníků.

Druhou možností je sledovat kód, protože ve finální fázi by se měl stabilizovat poměr anotovaných a vymazaných řádků kódu, jak ukazuje Obrázek 3.3.



Obrázek 3.3: Počet anotovaných a smazaných řádek kódu v průběhu projektu

Při sledování těchto signálů je stále operováno nad konečným množstvím feature, které máme pro danou verzi naplánované. Tyto signály nám pomůžou zjistit, jestli se verze stabilizuje, nebo ještě ne.

3.1.5 Znalostní báze

V K je jako znalostní báze a nástroj pro správu dokumentů používán produkt Samepage. Slouží pro rychlé vytváření obsahu, pro komunikaci a pro sdílení firemních dokumentů.

3.1.6 Doporučené úrovně znalostní báze

Znalostní báze by měla mít různé úrovně.

- Nejvyšší úroveň (tzv. elevator pitch) obsahuje v jedné větě hlavní myšlenku produktu a měla by okamžitě zaujmout.
- Na nižší úrovni je popis produktu pro případné zájemce o jeho užívání. Jsou zde popsány hlavní výhody, funkce, čemu se produkt věnuje a porovnání s ostatními produkty.
- Na další úrovni je pak popis konkrétního fungování produktu. Je zde popis základních funkcí produktu i z programátorského hlediska.
- Vývojová úroveň obsahuje informace pro vývojáře a programátory.

Správně postavená znalostní báze umožňuje uchovávat vizi produktu jinde, než pouze v lidských zdrojích.

3.2 AIMTEC

3.2.1 Produkt

Firma AIMTEC má jeden hlavní produkt, který stále vyvíjí. Produkt je však upravován na přání zákazníka. Pro každého zákazníka tedy může být vytvořena specifická verze produktu podle jeho přání.

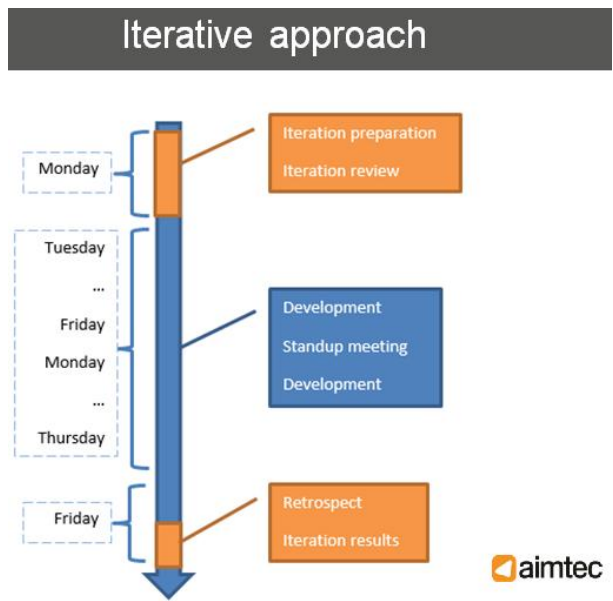
Vlastník produktu (tzv. product owner) dohlíží na to, aby v rámci práce pro zákazníky docházelo k úpravám, které je pak možné do celkového produktu zapracovat.

Jakoukoliv dodávku produktu je možné považovat za jednotlivý projekt. Rozsáhlé vývojové úpravy produktu na základě přání zákazníka jsou prováděny výjimečně (2-3 projekty ročně z 20-50 projektů). Každý projekt obsahuje celkovou implementaci a konfiguraci standardního produktu, včetně instalace a přípravy serverů, wifi infrastruktury, mobilních čteček, prvotní tisk etiket a další.

Práce na těchto projektech probíhají iterativním postupem. Na straně zákazníka i na straně AIMTECu je stanoven tým, který na projektu pracuje. V první fázi nového projektu celý tým AIMTECu navštíví zákazníka a upřesní jeho přání. Z této schůzky vznikne analýza projektu. Tato analýza je považována za nejdůležitější část projektu, protože v případě, že je cokoli specifikováno špatně, nemůže být zákazník nikdy spokojen.

Celá analýza projektu je pak předána do firmy. Zde se sejde tým vývojářů, kterým byl projekt přidělen. Všechny požadavky analýzy jsou odshora dolů rozkládány na story a jednotlivé úkoly. Na této schůzce má každý člen týmu možnost vnášet do projektu vlastní myšlenky a pohledy na práci. Lze zde ovlivnit celý průběh projektu. Po té je vše specifikováno tak, aby k určitým vstupům vznikaly specifické výstupy.

Firma používá iterativní způsob vývoje po čtrnácti dnech, jak ukazuje Obrázek 3.4. Během práce na projektu, jsou plněny jednotlivé úkoly. Na konci období proběhne schůzka týmu, kde je zjišťováno, jestli tým splnil požadavky zákazníka a jak dobře je splnil.

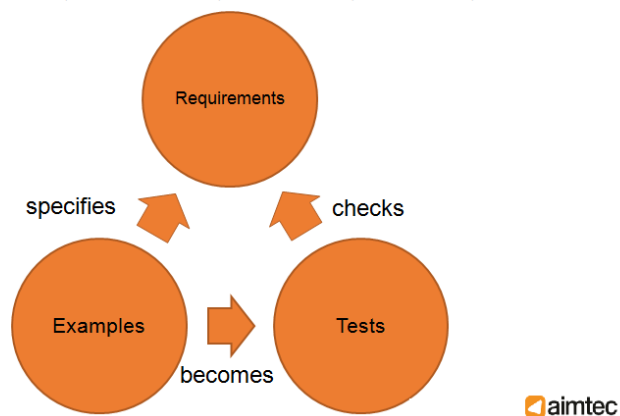


Obrázek 3.4: Iterativní vývoj v AIMTECu

Úkolem celé práce je splnit cíle projektu a vytvořit to, co zákazník potřebuje. Dosažení cílů projektu se měří podle spokojenosti zákazníka. Splnění cílů určí projektový manažer na základě svého dobrého pocitu a komunikace se zákazníkem.

Projekt je považovaný za úspěšně ukončený, pokud jsou splněny cíle projektu a zákazník je spokojen. Cíle projektu jsou ve většině případů velice konkrétní a měřitelné (např. ušetřit peníze, zlepšit logistiku skladu o 10 %, vyhodit 5% zaměstnanců...). Cíle firma určuje podle SMART, který ukazuje Obrázek 3.5. SMART byl vysvětlen v kapitole 2.2.1.

SMART (**S**pecific, **M**easurable, **A**ttainable, **R**elevant, **T**ime-bound)



Obrázek 3.5: SMART

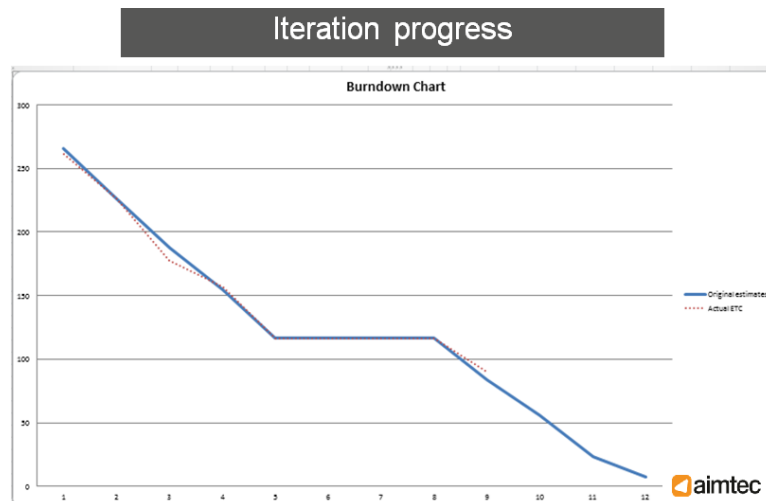
Naznačené šipky v obrázku 3.6 představují praktiku vytváření akceptačních scénářů pro plnění jednotlivých úkolů od zákazníka ke konzultantovi a programátorovi (tzv. ATTD, tedy acceptance test driven development. Konkrétní příklady (tzv. examples z obrázku) se pak stávají kostrou automatických testů.

Podnikovou metodikou pro řízení projektů je v AIMTECU upravený Scrum. Ve firmě probíhají denní rychlé schůzky (daily standup meeting). Dále jsou používány tabule s papírky, na kterých jsou natisknuté úkoly, jak ukazuje Obrázek 3.6. Další úkoly jsou pak zobrazené na velké obrazovce na lístečky. Ve firmě je používána vlastní speciální tabulka v programu Microsoft Excel, pomocí které jsou počítány veškeré zdroje projektu. Zdroje nesmí přesáhnout konkrétní číslo, které je v této tabulce. Je zde vidět také průběh projektů i jejich kód a kolik zabírají zdrojů (času).



Obrázek 3.6 Tabule s úkoly

Firma vytváří tzv. burndown diagramy, které ukazuje Obrázek 3.7: Burndown diagram Jsou dobré pro zobrazení spotřeby finančních zdrojů projektu. Tyto diagramy ale bohužel nemohou zaznamenat případ, kdy projekt naroste, protože v tomto diagramu se pouze posune čára. Proto se ve firmě používá ještě rozsahový diagram (tzv. scope diagram).



Obrázek 3.7: Burndown diagram

3.2.2 Mechanismy a nástroje, bez kterých se nelze obejít

Software pro organizaci a řízení práce, firemních procesů a projektů

Z historických důvodů používá firma k zadávání a práci s úkoly vlastní interní systém pro správu úkolů.

Milníky projektu

Každý projekt má stanovené milníky. Tyto milníky vycházejí z metodiky RUP popsané v kapitole 2.4.4. Vnitropodnikové milníky, které má firma stanovené, jsou analýza, implementace, prototypování, testování u zákazníka a nasazení u zákazníka. Každý projekt je vázán jednotlivými vnitřními pravidly. V průběhu projektu je dosahováno milníků a při jejich dosažení je požadována zpětná vazba od zákazníka. Po dosažení každého milníku jsou se zákazníkem podepsány předávací protokoly. Zákazník celou dobu kontroluje práci firmy a má za úkol si vše vyzkoušet a otestovat. Zdroje, čas i náklady na projekt jsou předem odhadnuty v analýze. Pracovníci jsou projektu přidělováni podle potřeb.

Dodržování metodiky projektu kontroluje auditor. Celá kontrola kvality je zafixovaná pomocí norem. Pokud se jedná o větší projekt, je možné normy přizpůsobit potřebám zákazníka.

Omezení a rizika

Omezení projektu se vztahují vždy ke konkrétnímu projektu a vždy plynou z analýzy projektu. Hlavním kritériem úspěšnosti je vždy spokojenost zákazníka.

Rizika projektu jsou rozlišována podle dvou kritérií a to pravděpodobnosti, že riziko nastane a závažnost rizika. Vznikají tedy čtyři kategorie rizik (malá pravděpodobnost a malý význam, malá pravděpodobnost a velký význam, velká pravděpodobnost a malý význam, velká pravděpodobnost a velký význam).

Rizika, která kombinují vysokou pravděpodobnost, že nastanou a zároveň je jejich význam pro projekt velký, jsou pečlivě hlídána manažerem projektu. Ten musí sám odhadnout nebezpečnost rizika a podniknout včas příslušná opatření. Nejčastějšími takovými riziky jsou například špatná, nebo žádná komunikace zákazníka, nekompetentní lidé v týmu zákazníka, zákazník nechce udělat rozhodnutí a další. Tato rizika jsou pro každý projekt specifická. Mechanismus, díky kterému tým rizikům přechází, je neustálá zpětná vazba od zákazníka. Projektový manažer také neustále kontroluje plnění plánu projektu. Další kontrolou stavu projektu je dosahování jednotlivých milníků. Také v rámci týmu pravidelně probíhají hodnocení jednotlivých iterací (tzv. retrospektivy). Pro kontrolu omezení a rizik firma nepoužívá konkrétní systém, vše probíhá právě zpětnou vazbou.

Kontrola celé organizace

O sestavování hlášení a informační systém firmy se stará jeden vyčleněný pracovník. Vyšší vedení pak pravidelně pořádá schůzky, na kterých jsou probírány jednotlivé položky z reportů, obzvláště pak ty, které vykazují neočekávané hodnoty (náklady, čas a další).

Jednou za měsíc probíhají kontroly projektových manažerů, v průběhu kterých se projektoví manažeři zodpovídají vyššímu vedení. Vysvětlují, jak jejich projekty fungují, co se v nich děje a jak probíhají. Tímto způsobem je ve firmě prováděna kontrola shora, zpětná vazba od pracovníků pak probíhá zdola nahoru. Kontroly tedy probíhají napříč celou organizací. Programátora kontroluje konzultant, konzultanta projektový manažer, projektového manažera jednou za čas vyšší vedení. Vše je také kontrolováno zákazníkem, protože zákazník na průběh projektu reaguje po celou dobu jeho průběhu

Testování

Ve firmě je používáno několik druhů testů. Jednotkové testy (tzv. unit testy) pro otestování jednotek kódu.

Dále jsou používány modulární testy, pomocí kterých jsou testovány jednotlivé ucelené části kódu (moduly). Pomocí tohoto testování je zjišťováno, zda pro určitý vstup existuje i odpovídající výstup.

Akceptační testy probíhají ve firmě u zákazníka, když zákazník produkt schvaluje. U zákazníka jsou také používány testovací scénáře, které zákazník zkouší. Pokud některé testovací scénáře skončí neúspěšně, je u zákazníka vyvolána debata o tom, jak moc je konkrétní testovací případ důležitý a podle jeho důležitosti je pak případně produkt upraven.

3.2.3 Rozdělení týmů a lidské zdroje

Firma rozeznává několik druhů pracovníků:

- projektový manažer
- konzultant
- programátor
- dokumentarista
- inženýr kvality (tzv. QA inženýr)
- obchodník.

Každému projektu ve firmě jsou přidělovány zdroje, podle jeho náročnosti. Projektoví manažeři přidělují zdroje podle vytiženosti jednotlivých pracovníků. Jeden pracovník tedy může pracovat na více projektech. V týmech jsou role pracovníků většinou zdvojené. Každý pracovník by tak měl být v případě výpadku nahraditelný. Například v případě konzultanta tedy jeden konzultant na projektu skutečně pracuje a druhý konzultant projekt sleduje, aby v případě výpadku mohl stávajícího konzultanta nahradit.

Na projektu nikdy nepracuje sám pracovník, který nemá dostatečné zkušenosti z předchozích projektů. Když se zaučuje nový pracovník, nejprve pouze pomáhá některému ze zkušenějších pracovníků. Samostatným programátorem, nebo konzultantem se může stát až ve chvíli, kdy se práci na několika předcházejících projektech naučil.

Ve firmě je snaha přidělovat projektu konzultanta tak, aby již měl z minulosti zkušenosti s podobným projektem. Pokud takový konzultant není dostupný, projekt je přidělen méně zkušenému konzultantovi, ale jiný zkušený konzultant musí být v tomto případě v roli mentora.

Běžné složení týmu je jeden projektový manažer, jeden programátor a jeden konzultant. Tyto pozice jsou hlavní a velice často jsou zdvojené, aby byla zaručena jejich nahraditelnost.

Zadavatel projektu (zákazník) musí také stanovit svůj tým. V tomto týmu má být zákaznický projektový manažer, který ručí za celý projekt a klíčoví zákazníci (testeři zákazníci). Ti musí v každé předávací fázi produkt otestovat a vyzkoušet, jestli odpovídá potřebám jejich firmy. Firma má přesně specifikované požadavky na spolupráci se zákazníkem, např. na schůzky je nutné mít k dispozici místnost s internetem, je bezpodmínečně nutné mít k dispozici zákaznický tým a další.

Komunikace se zákazníkem probíhá iterativním postupem. Jedna iterace trvá většinou čtrnáct dní. Problémy v komunikaci se zákazníkem musí projektový manažer najít a odstranit včas. Nejčastěji problémy vznikají, pokud jsou v týmu lidé, kteří nekomunikují. Další problémy také mohou vznikat v případě rozhodování, pokud zákazníci v týmu nemají dostatečné kompetence udělat rozhodnutí. Také se může stát, rozhodnutí neumí, nebo nechtějí udělat. V těchto případech je manažerem AIMTECu kontaktováno vedení zákaznické společnosti.

Odpovědnost je ve společnosti delegována dolů napříč celou společností. Za celý projekt zodpovídá projektový manažer, za jednotlivé části pak konzultant a za jednotlivé konkrétní části kódu zodpovídá programátor.

Pokud dojde ke sporům ve vývojovém týmu, je na projektovém manažerovi, aby situaci uklidnil. Spory na straně zákazníka je nutné řešit se zákaznickým projektovým manažerem. V případě neúspěchu je nutné spor neprodleně delegovat výš. V případě neúspěchu by bylo možné změnit pracující tým, ale s tímto případem se AIMTEC ještě nesetkal.

Komunikace ve firmě

Komunikace v rámci firmy probíhá e-mailem, nebo si členové firmy volají. Firma sídlí ve třech domech jedné ulice, ke komunikaci jsou tak často využívány také osobní návštěvy, nebo se členové společnosti mezi sebou zvou na schůzky do zasedacích místností. Speciální nástroje pro komunikaci firma nepoužívá.

3.2.4 Verze k vydání

Firma používá systém pro správu verzí Git. Z hlavní vývojové větve Master je ve vhodné fázi vývoje oddělena produkční větev pro vydání příslušné verze. Na produkční větvi jsou opraveny chyby, případně jsou provedeny modifikace na přání zákazníka.

Produktový manažer firmy hlídá celý produkt a snaží se požadavky zákazníků směřovat tak, aby šly ke stejnému řešení. Je zde snaha všechny modifikace na přání zákazníka integrovat zpět do celkové verze produktu. V případě, že není možné přidat konkrétní modifikaci do celkové verze, produktový manažer rozhodne o tom, jak vyřešit konflikty. Speciální úprava pro zákazníka se tak většinou promítne v celém produktu.

3.2.5 Znalostní báze

Obecný vývoj projektu je dokumentován na wiki projektu, dokumentace a konkrétní informace jsou pak uloženy v systému pro řízení vztahů se zákazníky (tzv. Customer Relationship Management, také CRM systém). Firma pro řízení vztahů se zákazníky používá systém společnosti Microsoft, Microsoft Dynamics. K projektům je také ukládána kompletní e-mailová komunikace. V případě, že jsou informace se zákazníkem upřesňovány telefonickým hovorem, dělá se vždy zápis hovoru, aby bylo vždy možné dohledat všechny domluvené informace. V případě osobních schůzek je vždy vytvořen zápis z jednání. V CRM systému je pak možné najít kompletně celou komunikaci a všechnu dokumentaci k projektu.

3.3 CCA

3.3.1 Produkt

Firma CCA se nezabývá jedním pilotním produktem, ale vytváří samostatné produkty na přání zákazníků. V této firmě byly získávány informace o jednom z konkrétních projektů, na kterém firma pracuje.

Tento systém slouží především ke správě a evidenci požadavků. Firma pomocí tohoto systému řídí a eviduje projekty od jejich počátku až po jejich ukončení. Systém také obsahuje pomoc s problémy zákazníků (tzv. helpdesk). Při práci na projektu musí být v systému vždy možné dohledat, ke kterým úpravám kdy došlo a proč. Je zde možné zpětně dohledat všechna hlášení (tzv. issue).

Tento projekt je z roku 1997. Jedná se o interní projekt, který firma používá sama pro sebe. Tento projekt se liší od definice projektu tím, že nemá konec. Je stále živý a stále dochází k jeho úpravám. Projekt má naplánované milníky a probíhá v procesním režimu. Má svůj rozpočet (tzv. budget), který čerpá. Rozpočet lze v rámci projektu čerpat na klasické opravy, servis a rozvoj. Pokud rozpočet dojde, je nutné žádat vedení společnosti o nový. O navýšení rozpočtu je s vedením společnosti možné smlouvat podle toho, co se od projektu očekává. Projekt se rozvíjí z potřeb běžných uživatelů, kteří ho používají při své práci. Všichni uživatelé mají možnost zadávat své požadavky na vylepšení systému. Zásadní požadavky na změnu mohou přijít přímo od vedení.

V projektu se plánuje po zhruba čtyřměsíčních cyklech. Realizační tým a ředitelé úseků schvalují a plánují jednotlivé požadavky, které budou v rámci projektu vytvářeny. Schválený balík požadavků je pak poslán do distribuce.

Systém je vyvíjen inkrementálně. Skládá se ze sady programů a sady závislostí. Díky tomu je možné provádět změny rychle a je možné je rychle nasadit.

Hlavním kritériem úspěšnosti projektu je spokojenost zaměstnanců, kteří systém používají. Systém obsahuje propracovaný systém hlášení o tom, jak jsou zvládnuty požadavky zákazníků. Průměrně jsou zde hlášeny dvě vady měsíčně.

3.3.2 Mechanismy a nástroje, bez kterých se nelze obejít

Automatická kompilace a automatické testování

V projektu probíhají automatické kompilace i automatické testování. Při nahrávání nového kódu programátorem je vše automaticky zkompileováno. Pokud z nějakého důvodu commit neprojde, je důležité, aby byla odeslána zpráva například vedoucímu projektu.

Kontrolní seznamy

Ve firmě je část kontroly práce prováděna pomocí kontrolních seznamů, (tzv. checklist). Každý krok, který je v projektu proveden, podléhá konkrétní akceptaci. Je zde sada kontrolních otázek, podle kterých se akceptace provádí (např. proč je činnost prováděna, jak byla prováděna a další). Náročnost kontrolního seznamu otázek je upravená podle časové náročnosti úkolů. Čím delší čas na zpracování úkolu si pracovník uvede, tím delší kontrolní seznam pak musí vyplnit. Pro drobné úpravy je kontrolní seznam nepovinný. Tímto způsobem je pracovník hodnocen a vedoucí projektu má díky těmto seznamům k dispozici informace o průběhu práce. Každý modul musí mít jasný cíl a úkol. Akceptace pomáhá testovat i kvalitu produktu a meziproduktů. Kontrolní seznam je formou převzetí odpovědnosti za danou část projektu.

Rizika a omezení projektu

Rizika projektu jsou hodnocena na stupnici od jedné do pěti, podle závažnosti rizika. Pro závažnější rizika je vypracován rizikový plán. Rizika jsou hlídána přímo v šablonách kontrolních seznamů firmy. Součástí kontrolních seznamů jsou tabulky, ve kterých se řeší dopad rizika a pravděpodobnost, že nastane. Ze součinu je zjištěna závažnost rizika. Díky kontrolním seznamům si každý může uvědomit rizika projektu a v případě, že jsou vysoká, je možné rovnou stanovit opatření.

Omezení projektu jsou dána zákazníkem. Při konzultaci se zákazníkem je rozhodováno, co je pro zákazníka akceptovatelné rizik.

Všechny náklady projektu jsou vykazovány. Je nutné vyčíslit pracnost, termíny (tzv. on time), plnění termínů, limity, produktivitu a cenu (tzv. on budget). Produktivita se vyhodnocuje po produktech i po střediscích. Kvalita je hlídána zpětnou vazbou.

Dalšími omezeními projektu jsou náklady a plnění termínů. Dále je sledována spokojenost zákazníka. Část systému se zabývá zákaznickou spokojeností. Na všechny požadavky, které do systému zákazníci vložili, musí vždy existovat zpětná vazba. Systém se snaží nutit zákazníky, aby vyjádřili svůj názor na splnění jejich požadavků (tzv. issue).

Postup práce realizačního týmu

Realizační tým si nejprve určí všechny činnosti, které vedou ke splnění požadavků zákazníků. Také jsou určeny povinné výstupy a způsob testování produktu. Každé issue musí projít analýzou, ověřením výstupů a finálním otestováním. Podle typu projektu lze postup optimalizovat. Projektový analytik přímo u zákazníka připraví návrh v nástroji Enterprise Architekt. Po odsouhlasení návrhu zákazníkem vytvoří projektový analytik zadání pro programátora a současně testový analytik vytváří návrhy testování. Testování je vytvářeno už v této fázi. V průběhu práce programátoři programují požadovanou aplikaci a analytik je v kontaktu se zákazníkem, u kterého ověřuje, zda jsou splněny jeho požadavky. Po splnění požadavků je aplikace otestována a předána zákazníkovi.

U projektů je nutné, aby zákazník věděl, co chce. Analýza zákaznických požadavků obsahuje kapitolu součinnost, která uvádí, co společnost od zákazníka potřebuje k realizaci projektu. Pro potvrzení zákaznických požadavků je zde akt validace požadavků, kdy zákazník potvrzuje své požadavky. V průběhu vývoje probíhají se zákazníkem schůzky, na kterých je seznamován s prototypy produktu v průběhu realizace. Díky těmto schůzkám má zákazník od počátku vývoje představu o tom, jaký produkt se pro něj vyvíjí.

Po předání projektu zákazníkovi funguje v CCA běžný zákaznický servis. O zákaznický servis se starají konzultanti. Konzultanti jsou zároveň i testeři a vytvářejí uživatelské příručky.

Plánování

Plánování v projektech má na starosti vedoucí projektu, který na základě požadavků rozděluje práci. Často má k dispozici hlavního programátora a hlavního konzultanta. Těm může delegovat větší části práce, které si rozdělí dle potřeby. Úkoly jsou plánovány cyklicky a v mezifázích dochází k vydání produktu. S přibližujícím termínem vydání se přechází do řízení Scrumem. V běžném vývoji Scrum není používán. Většina projektů firmy je nyní v servisním režimu.

Testování

CCA má pro testování vlastní tým. Jeho členové provádějí zejména uživatelské funkční testování. Mají k dispozici nástroj, ve kterém jsou schopni připravit si vlastní testování. CCA má pro evidenci testů vlastní testovací systém s evidencí jednotlivých testů. Evidence ukládá informace o tom, které části aplikace byly testovány, kdo je testoval a jak testy dopadly. Ve firmě je snaha psát k produktům automatické testy, ale ty jsou časově náročné a neřeší všechny problémy.

Testy vznikají ze sady požadavků z analýzy, které jsou rozpracovány. Rozpracováním vznikne dokumentace, programátorská část a příslušné testy. V CCA je rovnou při návrhu aplikace specifikováno, jak bude aplikace testována.

3.3.3 Rozdělení týmů a lidské zdroje

Odpovědnost za projekt se dělí podle rolí. Vedoucí je zodpovědný za celý projekt, programátor za program, tester za otestování, konzultant za dokumentaci. Za jednotlivé části zodpovídá jeden podřízený, za celek vedoucí projektu. V projektech musí být jasné, kdo zodpovídá za kterou část projektu. Vedoucí projektu má velkou volnost, ale finančně je sledovaný, aby byl ve správných číslech.

Lidé ve firmě komunikují telefonickými hovory, případně používají Skype. Dále svolávají osobní schůzky. Nejčastěji probíhá komunikace pomocí e-mailů. Interní systém firmy je schopný e-maily automaticky zpracovat a vytvořit z nich úkoly. Emaily, které systém nedokáže zpracovat, přidělují ručně pracovníci oddělení hotline. Pokud se jedná o komunikaci k projektu, je snaha, aby komunikace vždy probíhala přes interní systém firmy.

V případě volání zákazníka do firmy je komunikace po telefonu řízená vnitřním systémem, který je napojený na ústřednu. Ústředna přeměruje volajícího na kompetentního člověka z firmy. Komunikace ve firmě nemá předepsanou podobu, pouze všechny informace o projektu musí být zaneseny do interního systému firmy.

Dalším způsobem komunikace je předávání úkolů mezi pracovníky. Při předávání úkolů v projektu může vzniknout problém předávání úkolů mezi pracovníky stále dokola. Pokud zde nastane rozpor, rozhoduje o přidělení úkolu vedoucí projektu.

Ve firmě probíhají porady, kde se řeší případné spory. Většinu sporů si lidé vyřeší sami, případně s vedoucím projektu. V případě dalších problémů je delegují výš.

Přijímání nových členů do týmu

Školení při nástupu do projektu nemá smysl, protože nový člen týmu ještě nezná potřebné souvislosti. Ideální doba pro první školení přibližně po půl roce. Ideálně nový člen týmu začne rovnou pracovat na reálných problémech a dostane k ruce patrona, který mu radí. Patron je odměněn podle toho, jak rychle začne nováček pracovat na zákaznických projektech sám. Nejdůležitější je zde motivace. Více o motivaci v kapitole 4.6.

3.3.4 Verze k vydání

K vydání nové verze interního systému CCA většinou dochází v pátek večer a proběhne do soboty. V této době systém nefunguje a dojde k přírůstkové změně. Není vydávána každá drobná změna. Pokud je požadovaná změna velká, systém se na dočasnou dobu vypne a změna proběhne. Firma používá systém pro správu verzí GIT a v tomto projektu v případě verze k vydání oddělí od hlavní vývojové větve Master větev Release, která je po stabilizaci nahrána během pátečních změn.

3.3.5 Znalostní báze

Ve firmě je používána znalostní bázi trojího typu. Prvním typem je dokumentace vedoucího projektu (dokumentace řídicí). Dokumentace řídicí zahrnuje lidi (zdroje), sledování zdrojů, plánování, smlouvy, fakturace a administrativu spojenou s projektem. Firma tuto dokumentaci vede téměř výhradně v nástroji pro správu informací Sharepoint. Zde má dokumentace konkrétní členění. Každý projekt tu má svůj prostor. Sharepoint je propojený s interním systémem firmy.

Další dokumentací je dokumentace o produktu. Informace o produktu jsou velice cenné. U nového produktu jednoduše vznikne sada požadavků. Problém vzniká v případě změny staré dokumentace. Úprava staré dokumentace je pracná a poměrně drahá. Je mnohem snadnější popsat změnu, než cílový stav. Popis změn bez popisu cílového stavu vede k celkové nepřehlednosti dokumentace. Je pak velice těžké dohledat potřebné informace. Je tedy důležité udržovat dokumentaci aktuální a je nutné hlídat, aby se změnou aplikace proběhla vždy i změna dokumentace. Pro kontrolu dokumentace je zde zavedena sada kontrol, kdy je odpovědnost za dokumentaci delegována na pracovníky, kteří na projektu pracují. Programátor v průběhu implementace podle příslušné dokumentace vlastně kontroluje, že jde dokumentace

správná. Zároveň tím přebírá odpovědnost za změnu dokumentace v případě, že nesedí, protože všechny změny produktu musí být programátorem zdokumentované.

Třetí úroveň dokumentace je wiki firmy, do které se dávají výhradně obecné pravdy jako konvence, pravidla vývoje, triky, případy užití, a další, které nejsou evidovány přímo v projektu. O zastaralých článcích ve wiki informují sami uživatelé a podle toho jsou články aktualizovány.

V rámci dokumentace má firma připraveno asi 50 šablon, které je nutné dodržovat. Například návrhářské šablony, datové modely, klasické zadání, uvedení do provozu, šablony pro řízení, šablony pro analýzu a další. Díky šablonám je možné dosáhnout jednotné struktury projektové dokumentace a uživatel tak nemusí řešit strukturu dokumentu, ale může se soustředit na jeho obsah. Šablony vycházejí ze specifikace RUP, přímo z jeho artefaktů.

Za určitou část dokumentace, nebo projektu vždy zodpovídá konkrétní pracovník. Například v případě testování přebírá tester odpovědnost za testy, o kterých prohlásil, že v pořádku procházejí. Každý má odpovědnost za svou část práce. Pokud je někým akceptován dílčí výsledek (dokument, program, test...), stává za tento výsledek spoluodpovědný.

Doporučení pro zadávání prací v projektu Space Traffic formou výstupu

V závěru konzultací bylo doporučeno zadávat semestrální práce v projektu Space Traffic formou výstupu, tedy očekávaným výsledkem práce. Díky zadání výstupem je možné lépe dohlížet na splnění očekávané funkcionality. (Tento způsob zadávání byl ověřen na projektu Space Traffic v letním semestru 2016 a měl nesporné výhody oproti standardnímu bodovému zadání – více v kapitole 5.15)

3.4 Unicorn Systems

3.4.1 Produkt

Firma Unicorn Systems nevytváří jeden pilotní produkt, ale vytváří produkty na přání zákazníků. Ve firmě byl analyzován jeden konkrétní produkt. Největší část konzultace byla věnována tomuto produktu.

Unicorn Systems je dodavatelem obchodní platformy DAMAS pro elektroenergetický trh v Irsku a Severním Irsku (EirGrid). Účelem analyzovaného projektu je automatizace manuálních procesů společnosti EirGrid, která je dlouhodobým partnerem společnosti. Cílem je zefektivnit její práci a minimalizovat rizika zanesení manuální chyby. Cíl projektu bude naplněn rozšířením existující obchodní platformy DAMAS. Hlavním cílem projektu (tzv. target) je zvýšit prodej, hlavním očekávaným výsledkem (tzv. goal) je takové řešení, které naplňuje target, např. vytvoření materiálu.

Studie proveditelnosti

Studie proveditelnosti byla provedena přímo společností EirGrid, protože se jedná o přidání funkcionality do již existujícího produktu. Finanční plán realizace projektu byl nastaven na nulový peněžní tok (tzv. cashflow). Tedy tak, aby realizace Unicorn Systems v průběhu projektu negenerovala zisky ani ztráty. Studie proveditelnosti ze strany Unicorn Systems nebyla v tomto případě nutná.

Zdroje produktu:

- odpracované hodiny: 2 420 člověkohodin
- realizované náklady: 2 707 063 EUR
- termíny projektu

Milestone	Duration [weeks]	Plan		Reality		Status
		Start	End	Start	End	
Project Initialization	1	8-Jun	12-Jun	8-Jun	12-Jun	✔
Project Start	0		15-Jun		15-Jun	✔
Project Kick-Off	<1		Week 25		Week 25	✔
Engineering Project	7	15-Jun	31-Jul	15-Jun	26-Jul	✔
Development	8	27-Jul	18-Sep	22-Jul	16-Sep	✔
FAT	7	17-Aug	02-Oct	13-Aug	02-Oct	✔
SAT	4	05-Oct	30-Oct	05-Oct	22-Jan-16	✔
Unicorn Production to SLS Handover	1	26-Oct	30-Oct	20-Oct	07-Dec	✔
Contingency (will be used for SAT)	2	02-Nov	13-Nov	-	-	✔
Go-Live & Stabilization	3	16-Nov	04-Dec	25-Jan-16	22-Feb-16	✔
Go-Live	<1		18-Nov		25-Jan-16	✔
Stabilization	3	16-Nov	04-Dec	25-Jan-16	22-Feb-16	✔
Documentation Update	1	07-Dec	11-Dec	18-Jan-16	18-Jan-16	✔
Project End	0		11-Dec		22-Feb-16	✔

Legend: 🟡 pending 🟢 in progress ✔ done 🟠 attention (risk, issue, change foreseen)

UNICORN Systems

Obrázek 3.8: Termíny projektu

Z časového plánu projektu, který ukazuje Obrázek 3.8 je patrné, že projekt byl v únoru roku 2016 ukončen.

3.4.2 Mechanismy a nástroje, bez kterých se nelze obejít

Autobuild a autotest

Kvalita mezivýstupů u produktů je hlídána nástroji pro automatizované nasazování, automatickými testy a statickou kontrolou kódu.

Tyto nástroje průběžně validují výstupy vývojářů a podávají zprávy softwarovému architektovi projektu, který v případě problémů zajistí nápravu, a v případě závažnějších problémů informuje projektového manažera.

Fáze zmražení projektu

Ve firmě jsou známy případy, kdy kompletně celý tým včetně projektového manažera projekt opustí a na projekt je pak navázáno v průběhu dalších let novým týmem. Pokud je v projektu očekáváno, že dojde ke kompletní výměně týmu, je nutné velice dobře zvládnout fázi zmražení projektu (tzv. freeze). Projekt je nutné silně dokumentovat a veškerá implementace musí probíhat přesně podle dokumentace. Sice je pak nutné očekávat pomalejší vývoj, ale pokud je pak práce týmu kdykoli v průběhu práce přerušena, je na ní díky silné dokumentaci možné navázat. V dokumentaci se pouze označí, kde skončil v plánované implementaci původní tým a nový tým pak v projektu pokračuje podle dokumentace. V tomto systému nesmí existovat kód bez dokumentace. Také je v tomto případě doporučováno psát nejdříve testy a teprve potom se doplňovat

kód, který vyhovuje již hotovým testům. Také je nutné nepodcenit fázi plánování, která musí být přehledně a detailně zpracovaná. Projekt by měl být celý předem naplánován a rozdělen až na jednotlivé úkoly. Také je nutné projektu vytvořit množinu úkolů a malých podprojektů, které je nutné udělat (tzv. backlog), kde každý z těchto podprojektů musí mít vlastní dokumentaci tak, aby kdykoli se práce přeruší, bylo možné na ní navázat. Tento způsob se nazývá nejprve dokumentace (tzv. documentation first).

Plánování

Celkový projekt se rozpadá na úkoly (tzv. task), příběhy (tzv. story) a celkový příběh (tzv. epic story) Pro plánování projektů je primárně používán program Microsoft Excel, lze v něm řídit projekt do přibližně 2 milionů EUR. Pro složitější projekty je používán nástroj JIRA. Celý projekt z pohledu termínů, kapacity a financí je hlídán projektovým manažerem. Výběr způsobu plánování je zcela v kompetenci projektového manažera. Firma vyžaduje pouze hlášení o průběhu projektu (tzv. repot) a všechny důležité dokumenty o projektu musí být uloženy v systému firmy pro správu informací Unicorn Universe.

Zálohování

Firma má vlastní divizi, která se stará o servery firmy a zálohování. Další divize je vyčleněna pro datový sklad (tzv. data warehouse).

Studie proveditelnosti

Cílem studie proveditelnosti v Unicorn Systems je popsat vlastními slovy požadavky klienta. Dále zde probíhá mapování okolí, zda se požadavky klienta shodují s tím, co už používá. Mapování okolí probíhá vždy procházením dlouhodobého plánování a dlouhodobé strategie na 5-10 let, aby vše zapadlo do celkového rámce. Už v této fázi je stanoven rozpočet, ze kterého je ubíráno už při analýze. Pokud rozpočet dojde, práce jsou pozastaveny. Tento způsob řešení je prosazován hlavně u nových partnerů, u dlouhodobých partnerů nejsou tyto mechanismy nutné z důvodu důvěry.

Podniková metodika

Unicorn Systems dodává projekty s využitím interní metodiky Unicorn Universe Process. Jedná se o sběr nejlepších praktik metodik projektového řízení Prince2, certifikací projektového manažera IPMA, metodiky RUP, postupy a koncepty ITIL, popis funkcí IDEF0 a jejich implementaci do reálného prostředí. Výhodou tohoto přístupu je schopnost adaptovat se na interní metodiku klienta a vést dialog v jednotném

jazyku. Je zde definována sada seznamů rizik (tzv. risk list), plánování, vyhodnocování, plánování lidských zdrojů, komunikace s lidmi a další. Jsou zde definovány rozsáhlé sady pravidel pro řízení celých divizí. Vše je pouze doporučované, jsou zde příklady použití nástrojů a popis jejich výhod. Projektový manažer má při řízení projektu svobodnou volbu. Měl by však dodržovat základní pravidla kvůli kontrolování. Pokud totiž vybočí ze zažitých pravidel, není možné na jeho projekt uplatnit některé kontroly. K projektovému trojimperativu, podrobněji popsáném v kapitole 2.2.1, se zde přidává čtvrtý rozměr, který se dělí na scope a kvalitu. Projekt v Unicorn Systems má různé úrovně kompetencí a kontroly probíhají na základě výsledků, které jsou pečlivě kontrolovány.

Měření projektu

Vyhodnocování projektu je prováděno na týdenní bázi. Realizační tým sepisuje průběh práce (tzv. report), který předává svému projektovému manažerovi. Projektový manažer je na základě reportů schopen určit stav projektu. Každý týden probíhají schůzky, kde jsou reporty vyhodnocovány. Na jejich základě lze určit zbývající množství práce (kolik pracovník odpracoval, kolik měl odpracovat a kolik práce mu zbývá). Díky reportům lze také signalizovat překročení zdrojů (času, nákladů) včas. Pokud v projektu dojde k problémům, reporty včas upozorní vedení, že má projekt problémy. V době jeho ukončení tak jeho neúspěch není překvapením. Plnění plánu se hodnotí z pohledu následujících perspektiv, které jsou vázané na čas a pracnost:

- kolik balíků požadavků mělo být v daný čas implementováno a kolik z toho je implementováno (časová perspektiva)
- kolik hodin mělo být odpracováno, kolik hodin je odpracováno a kolik zbývá odpracovat (perspektiva pracnosti)

Obě dvě perspektivy jsou poté transformované do nákladů.

Pro řízení projektu a řízení jeho kvality, kvantity, termínů a rozpočtu (tzv. KPI) má firma šablonu přímo v programu Microsoft Excel. Naplnění KPI se také považuje za kritérium úspěšného ukončení projektu. Při odhlédnutí od KPI je za úspěšný projekt považován každý projekt, ke kterému klient podepsal pozitivní referenci, tedy byl spokojen.

Zpětná vazba je tedy poskytována také klientem. O měření naplnění cílů projektu se ve valné většině projektu stará klient, nebo jakožto poskytovatel služeb a řešení firma poskytuje nástroj pro naplnění klientových cílů. Kvalita zabere přibližně 1/3 projektového rozpočtu.

Rizika a omezení projektu

Pro úspěšné ukončení projektu je nutné mít jednoznačné zadání projektu a definovat jeho omezující podmínky projektu (termín, rozpočet, kvalita).

Rizika projektu jsou identifikována průběžně po celý životní cyklus projektu. Rizika prochází standardním procesem risk managementu, tedy identifikace, analýza, ošetření, vyhodnocení a archivace. U rizik hodnotíme dva aspekty, pravděpodobnost, že riziko nastane a jeho dopad. Hodnotíme od nuly do jedné (např. 0.25, 0.5, 1, ...). Vynásobením obou aspektů získáme hodnotu, která definuje závažnost rizika. Tu poté dle závažnosti řadíme do prioritních skupin, které jsou patrné z Obrázku 3.9. Každé riziko se stává podprojektem projektu.



Obrázek 3.9: Rizika

Bezpečnost, kterou musí projekt splňovat, je zde kategorizována podle popsanych pravidel.

3.4.3 Rozdělení týmů a lidské zdroje

Každý projekt má vždy tzv. svatou trojici. Projektový manažer je zodpovědný za celý projekt, Business Architect je zodpovědný za analýzu, Software Architect je zodpovědný za vývoj. Standardní struktura projektových zdrojů je následující:

- Projektový manager (1x)
 - Business Architect (1x)
 - Business Analyst/Team Leader (X)
 - Business Analyst/Business Analyst Junior (Y)
 - Software Architect (1x)
 - Senior Developer/Team Leader (X)
 - Developer/Junior Developer (Y)
 - Test Architect (1x) zodpovědný za testery, nástroje a metodiky testování
 - Tester (X)

Plánování projektů vychází z organizační struktury firmy. Analytické úlohy plánuje Business Architect, vývojové úlohy plánuje Software Architect, testy plánuje Test Architect. Dané role poskytují konzultace a vedení v řešení úloh a revidují jejich správnost.

Dle velikosti projektu se tato struktura může měnit. Role v projektu získávají lidé dle specifického firemního rozdělení, nebo podle vazeb na klienta. Ve věci popisovaného projektu byl extra vybrán člověk s detailní znalostí produktu DAMAS a schopností analýzy a návrhu řešení komplexních matematických úloh.

Výběr členů týmu je v kompetenci projektového manažera. Projektový manažer si sepíše seznam požadovaných pracovníků pro projekt a jejich požadované znalosti. V rámci projektu jsou pak vyhledáváni konkrétní pracovníci, kteří mají požadované znalosti. Je možné do týmu vybrat méně zkušené pracovníky, kteří jsou levnější, ale je pak nutné počítat například s jejich poloviční efektivitou, kdy se pracovník při práci ještě učí. Každý pracovník v Unicorn Systems má svou osobní kartu se svými zkušenostmi a podle nich je vybírán do projektů. Pro nábor nových zaměstnanců má Unicorn Systems vlastní personální agenturu, Unicorn Vigour.

Požadavky na zadavatele projektu nejsou specifikovány, důležité je, aby zadavatel uměl zformulovat své přání. V rámci každého projektu se definují požadavky na součinnost a ty je nutné plnit. Mezi standardní požadavky patří, že klient bude odpovídat na e-maily a hovory a musí se účastnit schůzek o projektu. Pokud například není klient schopen reagovat na e-mail do dvou dnů, musí se s tímto omezením počítat v časovém rozpočtu projektu a dojde k navýšení časových požadavků, čímž dojde i k zvýšení nákladů.

Spory v týmu jsou řešeny ihned. Jedním z pravidel komunikace firmy je: chválíme veřejně, káráme v soukromí.

Standardně jsou vedené následující komunikační kanály:

- denní/dvoudenní báze
 - Stand-up meeting, rychlá schůzka projektového týmu, rychle shrnutí co kdo dělal, co kdo bude dělat a jaká má kdo úskalí
- týdenní báze
 - Projektový status, externí setkání s klientem, projektový manažer prezentuje stav projektu klientovi
 - Interní řídicí rada, projektový manažer prezentuje stav a výhled projektu, nastavují se další kroky pro eliminaci rizik.
 - Interní posouzení statusu, projekty reportují svůj stav nadřízeným.
- měsíční báze
 - Externí řídicí rada, schůzka pro report stavu a směřování projektu na vyšší úrovni, nastavují se další kroky v eliminaci rizik.

Komunikace ve firmě probíhá pomocí e-mailu, telefonických hovorů, osobních schůzek, či přidělováním úkolů.

3.4.4 Verze k vydání

Nejvyšší úroveň projektu je mapa funkcionality (tzv. road map), kterou je nutné dodat. Vlastnosti (tzv. feature) produktu jsou zde členěny do skupin must have a nice to have, ale lze definovat i další úrovně. Všechny jsou seřazené v čase podle termínů. Z této mapy jsou vybírány podle priority. Produkt je vydán, pokud jsou všechny vlastnosti ze skupiny must have implementovány.

Další úroveň je akceptační úroveň. Každá vlastnost má akceptační kritéria, která jsou jasně dána již v přípravné fázi projektu. Akceptačními kritérii definujeme bod, kdy je produkt připraven pro nasazení na produkční prostředí. Jakmile tento bod nastane, obě strany, klient a Unicorn Systems, si potvrdí připravenost pro nasazení na produkci a je provedeno samotné nasazení. Projektový manažer může rozhodnout o nasazení projektu pouze v případě, že byl naplněn jeho obsah a rozsah. V jiném případě musí žádat vyšší úroveň o výjimku. U klienta o nasazení rozhoduje odpovědná osoba, nejčastěji také projektový manažer. Obě dvě kompetentní osoby přirozeně musí zajistit interní schválení dle specifik daného projektu.

3.4.5 Znalostní báze

Unicorn Systems Universe je informační systém vyvinutý Unicorn Systems. Je to systém pro správu dokumentů a řízení všech společností z holdingu (56 společností). V některých situacích společnost využívá dashboard z nástroje JIRA. Dále firma používá:

- Zadávání úkolů.
- Document management systém.
- Reporting.
- Archiving.
- Skype/Jabber.
- Workshopy s klientem.
- Vzdálené prezentace.
- MS Office balík.
- GIT (vlastní server).

Veškeré dokumenty jsou uloženy primárně v systému firmy Unicorn Universe. Je povinností každého pracovníka zde udržovat dokumenty kompletní a aktuální pro kolaboraci a archivaci. Veškerá dokumentace je v Unicorn Universe udržována i po ukončení projektu v archivovaném stavu. Unicorn Universe je znalostní báze pro vyhledávání informací napříč projekty.

3.5 Shrnutí poznatků získaných ve firmách

Z analýz ve firmách vyplynulo několik možností, jak řešit projektové problémy. U spousty navrhovaných řešení jsou však hlavním problémem zdroje. Tato podkapitola je věnována souhrnu hlavních mechanismů, které firmy používají pro úspěšné projektové řízení. Dále zde budou vypsána doporučená řešení pro problémy projektu Space Traffic, pokud byla navržena.

3.5.1 Firma K:

- Redmine, GIT.
- Autobuild, autotest.
- Backlog produktu.
- Signály stabilizace projektu (anotované/nové řádky kódu).
- Bugfixing.
- Úrovně znalostní báze.

3.5.2 AIMTEC

- Rizika projektu (malá pravděpodobnost že nastane, dopad velký atd.).
- Rozsáhlá analýza a plánování.
- Milníky projektu.
- Nástroje pro komunikaci.
- Kontrola zpětnou vazbou zákazníka.
- Měřitelné cíle.
- Delegace odpovědnosti dolů.
- Testování před implementací.

3.5.3 CCA

- Kontrolní seznamy pro všechny práce na projektu (checklist s otázkami).
- Oddělení dokumentace pro vedoucí projektu.
- Více úrovní znalostní báze (wiki obsahuje jen obecné pravdy).
- Kontrola dokumentace (checklist, kontrola aktuálnosti dokumentace, zodpovědnost za dokumentaci musí být delegována na lidi, šablony dokumentace).
- Zadání formou výstupu.
- Testy pro nové členy projektu (školení nemá význam).
- Automatické testy a automatické kompilace.
- Po činnosti commit odeslání kontrolní zprávy o výsledku.
- Kontrola kvality kódu.
- Definice projektových milníků.
- Řízení pomocí metodiky Scrum.
- Motivace studentů.

3.5.4 Unicorn Systems

- Definice fáze freeze projektu.
- Automatická kompilace a automatické testování.
- Kontroly kvality kódu.
- Podrobná dokumentace.
- Backlog projektu.

3.6 Hlavní odpovědi z firem

Následující kapitola je shrnutím odpovědí na hlavní otázky. Odpovědi byly zjišťovány v průběhu schůzek ve firmách.

Pokud je hotová část funkcionality:

- Firma K: začlenění této části do produktu hned, jak je to možné
- AIMTEC: začlenění do produktu pouze v případě, že je to možné
- CCA: začlenění do produktu při nahrání nové verze
- Unicors Systems: podle specifikace konkrétního projektu

Mechanismus, jak překlenout že v projektu mění programátoři (situace v životním cyklu produktu, kdy dochází ke změně lidských zdrojů):

- Firma K: v praxi nenastává, doporučuje řešit znalostní bází
- AIMTEC: v praxi nenastává, doporučuje řešit znalostní bází
- CCA: v praxi nenastává, doporučuje řešit znalostní bází
- Unicorn Systems: nastává výjimečně, řešit fázi freeze projektu, kdy podrobná dokumentace zajistí možnost pokračovat tam, kde předchozí tým skončil (documentation first). Dokumentovat se musí vše včetně komunikace.

Jak firma řeší uvedení nového člena týmu do projektu (co nejrychleji).

- Firma K: nenastává, postup ve firmě je zespoda
- AIMTEC: učí se v rámci projektu pomáháním zkušenějšímu členovi
- CCA: řešení reálných úkolů za pomoci patrona
- Unicorn Systems: učí se v rámci projektu pomáháním zkušenějšímu členovi

3.7 Obecné informace o schůzkách

Každá společnost přistupovala k vedení schůzek jiným způsobem, i přes velkou snahu se nepodařilo dosáhnout jednotnosti získaných informací z firem. U firmy K a AIMTECu byly řešeny obecné problémy projektového řízení a vnitropodnikových metodik. Analýza konkrétního projektu zde neproběhla. V CCA proběhla analýza konkrétního projektu, který ale nemá stanovený konec, z hlediska projektového řízení zde nastává otázka, zda se opravdu jedná o projekt. V Unicorn Systems byl analyzován konkrétní projekt, ale s nesrovnatelnými časovými zdroji.

4. Analýza projektu Space Traffic

Tato kapitola je věnována analýze stavu projektu Space Traffic v době, kdy byl přebrán autorkou práce a Janem Kotalíkem. Bude zde popsán stav v době jeho přebírání. Také zde budou popsány problémy a situace, které v této fázi projektu nastaly. Aby byla navržena opatření co nejúčinnější a bylo možné porovnat dosavadní proces řízení projektu, byl projekt první rok veden podle pravidel nastavených v diplomové práci Petra Vogla [1]. První rok vedení podle nastavených pravidel bude také popsán v této kapitole. Historie projektu je poměrně obsáhlá a je možné ji dohledat na wiki projektu.

4.1 Informace ze schůzky s původními tvůrci projektu

V první fázi přebírání projektu jsme se věnovali studiu diplomových prací a semestrálních prací projektu Space Traffic. Po přečtení diplomových prací a všech dostupných semestrálních prací jsme měli stále velmi málo informací o tom, jak projekt ve skutečnosti fungoval. Hledali jsme tedy další informace o průběhu projektu. Nakonec nám bylo umožněno kontaktovat původní tvůrce projektu, autory práce [1] a [15], kteří své práce dokončili společně v roce 2012. Následující podkapitola je shrnutím nejdůležitějších informací, které byly zjištěny na této schůzce a které pro nás v této fázi projektu byly velice cenné.

- Oba původní tvůrce doporučují převést Space Traffic mimo školní servery, protože školní server trpí častými výpadky a oprava často trvá i několik dní. Po dalším rozhovoru byl jako ideální řešení vybrán GitHub.
- Oba tvůrce projektu měli velmi špatný názor na některé semestrální a závěrečné práce některých studentů. Podle získaných informací byly tyto práce splněny na minimální možnou mez a i přes to, že bylo splněno zadání, do projektu je nelze integrovat. (Toto tvrzení se později ověřilo při další spolupráci se studentem, na kterého nás původní tvůrce přímo upozornili.)
- Vedoucí projektu, který vedl projekt po původních tvůrcích projektu v akademickém roce 2012/2013, se podle dostupných informací o týmy nestaral a pravděpodobně nevěděl, jak projekt vést. (Toto tvrzení bylo ověřeno po dalších schůzkách s některými členy jeho týmu.)
- Členové katedry dávají na projekt špatné reference, protože v akademickém roce 2012/2013 se v projektu pravděpodobně ukrývali studenti, u kterých nebylo možné dokázat jejich část práce a kteří pravděpodobně splnili semestrální práce bez požadovaného úsilí.

- Autoři doporučují vrácení změn minulého roku, ale doporučují se o něm nezmiňovat, protože se s ním zahodilo množství práce.
- Je velmi důležité, aby bylo možné dokázat, kdo co v projektu udělal. Rozebírání předchozích chyb je silně nedoporučeno.
- Automatické testování projektu není funkční, i když dokumentace tvrdí opak.
- Problém je vytváření herního obsahu, který není možné zadat jako semestrální práci, ale čas je do něj nutné investovat.
- Upozornění, že tým a noví členové týmu většinou nečtou dokumentace, lepší je podle názoru autorů pořádat pravidelné schůzky a školení.
- Členy týmu je nutné vyzkoušet, kvalita studentů je velice různorodá. Bohužel si většinou nelze vybírat. (Toto tvrzení bylo také ověřeno při další práci na projektu.)
- Je nutné sepsat pravidla projektu a donutit všechny studenty k jejich dodržování.
- Vytváří se hra, měla by to být zábava.

Následující informace jsou krátkým shrnutím schůzky s členem týmu projektu Space Traffic, který pracoval na projektu v rámci zmiňovaného akademického roku 2012/2013. S ostatními členy projektového týmu se nám nepodařilo sejit.

- Minulý vedoucí nedělal žádné schůzky, bylo velmi těžké na projektu pracovat.
- Po skončení původních tvůrců projektu v roce 2012 existovala funkční spustitelná verze, které ale chybělo mnoho funkcionality
- Betaverze z dalšího období vývoje ve skutečnosti nefunguje, byla vytvořena jen na Den otevřených dveří, aby vytvořila iluzi, že vývoj pokračuje dobře. Vznikla proto, aby vývojový tým neměl problémy ze zastavení vývoje.
- Doporučení vrácení změn za celé dva roky i přes to, že tento student v rámci projektu pracoval na refaktoringu kódu.
- V projektu se nikdy nestihlo nic otestovat. Vedení testování nepřikládalo velkou váhu, hlavním úkolem bylo vždy dokončit co největší část funkcionality a na testování nezbýval čas.
- Doporučuje převedení na GitHub.

4.2 Technická specifikace projektu

Architektura projektu je podrobně popsána v [15]. Vychází ze vzoru model-view-controller (MVC architektura). Klientská část webové aplikace běží v internetovém prohlížeči a je napsaná v HTML5 s použitím LESS. Interaktivní prvky zajišťuje JavaScript s frameworkem jQuery.

Virtuální server hry zajišťuje běh tří aplikací. Jedná se o vývojářskou wiki, zajišťující znalostní bázi projektu, která je vytvořena pomocí PHP CMS TikiWiki. Dále jde o aplikaci herní simulace, vytvořenou zejména za použití technologií C#, .NET a Entity framework. Třetí aplikací na virtuálním serveru je webové rozhraní hry, implementované pomocí MVC frameworku ASP.NET 3. Virtuální server používá operační systém Windows Server 2008 R2.

Jako hlavní vývojový nástroj bylo využíváno Visual Studio společnosti Microsoft.

4.3 Analýza projektu Space Traffic popisem situace

Podle definice projektu a programu v kapitole 3 je Space Traffic možné považovat spíše za program, než za projekt. Space Traffic nemá jasně definovaný konec a skládá se z mnoha projektů, které jsou členěné do semestrů a akademických let. Z časového hlediska se však jedná spíše o menší projekt. Na projektu sice pracuje velké množství studentů, ale ti mají omezené časové možnosti.

Poprvé v historii projektu je projekt svěřen na dva roky stejnému vedoucímu. Díky tomu je možné vyzkoušet v průběhu řízení projektu dva různé způsoby ve dvou letech.

Pro projekt Space Traffic byla provedena stejná analýza, jako pro projekty ve vybraných firmách z kapitoly 3. Vzhledem k tomu, že k této analýze došlo při přebírání projektu, nebylo ještě možné na některé otázky odpovědět. Analýza ukazuje, co bylo o projektu známo v době jeho přebírání. Tato analýza je součástí přílohy D.

Stav projektu v době převzetí byl žalostný. Vedoucího projektu z akademického roku 2012/2013 nebylo možné kontaktovat. Podle dostupných informací se mu nepodařilo za sebe nalézt nového vedoucího pro akademický rok 2013/2014 a z projektu odešel. Projekt proto převzal zpět Petr Vogl, původní tvůrce, a v roli mentora se snažil projekt dál podporovat, i když už byl v tuto dobu mimo katedru.

Informace o projektu byly zjišťovány s velkými obtížemi. Jediné, co bylo možné nastudovat, byla historie projektu a již dokončené diplomové práce pojednávající o projektu. Ze získaných informací vyplynulo, že na konci práce [1] a [15] existovala stabilní verze projektu. Zároveň bylo zjištěno, že veškerá komunikace ohledně projektu za vedení těchto autorů probíhala v rámci osobních schůzek a nebylo ji tedy možné dohledat. Znalostní báze projektu neobsahovala informace potřebné k tomu, aby projekt mohl převzít další vedoucí. Navíc od poslední stabilní verze proběhlo na projektu ještě

několik prací v dalších akademických letech 2012/2013 a 2013/2014, které podle získaných informací vedly k destabilizaci projektu. Hlavním problémem projektu Space Traffic v roce 2012/2013 bylo, že minulý vedoucí se věnoval hlavně své diplomové práci a studenti v projektu zůstali bezradní a bez vedení. Tyto skutečnosti vysvětlují špatnou pověst projektu a důvod, proč projekt šel ke svému zániku. V dalším roce 2013/2014 se o projekt sice staral původní tvůrce, ale měl pouze omezené časové možnosti.

Po převzetí projektu byl problém vůbec spustit hru i na lokálním prostředí. Bylo tedy nutné studovat doprovodné práce, ze kterých ale požadované informace pro spuštění projektu nebyly zjištěny. Proběhlo tedy množství schůzek s některými původními tvůrci projektu. Po těchto schůzkách již bylo trochu jasnější, jak projekt fungoval.

K projektu existovalo hned několik wiki bez zjevného důvodu. Některé informace byly duplicitní a některé nebyly obsaženy vůbec. I když bylo možné dohledat dokumentaci k již ukončeným semestrálním pracím, nebylo bohužel možné zjistit, jak kvalitně byla práce splněna. Jak bylo později zjištěno v průběhu vývoje, na kvalitě práce a dokumentace závisí další úspěšný vývoj projektu. Samotná informace o splnění práce zde nemá váhu. Pokud je práce splněna na nejnižší možnou mez, to znamená, že zadání je ve všech bodech splněno a hodnoceno stupněm dobře, student má nárok na zápočet. Taková práce ale obvykle není v celkovém projektu použitelná, protože jí chybí spousta detailů a ve výsledku kvalitu projektu jen zhorší. Vznikla tedy naléhavá potřeba zjistit, jak která zadání z minulých let dopadla. Také tím byla vysvětlena situace, kdy jsme v semestrální práci našli popis problému i jeho řešení, ale v implementační části projektu toto řešení chybělo. Dalším problémem několika prací bylo, že sice vzniknuly a měly splněné zadání, ale vzhledem k tomu, že neexistovala funkční verze projektu, nebylo možné vyzkoušet integraci zadání do projektu. Tímto způsobem vzniklo spousta nadějných zadání, ale kvůli nemožnosti jejich integrace do projektu a nedostatečné dokumentace byly ve výsledku zbytečnou prací.

4.3.1 Rozdělení týmů a lidské zdroje

Projekt Space Traffic se s problémem změny týmu včetně vedení potýká velice těžko. Pokud předchozí vedení našlo své nástupce, nebyl přechod tak drastický, protože po několika schůzkách a pozorování předchozího vedoucího mohl nový vedoucí převzít projekt bez větších obtíží. Velké problémy nastávaly a nastávají pravidelně v situaci, kdy z projektu odešlo kompletní vedení projektu, včetně celého týmu. V této situaci je velký problém se do projektu začlenit. Vedení projektu před námi bylo přímo závislé na množství schůzek, které v rámci vedení projektu pořádalo. Osobní schůzky pomohly k lepšímu začlenění členů týmu a lepšímu povědomí všech členů o situaci projektu, ale zároveň vedly k tomu, že velké množství informací nebylo nikde napsáno. Tato skutečnost velice ztížila začlenění nového týmu v začátcích této práce.

4.3.2 Znalostní báze

Projekt má samozřejmě vedenou svou znalostní bázi. Bohužel, i když existuje, je nejednotná a na mnoha místech. Ještě půl roku po převzetí projektu byly nacházeny další nové wiki projektu Space Traffic s nekonzistentním obsahem, jejichž existence nebyla vysvětlena. Nepřehledná situace ohledně znalostních bází také vedla k tomu, že nebylo možné dohledat relevantní informace o projektu v době, kdy byly potřeba. Často se stávalo, že existoval podrobný návod pro určitou problematiku, ale byl nalezen až o mnoho později, kdy už byl daný problém vyřešen a bylo na něm stráveno zbytečné množství času.

Dalším problémem je najít všechny kompetence vedení. Existuje například dokument HAND OVER NOTES, kde jsou přehledně uložena veškerá hesla k projektu. Informace o tom, kde se přihlásit a o jaký server se jedná už ale ovšem chybí. Tyto informace byly získávány konzultacemi a zjišťováním informací napříč celou katedrou.

4.3.3 Nástroje projektu

Pro práci je použit systém pro správu verzí SVN, ale spolupráce je zde velice nekontrolovatelná. Velice těžko se dohledávají změny a dělají opravy kvůli nepřehlednému způsobu použití, nedostatku dokumentace a slabých konvencí projektu.

Redmine projektu momentálně nemá pevně daný systém třídění podprojektů projektu Space Traffic. Je zde množství neuzavřených podprojektů. Zvláštností je, že zde není ani jediný projekt, nebo semestrální práce vedení z minulého roku. Poslední práce jsou z roku, kdy vznikla poslední stabilní verze hry. Následující vedení pravděpodobně Redmine nepoužívalo.

Komunikace v projektu v minulosti probíhala pomocí nástroje Skype. Konverzace o projektu, které by snad mohly vysvětlit některé části projektu, jsou tak nenávratně ztracené.

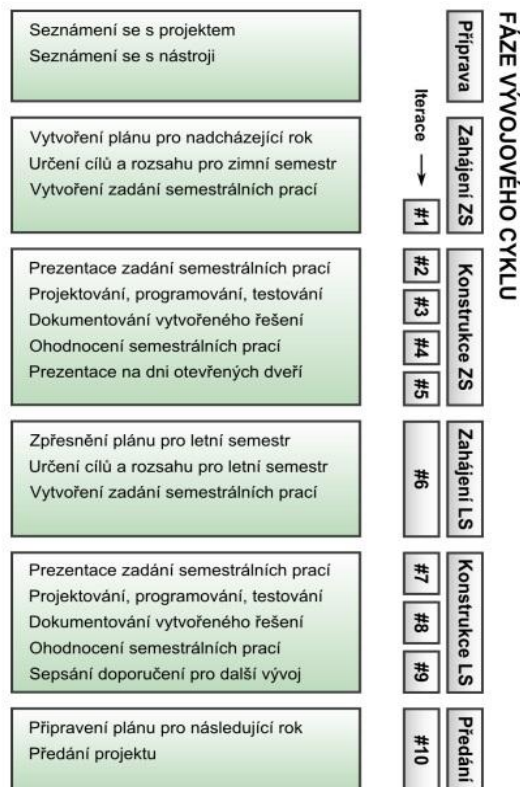
4.3.4 Současný proces projektu Space Traffic

Tento proces byl původním autorem vytvořen přímo pro potřeby projektu, který funguje ve školním prostředí. Proces je inspirován metodikou RUP. Před začátkem celé práce probíhá přípravná fáze, která časově odpovídá přípravě před začátkem semestru. V průběhu semestru pak probíhají fáze zahájení a konstrukce, které jsou vzhledem ke členění akademického roku zdvojené. Na konci celé práce pak proběhne fáze předání projektu dalšímu vývojovému týmu. Vše dobře ukazuje Obrázek 4.1: Fáze vývojového cyklu během akademického roku podle Petra Vogla.



Obrázek 4.1: Fáze vývojového cyklu během akademického roku podle Petra Vogla

Během těchto fází vývojového cyklu probíhají přesně definované činnosti, které ukazuje Obrázek 4.2: Činnosti prováděné během vývojového cyklu Petra Vogla.



Obrázek 4.2: Činnosti prováděné během vývojového cyklu Petra Vogla

Proces projektu Space Traffic je podrobněji popsán v [1]. V roce 2012/2013 byl tento proces novým vedoucím mírně upraven. Vzhledem k již vysvětleným problémům tohoto vedení a po diskuzi s vedoucím diplomové práce se ale v této práci vracíme k původnímu procesu popsaném v [1]. Pro první rok vedení v rámci této diplomové práce bude použit tento proces. V dalším roce pak dojde k jeho zhodnocení a případnému vylepšení.

4.4 SWOT analýza projektu

Pro určení silných a slabých stránek projektu byla provedena SWOT analýza projektu. SWOT analýza projektu pomáhá k určení hrozeb projektu.

Silné stránky

- Dobrý koncept a gamedesign projektu.
- Vývoj a dynamika projektu (nové myšlenky, noví členové).
- Zajímavost pro nové členy týmu, nové studenty.
- Možnost identifikace v týmu.
- Možnost práce v rámci semestrálních prací (odměna, motivovanost).
- Studentský projekt (očekávání, velké možnosti).

Slabé stránky

- Častá změna vedení (náchylnost ke špatnému úsudku).
- Zvyšující se implementační složitost (problém znát celý projekt).
- Testování.
- Dokumentační složitost (problém znát celý projekt).
- Závislosti jednotlivých zadání.

Příležitosti

- Inovace a nové postupy díky střídání týmů a vedení (nové myšlenky, lepší nástroje).
- Filtrování pouze potřebných informací.
- Zamezit neustálému vytváření nových dokumentů (je nutné upravovat stávající).
- Možnost velkého rozvoje miniher po implementaci rozhraní.
- Možnost pro studenty vyzkoušet si nástroje, myšlenky a způsoby vedení.
- V případě úspěchu velká podpora KIV (i stipendia).

Hrozby

- Nemožnost poznat úspěšnost projektu (provedení).
- Časté změny týmu i vedení (nepochopení projektu).
- Nepřehlednost znalostí o projektu.
- Předávání projektu
- Zvětšující se projekt s každým novým týmem (nabalování, problém rozumět projektu komplexně).
- Časté vrácení změn (po přebrání projektu novým vedením menší či větší vrácená změna zvyšuje nedůvěru v projekt).
- Nemožnost kontroly shora - od vyučujících.

Ze SWOT analýzy projektu Space Traffic je patrné, že pro projekt jsou klíčové otázky týkající se změny zdrojů, kontroly kvality kódu a vytváření znalostní báze.

4.5 Popis prvního roku vedení

Tato podkapitola se věnuje popisu prvního roku vedení projektu Space Traffic. Vedení celého projektu se věnovala autorka této práce spolu s autorem práce [16], Janem Kotalíkem. Tento autor se staral především o technické zázemí projektu. V prvním roce vedení bylo na základě získaných informací rozhodnuto, že projekt bude veden podle procesu popsaného v 4.3.4. Již o letních prázdninách před akademickým rokem

2014/2015 tedy začala přípravná fáze, ve které byly studovány všechny dostupné informace k projektu Space Traffic. Po nastudování dostupných prací jsme stále nedokázali projekt uchopit. Informace byly roztržité a nedostupné a vedení projektu z předchozího roku 2013/2014 bylo nedostupné. V zimním semestru byly tedy práce na projektu pozastaveny. Tento čas byl věnován snaze zkontaktovat původní týmy a vedení projektu a úpravě znalostní báze. Na toto téma byly v zimním semestru roku 2014/2015 zadány dvě semestrální práce, které byly zadány v rámci předmětu KIV/ZIM:

- Jan Kotalík: Návrh architektury TikiWiki, uživatelské wiki a realizace – KIV/ZIM.
- Klára Beránková, Ondřej Sadílek: Konsolidace dat ve wiki – KIV/ZIM.

Obě zadání byla úspěšně dokončena a splnila očekávání. Pro znalostní bázi byl využit původní nástroj TikiWiki, který byl popsán v [1]. Všechny staré vývojářské i hráčské wiki byly po dlouhém zjišťování přístupových údajů zrušeny. V případě, že zrušení nebylo možné, byl na wiki zadán odkaz na tuto jedinou wiki (je znám jeden takový případ). Po zimním semestru měl tedy projekt fungující znalostní bázi s informacemi, které byly posbírány z ostatních zdrojů. Projektu byl také v této době převeden na GitHub, protože SVN podle dostupných informací nebyla vhodným nástrojem (více kapitola 5.1). Také se během zimního semestru podařilo projekt poprvé spustit. Projekt nebyl funkční, na což jsme ale byli upozorněni při schůzce s původními tvůrci projektu. Také jsme zjistili na kterém serveru je vše uloženo a že kontaktní osoba pro problémy se serverem je pan Ing. Ladislav Pešička.

V letním semestru pak vznikla první implementační zadání, která byla vedena podle procesu popsaného Petrem Voglem [1]. V únoru roku 2014 se také přihlásil k projektu student, který na něm měl postavenou svou bakalářskou práci a o této skutečnosti nás neinformoval. Byl tedy velice překvapený ze změn, které se v projektu udály. Jeho zadání nakonec muselo být změněno.

V letním semestru probíhaly na projektu tyto práce:

- Jan Kotalík: Návrh obchodu s programy pro lodní počítač webové hry – KIV/OPSWI.
- Jan Kotalík: Obecná podpora – KIV/NET
- Jan Smitka: Testování, ukládání stavu a podpora – KIV/NET.
- Martin Bláha: Zobrazení lodí na mapě – KIV/NET.
- Pavel Fakan: Generátor hvězdných systémů a mapa galaxie – bakalářská práce
- Tým IT Crowd (Radek Bouda, Jan Bláha, Jakub Marek, Miroslav Lang): Nemovitosti – KIV/ASWI.
- Tým M4 Developers (Michel Soběhart , Martin Hron, Michael Hadáček, Martin Bláha): Plánovač nákupů a prodejů – KIV/ASWI.

Celkem tedy na projektu tento semestr probíhalo 6 semestrálních prací, na kterých pracovalo 11 studentů, někteří pracovali na více zadáních najednou. Celý proces probíhal podle pravidel popsanych v diplomové práci [1].

Vyhodnocování prací v tomto semestru zatím probíhalo pouze systémem splněno/nesplněno. Pouze u týmů z předmětu KIV/ASWI byl pro vyhodnocení týmu použit formulář, který byl vyžadován garantem předmětu. Tento formulář se pak v dalším roce stal běžnou součástí hodnocení všech semestrálních prací. Ostatní práce jsou stručně zhodnocené v následujících řádcích a toto stručné hodnocení je dostupné také na wiki projektu v sekci pro vedoucí.

Práce návrh obchodu s programy byla splněna a v dalším roce na ni bylo úspěšně navázáno.

Práce obecná podpora se věnovala řešení drobných problémů v projektu a opravování chyb. Byla zadána hodinově a splněna nad svůj časový rámec.

Práce testování, ukládání stavu a podpora byla splněna kvalitně a odborně, ale vzhledem k problémům s přístupy k serveru a Redmine nebyla dokončena integrace tohoto zadání do celkového projektu.

Práce zobrazení lodi na mapě nebyla dokončena. Student na této práci ani nezačal pracovat. Práce navazovala na zadání plánovač nákupů a prodejů a vzhledem k tomu, že student přestal pracovat už v rámci předcházející týmové práce, nemohl pak na zadání plánovač navázat a práci rovnou vzdal.

Hodnocení bakalářské práce bylo inspirováno formulářem pro vyhodnocení týmu předmětu KIV/ASWI. Pro tyto účely byl formulář mírně upraven a je sekce pro vedoucí wiki projektu Space Traffic [17].

Hodnocení týmu IT Crowd není v příloze obsaženo, protože tým práci nedokončil a nedodal žádnou funkční část implementace. Tým navštěvoval schůzky a prvních pět týdnů se věnoval studování projektu Space Traffic a návrhům své další práce. Když byla požadována alespoň část funkcionality, tým práci ukončil. Přes velkou snahu vedoucích projektu žádný z členů týmu nechtěl individuální část práce, aby mohl část zadání dokončit sám.

Hodnocení týmu M4 developers je wiki projektu, sekce pro vedoucí projektu [17].

4.6 Další zjištěné informace v průběhu vedení prvního roku

V průběhu prvního roku vedení podle popisu z diplomové práce [1] byly zjištěny další informace o průběhu studentského projektu. Hlavním zjištěním je, že ani přečtení veškeré dokumentace a četné schůzky nepomohly ke komplexnímu pochopení celého projektu. Po prvním roce lze říci, že jsme se stali součástí projektu, ale i přes to stále nastávají situace, které nás mohou překvapit. Klíčem k úspěchu je hledání okamžitého řešení a v případě neúspěchu je nutné se poradit se s mentorem projektu. Tímto způsobem byly všechny aktuální hrozby vyřešeny tak, aby projekt utrpěl co možná nejméně.

4.6.1 Kontrola vedoucích

Vzhledem k tomu, že se jedná o studentský projekt, nebyla zde nikdy zavedena kontrola shora (od vyučujících). V případě, že vedoucí zastával svou funkci dobře, nebylo nutné tuto kontrolu provádět. Jak však nyní ukazuje historie projektu, velký problém nastává v případě, kdy vedoucí projektu plní práci v nejnižším možném rozsahu a zajímá se víc o splnění svého zadání, než o cíle celého projektu. I když tento vedoucí splní zadání své práce a členové týmu také zadání splní, projekt tímto přístupem ve výsledku utrpí a spolupracující studenti a týmy jsou v průběhu práce velice zmateni a bez vedení, jak ukazují informace ze schůzek s původními členy vývojového týmu Space Traffic.

4.6.2 Přesné dodržení zadání

Již Petr Vogl ve své práci [1] uváděl, že není možné po studentech chtít práci nad rámec jejich zadání. Toto tvrzení se potvrdilo. Studenti v rámci semestrálních prací striktně plní zadání. Může snadno nastat situace, kdy je zadání splněné, ale pro projekt nepoužitelné. Zadání také musí obsahovat všechny náležitosti. Bylo ověřeno, že i přes to, že jsou body testování a podrobná dokumentace součástí konvencí projektu, je velice těžké vysvětlit tuto skutečnost studentům, pokud není přímo součástí zadání, které obdrželi.

4.7 Motivace studentů

Motivace studentů je velmi důležitý faktor úspěšnosti celého projektu. Byla identifikována dvojí motivace studentů:

- Vlastní zájem studenta
- Motivace splněním předmětu

To je také rozdíl oproti komerčnímu projektu, protože zde není možná motivace penězi. Jedinou motivací studentů je získání zápočtu a dobrý pocit. Student většinou, pokud má na výběr, zvolí lehčí zadání semestrální práce časově co nejméně náročné.

Po schůzkách se studenty, kteří své práce na projektu již dokončili, bylo často uvedeno, že pracovat v projektu Space Traffic je těžší než běžné práce a obhájit práce v tomto projektu je ještě těžší. Velké množství hodin do projektu vložených se tedy nevrátí. Jedinou motivací těchto studentů tak zůstal dobrý pocit, že projekt bude součástí propagace katedry, ale zpětně by si práci v projektu většinou znovu nevybrali.

4.8 Shrnutí problémů projektu

Space Traffic se jako rozsáhlý studentský projekt potýká s řadou problémů, se kterými se běžné komerční projekty neseťkávají. Tyto problémy jsou patrné z analýz. Například vedoucí projektu nepracuje s časem vyjádřeným penězi, ale se zadáními. Pokud se během práce vyskytnou úkoly nad rámec zadání, vedoucí pak nemá prostředek, jak zajistit doplnění těchto úkolů. Velký problém také nastává při předávání projektu novému vedoucímu a novému týmu. Projekt velmi trpí problémy s předáváním informací a znalostí. Podle informací z praxe v kapitole 3 v průběhu většiny komerčních projektů nenastávají situace, kdy se změní kompletní tým projektu včetně vedení.

5. Výběr vhodných mechanismů a provedené změny

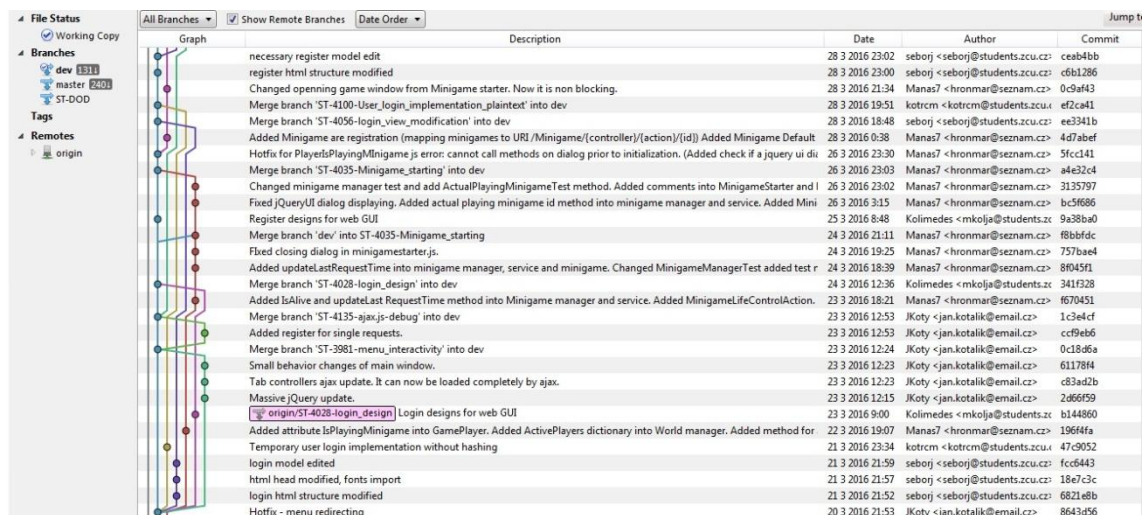
V kapitole 4 byl čtenář seznámen s průběhem prvního roku vedení projektu a s jeho hlavními problémy. Pro nalezené problémy byly po prvním roce vedení navrženy mechanismy pro jejich odstranění, nebo minimalizaci. Pro lepší představu o tom, jak se projektové problémy řeší v praxi, byly navštíveny čtyři firmy (viz kapitola 3), kde bylo pozorováno řešení těchto problémů v praxi.

Tato kapitola je věnována vytváření mechanismů přímo pro projekt Space Traffic. Dále jsou popsány změny, ke kterým došlo v rámci vedení projektu v letech 2014/2015 a 2015/2016. Ve školním prostředí nelze použít většinu komerčních řešení a to hlavně z důvodu nedostupných zdrojů. Například Firma K ve 3.1 uvádí, že pro kontrolu kvality produktu je spotřebována přibližně třetina všech zdrojů projektu.

Po dlouhém zjišťování informací o projektu, konzultacích s minulými vedoucími a konzultacích s mentory projektu v rámci katedry bylo rozhodnuto o vrácení změn posledních dvou let projektu. Rozhodovalo se o tom, zda je lepší projekt čistit od nepřehledného kódu, nebo zda je lepší se vrátit k poslední kvalitní verzi. Poslední kvalitní verzi jsme pak dostali od původního tvůrce projektu, Martina Štěpánka.

5.1 Převedení na GitHub

Verze od Martina Štěpánka byla nahrána na server GitHub, kam jsme po konzultacích s původním vedením a mentorem projektu celou hru převedli. Před samotným převedením byl od všech původních tvůrců projektu získán souhlas k nahrání jejich práce na server GitHub pod licenci Apache License, Version 2.0. Tato licence byla doporučena mentorem projektu. Kopie těchto podepsaných povolení jsou na wiki projektu [17] v sekci pro vedoucí projektu. Hlavním důvodem převodu byla snaha o zpřehlednění vývoje projektu. Do té doby používaný systém pro správu kódu SVN byl pro tento účel velmi nedostačující. Aplikace Source Tree pro práci s verzovacím systémem Git nabízí velmi přehledné zobrazení probíhající práce, jak ukazuje Obrázek 5.1: verzovací systém Git v aplikaci Source Tree.



Obrázek 5.1: verzovací systém Git v aplikaci Source Tree

Do wiki projektu [17] byl vložen podrobný návod, jak stáhnout a nainstalovat aplikaci Source Tree, jak projekt do aplikace stáhnout a jak s aplikací pracovat (vytváření nových větví, commit).

5.2 Redmine projektu

Jak je patrné z analýzy v kapitole 4.3.3, projekt používá systém pro správu úkolů Redmine. V tomto systému ale neexistovala jednotná pravidla pro vytváření nových podprojektů projektu Space Traffic. Také zde bylo nalezeno několik nedokončených podprojektů z roku 2012. V roce 2012/2013 a 2013/2014 nebyl systém Redmine používána vůbec, nebo velmi zřídka. V sekci pro vedoucí projektu tak vznikla nová stránka o tom, jak do Redmine vkládat nové podprojekty a jaké náležitosti by měly splňovat. (jméno podprojektu, vedoucí, členové týmu, jméno zadání, kompletní zadání, rok zadání)

5.3 Vylepšení znalostní báze

Z analýzy firem v kapitole 3 vyplývá, že všechny firmy považují znalostní bázi za mechanismus, pomocí kterého lze překlenout rychlé střídání se studentů v projektu. Obsáhlá a přehledná znalostní báze také umožňuje hladké předávání projektu, protože v případě, že je vše dokumentované, nedochází ke ztrátě informací.

Jak je patrné z analýzy projektu v kapitole 4.3.2, znalostní báze projektu sice existovala, ale jak jsme se několikrát přesvědčili, byla nedostačující. Hlavním problémem bylo, že pokud už ve znalostní bázi daná stránka existovala, většinou ji nikdo nenašel. Analýza výběru vhodné znalostní báze pro projekt Space Traffic byla provedena v rámci [1]. Po

nastudování důvodů jejího výběru a vzhledem k tomu, že v projektu bez problémů fungovala, byla ponechána. Po převzetí projektu však došlo k jejímu kompletnímu přepracování v rámci dvou semestrálních prací z předmětu KIV/ZIM. Navržena byla celá nová struktura projektové wiki [17] a wiki byla do této struktury předělána. Druhá práce byla věnována kompletnímu roztřídění archívu z SVN projektu do struktury wiki. Součástí práce také bylo vhodné odkazování a tagování všech dostupných dokumentů a návodů.

Nově do projektové wiki [17] také přibyla sekce pro vedoucí, kam běžný student nemá přístup. V této sekci jsou rady pro nastupující vedoucí, vznikla tu obdoba stránky *Jak do projektu* pro vedoucí, je tu napsán seznam všech povinností vedoucího, a co všechno k vedoucí roli v projektu patří. Dále je tu plánování, plány do budoucna a co si má nový vedoucí zjistit.

Hesla projektu Space Traffic nejsou uložena v projektové wiki [17]. Z bezpečnostních důvodů existuje dokument Hand over notes, který si vedoucí projektu předávají mezi sebou v tištěné podobě a ten všechna hesla obsahuje.

Hlavním novým znakem projektové wiki [17] je jednotnost. Z analýzy v kapitole 4.3.2 je patrné, že zde nastal velký problém s nejednotností wiki. Všechny existující wiki byly převedeny do této jediné vývojářské wiki a zrušeny.

Kvalita stránek vkládaných do wiki projektu [17] je kontrolována pomocí kontrolních seznamů, které jsou podrobněji popsány v kapitole 5.6. Také jsou zde uvedené konvence pro vytváření nové dokumentace.

Wiki pro hráče zatím nebyla implementována, je součástí implementačních plánů do budoucna. Wiki v CMS Wordpress, která byla v projektu při našem převzetí, byla zrušena vrácením změn a návratem k verzi Martina Štěpánka.

5.4 Kontrola celého projektu

Na wiki projektu [17], hned v úvodní stránce, byla vytvořena stránka konvence projektu. Jedná se o soupis základních pravidel, které je nutné v rámci projektu dodržovat. I přes to, že je zde vše podrobně uvedené, občas v projektu dochází k nejasnostem. Nejasnosti vyplývají z neznalosti studentů. Je potřeba studenty na konvence projektu několikrát upozornit a konvence jim přímo přečíst. Na dodržování konvencí projektu je nutné důsledně trvat, protože díky nim je možné projekt jednotně vyvíjet.

5.5 Kontroly shora

V kapitole 4.6.1 byl vysvětlen problém kontroly vedoucích. Je velmi nutné tuto kontrolu zavést, protože další ztrátu důvěry v projekt ze strany katedry již nelze připustit. Bez podpory katedry projekt dospěje ke svému nevyhnutelnému konci. K přidání kontroly vedoucích projektu slouží nový kontrolní proces, který byl přidán nad původní proces a zobrazuje ho Obrázek 5.2: Nový proces s kontrolní úrovní. Původní proces byl podrobně rozepsán v [1] a v kapitole 4.3.4.



Obrázek 5.2: Nový proces s kontrolní úrovní

Jak naznačuje Obrázek 5.2: Nový proces s kontrolní úrovní, došlo k doplnění původního procesu, který se ukázal za první rok jeho používání jako nedostatečný. Především zde naprosto chyběly kontroly shora, proces byl tedy doplněn o další úroveň, kde byla implementována kontrola. K celému původnímu procesu tím přibyl nadřazený proces, který bude sloužit k tomu, aby byla možnost kontrolovat stávající proces. Tento návrh je inspirován metodikou RUP. Díky této kontrole je možné v průběhu projektu kontrolovat jeho směřování a případně odstínit riziko špatného rozhodnutí už ve fázi plánování.

Kontroly budou probíhat každý semestr začátkem a koncem při dosažení příslušného milníku. Vedoucí projektu je povinný na začátku semestru přednést na prezentaci mentorovi projektu současný stav projektu a plán, kam se projekt má za půl roku dostat. Další prezentace proběhne na začátku letního semestru, kde vedoucí projektu opět ukáže dosažené výsledky za zimní semestr a plánování na letní semestr. Poslední prezentace pak proběhne na konci letního semestru, kdy vedoucí projektu odevzdá mentorovi projektu plán pro dalšího vedoucího a také zhodnocení výsledků, kterých dosáhl.

5.6 Kontroly zdola

Podle informací získaných ve firmách v kapitole 3 lze problém nedostatečného množství zdrojů pro kontroly projektu obejít propracovaným systémem kontrol zdola, kde tyto kontroly provádějí sami pracovníci projektů. Po těchto konzultacích a po konzultacích s mentorem projektu vznikly jednoduché kontrolní seznamy (tzv. checklisty), které slouží k samokontrolě odvedené práce.

Jedná se o kontrolu splněných úkolů studentů formou několika jednoduchých otázek, na které si studenti po dokončení úkolu odpovídají. Sami si tím hodnotí kvalitu své práce a dodržování konvencí projektu. Vedoucí má díky kontrolním seznamům rychlý přehled o tom, jak práce probíhala.

Povinnost používání kontrolních seznamů byla v druhém roce vedení projektu přidána ke konvencím projektu. Diplomanti, kteří do projektu nastoupili před touto změnou, neměli povinnost tyto kontrolní seznamy vytvářet. Tato povinnost platila pro všechny nové členy projektu.

Další výhodou používání kontrolních seznamů je, že jsou vývojáři nuceni plánovat co nejmenší úkoly, protože se zvyšující se časovou náročností úkolu roste i počet otázek v kontrolním seznamu.

Kontrolní seznamy, které byly navrženy pro potřeby projektu Space Traffic, jsou součástí přílohy A, která je kopií wiki stránek. Dále jsou dostupné na wiki projektu [17] v sekci konvence projektu.

5.7 Přijímání nového člena projektu

Jak bylo zjištěno, přijímání nových členů projektu nebo celých týmů, není zcela jednoduchou záležitostí. Rozdíly ve znalostech studentů jsou velké. Pokud je možnost výběru, je problém kvalifikovaně rozhodnout, který tým, nebo student se na vypracování sporného zadání hodí více. Z tohoto důvodu vznikl dotazník pro nové členy týmu, který je součástí přílohy B, kde jsou seřazeny otázky od triviálních po složitější. Použití tohoto dotazníku bylo ověřeno při zadávání semestrálních prací v průběhu letního semestru akademického roku 2015/2016, kdy se o stejné zadání ucházely dva týmy. Otázka „Co je to getter?“ byla pro první tým neřešitelným problémem. Druhý tým naopak prokázal technickou zdatnost i u velmi složitých otázek. Zadání získal a úspěšně dokončil. První tým, který si vybral jiné zadání Space Traffic, své snažení po čtyřech týdnech semestru vzdal. Tento dotazník, který je součástí wiki projektu v sekci pro vedoucí. Pomohl vybrat lepší tým a vyhnout se zklamání z nedokončení zadání.

Ve spolupráci s Janem Kotalíkem byla také vytvořena nová stránka „Jak si zprovoznit projekt“ na wiki projektu [17], kde je přehledný návod v bodech, jak se do projektu začlenit. Je zde informace o tom, jaké nástroje je nutné nainstalovat a jejich nastavení. Dále je zde popsáno, jak si stáhnout a zprovoznit projekt. Funkčnost tohoto návodu byla prakticky ověřena vývojovým týmem Th3 Developers, v létě 2016. Šlo o studenty druhého ročníku bakalářského studia a všichni tři členové týmu Th3 Developers si podle tohoto návodu dokázali sami stáhnout a spustit projekt.

Podobné testovací otázky jsou vypracovány i pro nového vedoucího projektu, v případě více uchazečů je tedy možné zjistit, který se do vedení projektu lépe hodí.

Otázkou je, zda je pro projekt žádoucí, aby byl za každou cenu přidělen jakémukoliv vedoucímu. Zkušenosti z minulosti ukazují, že je pro projekt pravděpodobně lepší, když zůstane na několik let pozastaven, než když jsou do něj zaneseny neodstranitelné chyby a stejně musí dojít k vrácení změn.

5.8 Hodnocení splněných semestrálních prací

V minulosti projekt často trpěl problémem nemožnosti zjistit, jak dané zadání dopadlo. Pokud k projektu přišel vedoucí, který se již nepotkal se svým předchůdcem, našel například mnoho zmínek o určité semestrální práci, nakonec i text této práce, a byl zde přehnaný popis kvality výsledku této práce. Informace o tom, jak je práce kvalitní ve

skutečnosti, zde chyběla. K tomuto účelu také přibyla v sekci pro vedoucí stránka hodnocení úspěšnosti projektu. Do budoucna by tedy již nemělo docházet k problémům s nemožností rozhodnout, zda zadání bylo splněno pouze na nejnižší možnou mez. Pro hodnocení kvality provedení semestrální práce byly využity dotazníky pro vyhodnocení spolupráce v rámci předmětu KIV/ASWI, po domluvě s garantem předmětu jsou pouze mírně upravené. Všechna tato hodnocení jsou součástí sekce pro vedoucí projektové wiki. Hodnocení probíhala až v průběhu druhého roku, první rok byla provedena pouze pro bakalářskou práci a tým skupiny M4 developers, který svou práci dokončil.

5.9 Nástroje pro komunikace v projektu

Hlavním nástrojem komunikace v projektu u našich předchůdců byl Skype a hlavně množství osobních schůzek. Na Skype byla založena první hromadná konverzace, do které byli přiřazeni původní tvůrci projektu. Vznikla zde diskuze o problémech s předchozí verzí, ale pro komunikaci s novými členy týmu se Skype jevil jako špatný nástroj. Již v zimním semestru, při práci na znalostní bázi, probíhala jediná a nejrychlejší komunikace přes sociální síť Facebook. Pokusy o zavedení jiného projektového chatu skončily neúspěchem, protože studenti jiné zprávy nekontrolovali, nebo reagovali velice pozdě (vyzkoušen byl Skype a Redmine fórum). Síť Facebook tedy nejprve sloužila pouze k upomínání studentů, aby si přečetli projektovou konverzaci v nástroji Skype, ale časem byla konverzace na Facebook kompletně převedena. V letním semestru byla založena skupina Space Traffic_developers a byly přidáni další noví členové projektu. Vytvořila se zde malá komunita okolo projektu a dokonce i bývalí členové týmu zde občas reagují na některé problémy. Facebook splňuje požadavky na rychlou a snadnou komunikaci. Reakční čas v této skupině je většinou do půl hodiny a většina problémů je tak velice rychle vyřešena. Každý nový člen zde nalezne odpovědi na své otázky a historii skupiny si může každý nový člen skupiny přečíst. Kromě této skupiny vznikla ještě veřejná skupina Space Traffic, která bude sloužit pro propagaci projektu, až bude hotová jeho první funkční verze a soukromá skupina pouze pro vedení projektu, kde velice rychle a efektivně řešíme problémy s vedením projektu.

5.10 Zadávání semestrálních prací v projektu

Jak je uvedeno v kapitole 4.6.2, je velmi problematické provést jakoukoli změnu v průběhu práce na projektu. Pokud je v průběhu semestrální práce zjištěno, že zadání bylo špatně odhadnuto, nemá vedoucí projektu téměř žádnou šanci zadání upravit.

Studenti plní striktně zadání, protože za splněné zadání dostanou zápočet. Také pokud dojde v průběhu práce k nutnosti doplnění zadání, například o integraci do projektu, nebo kosmetickou úpravu, může se vedoucí setkat s odporem a nemá žádný vynucovací prostředek. Je tedy velmi důležité důsledně trvat na uvedení všech náležitostí zadání. Zvláštní důraz by měl být kladen na to, aby povinnost dokumentace a otestování byla přímo součástí zadání a ne pouze ústní domluvy, kterou nelze později prokázat. Dalším nezbytným bodem zadání je upozornění na nutnost dodržovat veškeré konvence projektu, které jsou uvedené na wiki projektu. Na základě informací z firem z kapitoly 3 byl v druhém roce vedení vyzkoušen způsob zadávání prací pomocí výstupu. Bylo ověřeno, že tento způsob je pro projekt o mnoho přínosnější, protože specifikace očekávaného výstupu je mnohem jednodušší, než zvážení všech okolností projektu. Právě tento způsob vytváření zadání semestrálních prací je uveden jako správný v sekci pro vedoucí projektu.

5.11 Shrnutí odstranění hrozeb SWOT analýzy

Tato kapitola je věnována stručnému shrnutí odstranění hrozeb, které byly popsány v rámci SWOT analýzy v kapitole 4.4. K odstranění hrozeb jsou použita všechna opatření uvedená v předchozích kapitolách.

- Nemožnost poznat úspěšnost provedení projektu.
 - Zadání pomocí výstupů (jednodušší kontrola, zda bylo výstupů dosaženo).
 - Zadání musí povinně obsahovat testování, dokumentaci a dodržování konvencí projektu.
- Časté změny týmu i vedení (nepochopení projektu).
 - Zavedení testování nově příchozích vedoucích i členů týmu.
 - Propracovaná znalostní báze s bodovými návody jak do toho pro vedoucí i nové členy týmu.
- Nepřehlednost znalostí o projektu.
 - Zavedení kontrolních seznamů.
- Předávání projektu
 - Test pro nového vedoucího projektu.
 - Snaha, aby odcházející vedoucí našel zástupce sám (prezentace pro nižší ročníky).
- Zvětšující se projekt s každým novým týmem (nabalování, problém rozumět projektu komplexně).
 - Propracovaná znalostní báze.
 - Otázka v kontrolních seznamech (proč změna proběhla).

- Časté vracení změn.
 - Předání plánů projektu do budoucna následujícímu vedoucímu.
 - Vedoucí je kontrolován shora ve fázi plánování a ve fázi odevzdání projektu.
 - Vysvětlit následujícímu vedoucímu důvody jednotlivých opatření, která jsou v sekci pro vedoucí na wiki projektu.
- Nemožnost kontroly shora od vyučujících.
 - Zavedení kontrolních prezentací.
 - Kontrola pomocí testu už ve fázi přijímání nového vedoucího.

5.12 Další zjištěné informace v průběhu vedení druhého roku

Jak uvedl Petr Vogl v [1], studentský syndrom odkládání práce na konec semestru lze mírně odstínit pravidelnými schůzkami. Větší problém ale nastává v případě diplomantů, kteří jsou tlačeni blížící se státní závěrečnou zkouškou. Z důvodu nedostatku času před koncem semestru i ti nejlepší a nejpracovitější studenti podléhají snaze si zadání co nejvíce zjednodušit a některé jeho části pokud možno vypustit. Je tedy nesmírně důležitá dobrá příprava zadání. Je to ochrana projektu i před samotnými vedoucími, kteří jsou většinou kontrolováni pouze splněním zadání.

5.13 Popis druhého roku vedení

V průběhu druhého roku se situace stabilizovala. Do projektu byla zavedena většina změn, které vycházely z analýz ve firmách během zimního roku 2015/2016. I přes velkou snahu vedoucích projektu se nepovedlo zadat semestrální práce na zimní semestr tohoto roku. Z týmu M4 developers byl ale získán další diplomant Martin Hron, který převzal zadání Návrh a implementace miniher pro hru Space Traffic. Dále byly zadány dva projekty v rámci předmětu KIV/OPSWI. Veronika Švecová: simulace modelu ekonomiky herního světa a Martin Hron: implementace modelu ekonomiky herního světa. Dále byla snaha zadat práci na testování projektu, ale i přes to, že toto zadání bylo předneseno v rámci několika předmětů katedry, si ho bohužel žádný ze studentů nevybral. V zimním semestru tedy neprobíhaly žádné týmové práce, ale proběhly veškeré schůzky ve firmách a byla zavedena opatření, která byla popsána výše.

V letním semestru byla snaha zadat především zadání Jazyk lodi, překladač a interpret pro hru Space Traffic, dále pak bylo nutné implementovat herní web. První zadání bohužel nebylo možné zadat, protože se pro něj nenašel v rámci předmětu KIV/ASWI vhodný realizační tým. O zadání herního webu se přihlásily dva týmy studentů

předmětu KIV/ZSWI. Pomocí dotazníku pro přijímání nových členů týmu bylo snadno rozhodnuto, který tým je pro toto zadání vhodnější. Tento tým zadání perfektně splnil a dodal i části nad rámec zadání. Druhý tým žádal o jiné zadání v rámci projektu Space Traffic. Bylo mu přiděleno zadání Testování, které se nepovedlo zadat v zimním semestru tohoto roku. Tým nejprve zrušil schůzku bez udání důvodu a pak svou práci vzdal. I přes snahu vedení projektu zadat individuální zadání členům týmu samostatně tato možnost nebyla využita a žádný ze členů týmu si nepřál pokračovat v práci.

Tento rok tedy na projektu Space Traffic proběhla následující zadání:

- Veronika Švecová: Řízení a procesní mechanismy malých softwarových projektů – KIV/DIP.
- Jan Kotalík: Návrh a implementace uživatelského rozhraní pro hru Space Traffic – KIV/DIP.
- Martin Hron: Návrh a implementace miniher pro hru Space Traffic – KIV/DIP.
- Veronika Švecová: simulace modelu ekonomiky herního světa – KIV/OPSWI.
- Martin Hron: implementace modelu ekonomiky herního světa – KIV/OPSWI.
- Tým Th3 developers (Kolja Matuševský, Marek Kotrč, Jiří Šebor) – webové rozhraní hry Space Traffic.
- Tým SAPAKRI (Sabina Hovorková, Pavel Novák, Kristýna Bernadová) – testování.

Na projektu tedy pracovalo celkem devět studentů v rámci svých prací.

Diplomová práce Jana Kotalíka byla tento rok úspěšně obhájena a byla hodnocena stupněm velmi dobře.

Diplomová práce Martina Hrona byla tento rok úspěšně obhájena a byla hodnocena stupněm velmi dobře.

Zadání implementace modelu ekonomiky herního světa bylo dokončeno, simulace proběhne na tomto modelu v létě 2016.

Hodnocení týmu Th3 developers, který práci dokončil, je součástí sekce pro vedoucí na wiki projektu [17].

Pro projekt Space Traffic byla závěrem provedena stejná analýza, jako pro projekty ve vybraných firmách z kapitoly 3 a stejná analýza byla provedena i při přebírání tohoto projektu. Z této analýzy je možné vysledovat změny, které se v projektu staly. Tato analýza je součástí přílohy E.

5.14 Propagace projektu Space Traffic

Vzhledem k tomu, že se skončením práce Jana Kotalíka [16] byla uvedena první funkční demoverze hry, jsou nyní připravovány způsoby její propagace pro následující akademický rok.

DoD

Den otevřených dveří Katedry informatiky a výpočetní techniky Západočeské univerzity v Plzni umožňuje výborný způsob propagace projektu, kdy si projekt můžou prohlédnout přímo studenti středních škol, pro které je projekt určen. V rámci dne otevřených dveří si studenti středních škol budou moci demoverzi hry vyzkoušet.

Popularizace vědy

Bylo zjištěno, že v rámci univerzity probíhá projekt Popularizace vědy, kdy studenti středních škol připravují rozhovory se studenty vysokých škol o zajímavých projektech, na kterých studenti vysokých škol pracují. Bylo kontaktováno vedení tohoto projektu a projekt Space Traffic byl uznán za vhodný k propagaci tímto způsobem. První rozhovory jsou naplánovány na září 2016.

Přednášky na středních školách

Vzhledem k tomu, že projekt má svou první funkční demoverzi, je nyní možné oslovit střední školy a propagovat projekt přímo tam. Také bude možné díky dokončení zadání [18] implementovat nové minihry, což by mohlo být zajímavou příležitostí pro vyhlášení soutěže pro nadané středoškolské studenty.

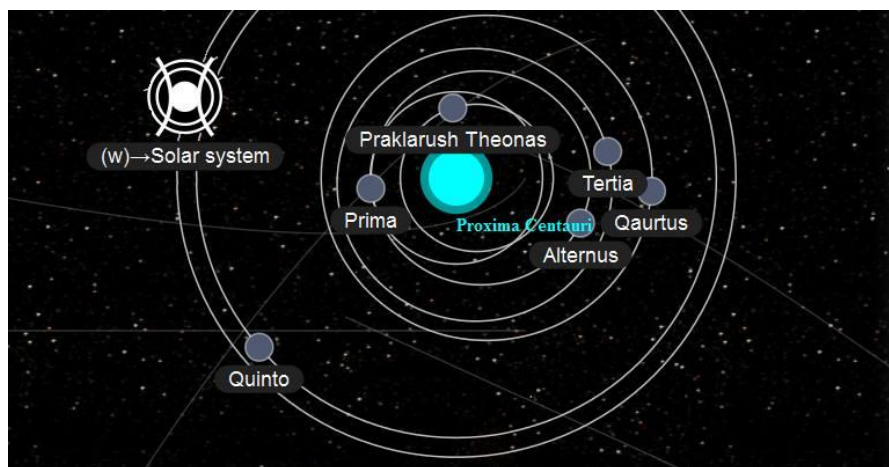
5.15 Hlavní zavedené mechanismy

Závěrem této kapitoly je uvedeno shrnutí všech mechanismů, které byly pro projekt v rámci této diplomové práce zavedeny.

- Převedení projektu na GitHub.
- Konsolidace wiki a vytvoření přehledné znalostní báze.
- Zavedení pravidel tvoření podprojektu v Redmine.
- Vytváření pravidel a konvencí projektu.
- Vylepšení procesu o kontroly shora.
- Zavedení samokontroly pomocí kontrolních seznamů.
- Komunikace přes Facebook.
- Zadávání výstupem.
- Testy nových členů projektu.
- Hodnocení semestrálních prací.

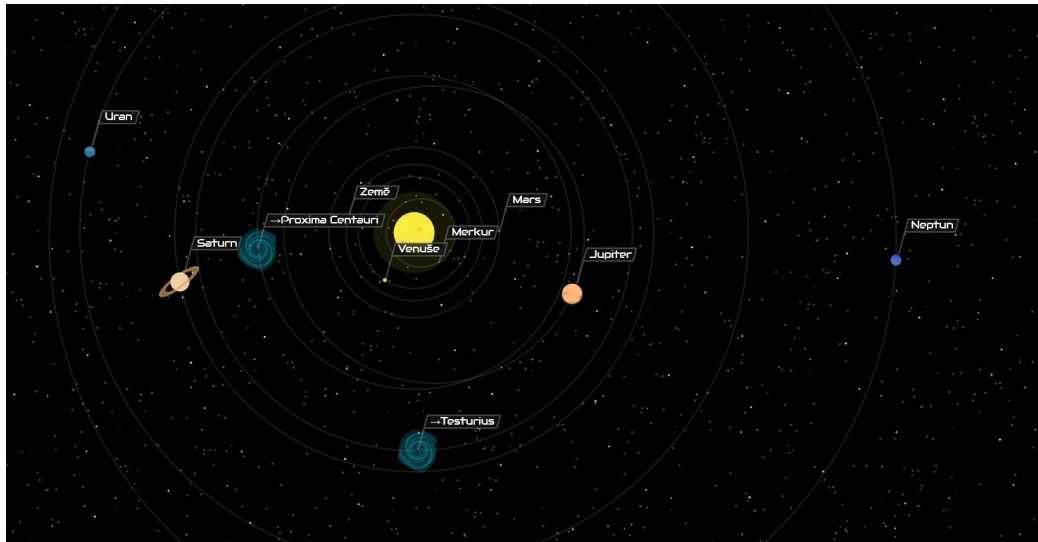
5.16 Viditelná změna

Pro dokreslení představy, jak hra vypadá, jsou zde přiloženy obrázky. Jak hra vypadala v době jejího převzetí ukazuje Obrázek 5.3: Vzhled hry při jejím převzetí



Obrázek 5.3: Vzhled hry při jejím převzetí

Druhý Obrázek 5.4 ukazuje vzhled hry po dvouletém vývoji a vedení projektu.



Obrázek 5.4 Vzhled hry při jejím odevzdávání

O dalších implementačních změnách pojednává práce Jana Kotalíka [16] a Martina Hrona [18].

6. Zhodnocení zkušeností a doporučení pro další vývoj

Následující kapitola je věnována zhodnocení zavedených mechanismů a popisu změn, které tyto mechanismy přinesly. Hlavními cíli této práce bylo navrhnout a popsat pravidla, postupy a nástroje, které budou sloužit pro ověřování postupu prací zejména s ohledem na kvalitu jejich mezivýsledků. Dále bylo nutné vytvořit mechanismy pro archivování a předávání znalostí a zkušeností souvisejících s vedením projektu, s cílem zajistit dlouhodobou kvalitu jeho výstupů. Navržené postupy a mechanismy byly v druhém roce vedení prakticky vyzkoušeny a zdokumentovány. Tato kapitola je shrnutím jejich dopadů a zhodnocením dosažených cílů.

Na konci této kapitoly budou navržena doporučení pro příští vedení projektu.

Změny, které v projektu byly zavedeny, mají vliv na celkové zlepšení vývojového procesu projektu Space Traffic. Díky zavedeným mechanismům je možné lépe kontrolovat průběh celého projektu. Zavedené změny a jejich dopady jsou následující:

- Díky převedení projektu na GitHub je správa verzí přehlednější.
- Konsolidací několika wiki a uložení informací na jedno místo bylo dosaženo jednotné a přehledně členěné znalostní báze projektu. Všechny informace pro developery jsou na jednom místě.
- Sekce pro vedoucí projektu umožňuje uchovávání znalostí a potřebných dokumentů o vedení ve wiki, ale s omezeným přístupem.
- Zavedení pravidel pro vytváření podprojektů v Redmine zajišťuje jednotnost vytváření podprojektů a lepší orientaci v podprojektech.
- Zavedení konvencí projektu zajišťuje jednotná pravidla pro všechny vývojáře projektu.
- Pomocí dotazníků pro přijímání nových členů je možné ihned určit schopnosti týmu.
- Vedoucí je kontrolován, zda je jeho plán dobrý a zda dosáhl slíbených výsledků pomocí kontrol shora od mentora projektu.
- Členové týmu mohou kontrolovat kvalitu odvedené práce podle kritérií kontrolních seznamů, tuto kvalitu si členové projektu kontrolují sami.
- Díky kontrolním seznamům má vedoucí projektu lepší přehled o tom, jak kvalitně byly splněny vypsání úkoly.
- Studenti jsou nuceni dělat menší úkoly a přebírají odpovědnost za svou práci.

- Dotazníky komunikace v průběhu práce umožňují odhalit problémy v týmu.
- Facebooková skupina Space Traffic_developers má reakční čas do půl hodiny.
- Skype s původními tvůrci zajišťuje získávání potřebných informací.
- Zadávání výstupem zajišťuje lepší shrnutí požadavků vedoucí na semestrální práci.
- Hodnocení semestrálních prací umožňuje zjistit, jak dobře které zadání semestrální práce dopadlo.

6.1 Hodnocení vedení projektu členy týmu

V průběhu druhého roku byl kromě již výše zmíněných mechanismů zaveden na žádost mentora projektu mechanismus vyhodnocení spolupráce s vedoucími projektu. Vyhodnocení spolupráce proběhlo pomocí dotazníku, který je součástí H a který byl dán týmu Th3 developers při odevzdávání jejich práce. Jde o nezávislý pohled na spolupráci v rámci projektu. Členové týmu zde hodnotí spolupráci s vedením projektu a zavedené kontrolní seznamy především z hlediska jejich užitečnosti.

Ohlasy týmu na kontrolní seznamy jsou velice dobré. Jeden člen týmu doplnil nápady, jak by bylo možné v rámci dalších prací tyto seznamy vylepšit. Všichni členové týmu uvedli, že je seznamy z počátku zdržovaly, ale když si na ně zvyknuli, pomohli jim zkontrolovat si vlastní práci.

Spolupráce s vedením projektu byla vyhodnocena jako nadprůměrně dobrá. Úkoly byly zadávány jasně a srozumitelně a vedení reagovalo na všechny nejasnosti téměř okamžitě. Odpovědi těchto dotazníků jsou součástí přílohy I.

6.2 Zhodnocení dosažených cílů

Na základě zjištěných skutečností byla navržena a zavedena vylepšení, která povedou především k hladkému předávání projektu Space Traffic mezi studenty a ke zlepšení předávání znalostí o projektu. Pro vedoucí byla vytvořena speciální sekce wiki s přístupem pouze pro vedoucí projektu. Zde je pro nového vedoucího připravený podrobně zpracovaný návod, jak s projektem Space Traffic začít.

Odpovědnost za projekt je delegována na členy týmu pomocí kontrolních seznamů, tzv. checklistů. Checklisty byly začleněny do konvencí projektu. Díky nim jsou také v Redmine projektu plánovány kratší úkoly, protože ke kratšímu úkolu se váže také kratší kontrolní seznam. Dělení projektu na menší úkoly lze pozorovat na Redmine projektu porovnáním projektů z prvního a druhého roku vedení. Kontrolní seznamy také umožňují kontrolovat kvalitu práce, kterou tým odvedl.

Zadávání práce studentům formou výstupu bylo ověřeno na zadání studentů předmětu KIV/ZSWI v průběhu letního semestru 2016. Tento způsob zadání pomohl lépe kontrolovat hotové části implementace během semestru a lépe se ověřovala funkčnost celého řešení v závěru semestru. Zadání výstupem bylo ve výsledku lépe kontrolovatelné, než do té doby používané zadání v bodech. Hlavním přínosem tohoto zadávání bylo, že jsme dostali funkční část aplikace v požadované podobě.

Testy pro přijímání nových členů projektu pomohly hlavně v první fázi letního semestru, kdy se o stejné zadání přihlásily dva různé týmy studentů předmětu KIV/ZSWI. Díky těmto testům bylo možné hned na začátku práce odhadnout schopnost členů týmu a vybrat správný tým pro implementaci požadovaného zadání.

Pro nastupující vedoucí projektu je doporučeno dodržovat nastavená pravidla projektu, obzvláště pak v prvotní fázi projektu. V případě, že se některé z pravidel zdá zbytečné, nejdřív prosím zkoumejte jeho význam. Důležité je nepodcenit fázi přípravy projektu, jak uvedl ve své práci [1] Petr Vogl.

6.3 Hlavní doporučení pro úspěšné vedení projektu

- Je bezpodmínečně nutné udržovat wiki projektu aktuální a jednotnou. Všechny změny projektu je nutné zanést i do wiki. Stránky ve wiki je nutné upravovat a ne vytvářet stále nové, nebo vytvářet pouze záznamy o změnách.
- Je nutné dodržovat konvence projektu a zavedené mechanismy.
- Zadání semestrálních prací pomocí výstupu usnadní v mnohém práci ve fázi závěrečného odevzdání projektu. Součástí každého zadání musí být testování, dokumentace na wiki projektu a požadavek na dodržování konvencí projektu.
- Komunikací během schůzek udržujeme kontakt s projektovými týmy a zjišťujeme jejich porozumění zadání, které v případě nepochopení upřesníme. Je nutné průběh prací pravidelně kontrolovat a nedovolit u žádné z prací vznik metodou “Velkého třesku“. Také je nutné kontrolovat způsob provedení a dodržování projektových konvencí.

- Na základě tvrzení v [1] je doporučeno nepodcenit přípravnou fázi projektu v průběhu prázdnin. Průběh této fáze navíc bude zkontrolován mentorem projektu při dosažení prvního milníku.
- Pokud si vedení není jisté řešením nějakého problému, je nejlepší jít rovnou za mentorem projektu. Zkušenosti ukázaly, že problém, který byl řešen skoro měsíc, mentor projektu vyřešil na místě za pět minut.
- Je lepší zadat menší část funkcionality, která je otestována a zdokumentována, než když vznikne velká práce bez dokumentace a tím možnosti jejího dalšího použití.
- Ve všech knihách a příručkách projektového manažera se píše: Pokud má projekt problémy, informujte nadřízené, v tomto případě mentora projektu. Pokud je totiž vedení o problémech projektu informováno předem, není pak překvapeno jeho neúspěchem. Pokud ovšem problémy projektu skrýváte, nebo skrýváte jiného studenta, vrátí se Vám to při obhajobě Vaší vlastní práce.

7. Závěr

V průběhu práce jsem se naučila, jak lépe analyzovat projekty a procesy. Také jsem se naučila, jak funguje projektové řízení v praxi.

Na základě zjištěných skutečností byla navržena a zavedena vylepšení, která povedou především k hladkému předávání projektu Space Traffic mezi studenty a ke zlepšení předávání znalostí o projektu. Vybrané metody pro podporu projektu jsou zavedené a výsledky jsou zdokumentované. Projekt je připraven k předání dalšímu vedoucímu. Jsou stanovené milníky projektu a nad proces vývoje projektu byl přidán kontrolní proces, který kontroluje dosahování jednotlivých milníků. Je vytvořená přehledná a podrobná znalostní báze se sekcí pro vedoucí projektu. Odpovědnost za projekt je delegována na členy týmu pomocí kontrolních seznamů, tzv. checklistů. Také byla vytvořena doporučení pro další vedení projektu. Byl stanoven nástroj pro komunikaci v projektu. Jsou zavedené konvence projektu a práce byly zadávány formou výstupu, což se osvědčilo. Také byly zavedeny testy pro přijímání nových členů projektu.

Všechny uvedené mechanismy byly prakticky vyzkoušeny během druhého roku vedení projektu a významným způsobem usnadnily průběh projektu oproti prvnímu roku.

Množstvím schůzek a komunikací s lidmi ve firmách jsem získávala potřebné informace o fungování projektů a získané postupy jsem mohla uplatnit v projektu Space Traffic. Byla to pro mě velmi přínosná a zajímavá práce. Práce poslouží budoucímu vedení projektu k získání informací o tom, jak projekt vést a k jakým situacím může v průběhu projektu doházet. Pokud budou v rámci následujících let dodržována implementovaná opatření, projekt bude pro další vedení lépe uchopitelný a pro nové týmy bude mnohem snadnější se do projektu začlenit.

Na základě výše uvedených údajů jsem přesvědčena, že zadání bylo splněno.

Přehled zkratk

· <i>autobuild</i>	-	<i>automatické sestavení</i>
· <i>autotest</i>	-	<i>automatické testování aplikace</i>
· <i>backlog</i>	-	<i>množina úkolů, které teprve budou naplánované a jsou nutné pro rozpočet</i>
· <i>budget</i>	-	<i>rozpočet</i>
· <i>benefit</i>	-	<i>výhoda, prospěch, užitek</i>
· <i>best practices</i>	-	<i>posbírané nejlepší praktiky z praxe</i>
· <i>branch</i>	-	<i>oddělení nové větve v systému pro správu verzí Git</i>
· <i>bug</i>	-	<i>chyba v programu</i>
· <i>burndown diagram</i>	-	<i>grafické znázornění práce, kterou zbývá udělat v čase</i>
· <i>cashflow</i>	-	<i>peněžní tok</i>
· <i>cloud</i>	-	<i>síť serverů, která slouží k ukládání dat</i>
· <i>commit</i>	-	<i>zaznamenání změn, potvrzení změn, přidání změn</i>
· <i>contend</i>	-	<i>obsah</i>
· <i>cyklus</i>	-	<i>opakující se činnost</i>
· <i>daily standup meeting</i>	-	<i>denní schůzka definovaná dle metodiky Scrum</i>
· <i>dashboard</i>	-	<i>nástěnka, tabule</i>
· <i>data warehouse</i>	-	<i>datový sklad</i>
· <i>design</i>	-	<i>návrh vzhledu</i>
· <i>documentation first</i>	-	<i>nejdříve je postup práce popsán v dokumentaci, vývoj je pak veden podle této dokumentace</i>
· <i>Engine</i>	-	<i>jádro</i>
· <i>enterprise architecture</i>	-	<i>podniková architektura</i>
· <i>epic story</i>	-	<i>nejvyšší množina úkolů projektu, skládá se z jednotlivých story</i>
· <i>Facebook</i>	-	<i>sociální síť</i>
· <i>feature</i>	-	<i>vlastnost</i>
· <i>funkcionalita</i>	-	<i>souhrn všech funkcí, které zaručují funkčnost</i>
· <i>Git</i>	-	<i>systém pro správu verzí</i>
· <i>Github</i>	-	<i>webová služba podporující verzovací nástroj Git</i>
· <i>goal</i>	-	<i>výsledek, kterého se snažíme dosáhnout</i>
· <i>hardware</i>	-	<i>fyzické vybavení počítače</i>
· <i>help desk</i>	-	<i>systém pomoci zákazníkovi</i>
· <i>hot line</i>	-	<i>linka pro pomoc zákazníkům</i>
· <i>check list</i>	-	<i>kontrolní seznam</i>
· <i>IDEF0</i>	-	<i>slouží ke specifikaci funkčních modelů podniku</i>
· <i>IPMA</i>	-	<i>certifikace projektového manažera</i>
· <i>inkrement</i>	-	<i>přírůstek</i>
· <i>issue</i>	-	<i>požadavek, většinou uživatelský</i>
· <i>ITIL</i>	-	<i>soubor praxí a konceptů pro zlepšení kvality služeb</i>
· <i>JIRA</i>	-	<i>softwarový nástroj pro evidenci chyb a problémů</i>
· <i>know how</i>	-	<i>znalost postupů a řešení určité problematiky (vědět jak)</i>
· <i>Kobe (levels security)</i>	-	<i>stupně zabezpečení aplikace</i>
· <i>konfigurace</i>	-	<i>požadované nastavení</i>
· <i>KPI</i>	-	<i>klíčové ukazatele výkonnosti (key performance indicators)</i>
· <i>log</i>	-	<i>zaznamenávání změn</i>
· <i>manager</i>	-	<i>manažer</i>
· <i>mentor</i>	-	<i>rádce, zkušenější člověk</i>
· <i>metrika</i>	-	<i>měřitelný údaj</i>
· <i>milník</i>	-	<i>dosažení předem určené události v průběhu projektu</i>

· <i>on budget</i>	- <i>v rozpočtu, projekt nesmí přesáhnout svůj rozpočet</i>
· <i>on time</i>	- <i>včas, na čas</i>
· <i>on premise</i>	- <i>lokální řešení</i>
· <i>PDM manager</i>	- <i>manažer vývoje, (product development manager)</i>
· <i>Prince2</i>	- <i>metodika projektového řízení</i>
· <i>priority</i>	- <i>důležitost, přednost, priorita</i>
· <i>proces</i>	- <i>přeměna vstupu na výstupy za pomoci zdrojů</i>
· <i>product owner</i>	- <i>vlastník produktu</i>
· <i>product road map</i>	- <i>mapa celkového vývoje produktu</i>
· <i>progress</i>	- <i>pokrok, posun projektu</i>
· <i>projekt</i>	- <i>proces s ohraničeným koncem a začátkem</i>
· <i>projektový cíl</i>	- <i>čeho má projekt dosáhnout</i>
· <i>prototypování</i>	- <i>vývoj systému způsobem schvalování prototypů</i>
· <i>Redmine</i>	- <i>softwarový nástroj pro evidenci úkolů projektu</i>
· <i>release</i>	- <i>verze produktu určená k vydání</i>
· <i>retrospektivy</i>	- <i>vyhodnocení předešlé iterace</i>
· <i>review manager</i>	- <i>manažer změn</i>
· <i>risk list</i>	- <i>seznam rizik</i>
· <i>RUP</i>	- <i>metodika vývoje software (Rational Unified Process)</i>
· <i>Samepage</i>	- <i>systém pro správu obsahu</i>
· <i>scope diagram</i>	- <i>diagram rozsahu</i>
· <i>Scrum</i>	- <i>metodika agilního vývoje software</i>
· <i>severity</i>	- <i>vážnost</i>
· <i>Share point</i>	- <i>systém pro správu informací</i>
· <i>software</i>	- <i>programové vybavení počítače</i>
· <i>story</i>	- <i>příběhy, do kterých se člení úkoly (task)</i>
· <i>SVN</i>	- <i>systém pro správu zdrojových kódů</i>
· <i>target</i>	- <i>cíl</i>
· <i>task</i>	- <i>úkol v systému pro zadávání a správu úkolů</i>
· <i>Teamcity</i>	- <i>systém pro monitorování sestavení projektu</i>
· <i>template</i>	- <i>šablona</i>
· <i>tester</i>	- <i>pracovník, který testuje aplikaci</i>
· <i>unit testy</i>	- <i>jednotkové testy</i>
· <i>větev Master</i>	- <i>hlavní vývojová větev v systému pro správu verzí</i>
· <i>větev Release</i>	- <i>větev v systému pro správu verzí určená pro verzi k vydání</i>
· <i>vize produktu</i>	- <i>význam produktu, kam se má produkt ubírat</i>
· <i>web</i>	- <i>počítačová síť</i>
· <i>wiki projektu</i>	- <i>místo, kde jsou přehledně uloženy všechny informace o vývoji projektu</i>

Reference

1. **Vogl, Petr.** *Podpora vývoje webových hry pro více hráčů.* Plzeň, 2012, Diplomová práce na Fakultě Aplikovaných věd Západočeské univerzity v Plzni na katedře informatiky a výpočetní techniky. Vedoucí diplomové práce : doc. Ing. Přemysl BRADA, MSc. PhD. .
2. **Schwalbe, Kathy.** *Řízení projektů v IT.* Brno : Computer Press, 2007. ISBN 978-80-251-1526-8.
3. **Rosenau, Milton D.** *Řízení projektů.* Brno : Computer Press, 2000. ISBN 978-80-251-1506-0.
4. **Skalický Jiří, Jermář Milan, Svoboda Jaroslav.** *Projektový management a potřebné kompetence.* Plzeň : Západočeská univerzita v Plzni, 2010. ISBN 978-80-7043-975-3.
5. **Řepa, Václav.** *Podnikové procesy, procesní řízení a modelování.* Praha : Grada Publishing, 2007. ISBN 987-80-247-2252-8.
6. **Mládková, Ludmila.** *Management znalostí v praxi.* Praha : Profesional Publishing, 2004. ISBN:80-86419-51-7.
7. **Němec, Vladimír.** *Projektový management.* Praha : Graga Publishing, 2002. ISBN 80-247-0392-0.
8. **Meredith, Jack, R. a Mantel, Samuel, J.** *Project management A Managerial Approach.* Singapore : John Wiley, 2012. ISBN 987-1-118-09373-3.
9. **Štefánek, Radoslav, a další.** *Projektové řízení pro začátečníky.* Brno : Computer Press, 2011. ISBN 978-80-251-2835-0.
10. **Fiala Josef, Ministr Jan.** *Průvodce analýzou a modelováním procesů.* Ostrava : VŠB-Technická univerzita Ostrava, 2003. ISBN 20-248-0500-6.

11. **kol., Rolínek Ladislav a.** *Procesní management vybrané aspekty*. České Budějovice : Jihočeská univerzita v Českých Budějovicích, 2008. ISBN 978-80-7394-148-2.
12. **Wesley, Addison.** *Softwarové inženýrství*. Brno : Comuter Press, 2013. ISBN 978-80-251-3826-7.
13. **Krol, Petr a Kruchten, Philippe.** *The Rational Unified Process Made Easy*. Boston : Pearson Education, 2003. ISBN 0-321-16609-4.
14. **Beck, Kent.** *Extreme Programming explained*. Boston : Addison-Wesley, 2000. ISBN 0-201-61641-6.
15. **Štěpánek, Martin.** *Architektura a implementace webové hry pro více hráčů*. Plzeň, 2012, Diplomová práce na Fakultě Aplikovaných věd Západočeské univerzity v Plzni na katedře informatiky a výpočetní techniky. Vedoucí diplomové práce : doc. Ing. Přemysl BRADA, MSc. PhD. .
16. **Kotalík, Jan.** *Návrh a implementace uživatelského rozhraní pro hru Space Traffic*. Plzeň, 2016, Diplomová práce na Fakultě Aplikovaných věd Západočeské univerzity v Plzni na katedře informatiky a výpočetní techniky. Vedoucí diplomové práce : Ing. Petr VANĚČEK, Ph.D. .
17. **Vývojový tým Space Traffic.** Vývojářská wiki projektu space Traffic. *spacetraffic.kiv.zcu.cz/code/*. [Online]
18. **Hron, Martin.** *Návrh a implementace miniher pro hru Space Traffic*. Plzeň, 2016, Diplomová práce na Fakultě Aplikovaných věd Západočeské univerzity v Plzni na katedře informatiky a výpočetní techniky. Vedoucí diplomové práce : Ing. Petr VANĚČEK, Ph.D. .

Seznam obrázků

Obrázek 2.1: Projektový trojimperativ (vlastní zpracování)	4
Obrázek 2.2: Maslowova pyramida potřeb (vlastní zpracování)	7
Obrázek 2.3: Vodopádový model (vlastní zpracování)	10
Obrázek 2.4: Spirálový model (vlastní zpracování)	11
Obrázek 2.5: RUP (vlastní zpracování)	12
Obrázek 2.6: Scrum (vlastní zpracování)	14
Obrázek 3.1: Ideální testovací pyramida (poskytnut firmou)	18
Obrázek 3.2: Počet chyb (poskytnut firmou).....	20
Obrázek 3.3: Počet anotovaných a smazaných řádek kódu v průběhu projektu (poskytnut firmou)	21
Obrázek 3.4: Iterativní vývoj v AIMTECu (poskytnut firmou)	23
Obrázek 3.5: SMART (poskytnut firmou).....	23
Obrázek 3.6 Tabule s úkoly (poskytnut firmou).....	24
Obrázek 3.7: Burndown diagram(poskytnut firmou)	25
Obrázek 3.8: Termíny projektu (poskytnut firmou)	37
Obrázek 3.9: Rizika (poskytnut firmou)	40
Obrázek 4.1: Fáze vývojového cyklu během akademického roku podle Petra Vogla (přebrán z Diplomové práce Petra Vogla [1])	52
Obrázek 4.2: Činnosti prováděné během vývojového cyklu Petra Vogla (přebrán z Diplomové práce Petra Vogla [1])	53
Obrázek 5.1: Verzovací systém Git v aplikaci Source Tree (printscrenn)	61
Obrázek 5.2: Nový proces s kontrolní úrovní(vlastní zpracování).....	63
Obrázek 5.3: Vzhled hry při jejím převzetí (printscrenn).....	71
Obrázek 5.4 Vzhled hry při jejím odevzdávání (printscrenn)	72

Přílohy

A. Kontrolní seznamy

Wiki stránka projektu, součást konvencí projektu (viditelná kýmkoliv)

Z důvodu zvyšování kvality výstupů vaší práce je nutné používat tyto kontrolní seznamy.

S jejich pomocí si ověříte, zda jste při práci nic neopomenuli. Vyplněním této sady kontrolních dotazů také dáváte vedoucímu najevo, že přebíráte odpovědnost za Vámi upravenou nebo vytvořenou část projektu.

Z následujících formulářů si vyberete správný podle toho, zda se jedná o implementační, testovací, nebo dokumentační úkol. U implementačních úkolů pak dále záleží na skutečné době trvání úkolu. (ne naplánované) K implementačním a testovacím úkolům seznamy vkládáme do komentáře v Redmine, u dokumentačních úkolů vkládáme seznam pod čáru přímo do stránky.

Na otázky 1 a 2 odpovíte větou, na ostatní otázky odpovíte pouze čísly 1 (naprosto souhlasí) až 5 (naprosto nesouhlasí)
V případě, že otázku nelze odpovědět, napište X.

Implementační úkol do 30 min

1. Číslo mergovacího commitu:
2. K čemu tento úkol slouží, proč byl zadán?
3. Práce je odvedena kvalitně a dobře.

Implementační úkol 30 min - 2h

1. Číslo mergovacího commitu:
2. K čemu tento úkol slouží, proč byl zadán?
3. K tomuto úkolu je vypsán příslušný úkol na testování.
4. K této dílčí části práce existuje potřebná dokumentace na wiki projektu, nebo je vypsán úkol na dokumentaci.
5. Zkontroloval jsem si po sobě commitlé změny.
6. Všechny commit jsou messages vyplněné.
7. Kód je komentován podle konvencí projektu.
8. Odhad času na úkol nebyl příliš rozdílný oproti skutečnosti.
9. Zalogovaný čas je skutečný čas strávený nad úkolem. (není nadhodnocený/podhodnocený)
10. Práce je odvedena kvalitně a dobře.

Implementační úkol nad 2h

1. Číslo mergovacího commitu:
2. K čemu tento úkol slouží, proč byl zadán?
3. K tomuto úkolu je vypsán příslušný úkol na testování.
4. K této dílčí části práce existuje potřebná dokumentace na wiki projektu, nebo je vypsán úkol na dokumentaci.
5. Zkontroloval jsem si po sobě commitlé změny.
6. Všechny commit jsou messages vyplněné.
7. Kód je komentován podle konvencí projektu.
8. Odhad času na úkol nebyl příliš rozdílný oproti skutečnosti.
9. Zalogovaný čas je skutečný čas strávený nad úkolem.(není nadhodnocený/podhodnocený)
10. Zamyslel jsem se nad tím, zda je řešení ideální.
11. Pokud jsem měl nápad, jak řešení vylepšit, informoval jsem svého vedoucího.
12. Splnění úkolu šlo podle plánu a neměl jsem s ním problémy.
13. Nevyskytly se nečekané okolnosti, které mě donutili přerušit práci a odložit jí na jindy.
14. Nevím o žádných problémech, které bych nenahlásil vedoucímu projektu.
15. Práce je odvedena kvalitně a dobře.

Testování

1. Číslo mergovacího commitu:
2. K čemu tento úkol slouží, proč byl zadán?
3. Testování proběhlo podle konvencí projektu.
4. Práce je odvedena kvalitně a dobře.

Dokumentace (na wiki projektu)

1. Je dokumentace dostatečně podrobná a kompletní, co byste doplnili?
2. K čemu tato dokumentace slouží, proč vznikla?
3. Stránka je naformátovaná dle konvencí projektu.
4. Hledal jsem stránky s podobnou tematikou a ujistil jsem se, že neexistuje taková, kterou bych mohl pouze upravit.
5. Dokumentace je provedena kvalitně a pečlivě.
6. Dokumentaci jsem si po sobě přečetl a dokázal bych podle ní zasáhnout do projektu.
7. Ujistil jsem se, že na mnou vytvořenou stránku je v menu odkaz.
8. Ujistil jsem se, že jsem své stránce přidal tagy a kategorii.
9. Prošel jsem všechny existující stránky a na vhodná místa dal odkaz na novou stránku.
10. Vytvořil jsem číslované shrnutí v bodech, jak funkcionalitu použít.

11. Vytvořil jsem číslované shrnutí v bodech, jak funkcionalita funguje.

Shrnutí:

Př1 implementační úkol:

1. Dokončím implementační úkol na redmine.
2. Zkontroluji celkový čas úkolu.
3. Podle času vyberu variantu kontrolního seznamu.
4. Na začátek kontrolního seznamu přidám <Kontrolní seznam>.
5. Na konec kontrolního seznamu přidám </Kontrolní seznam>
6. Odpovím na jednotlivé otázky (doporučuji zkopírovat a nahradit odpovědí)

Př2 dokumentace:

1. Dokončím svou stránku ve wiki.
2. Pokud jsem pouze upravoval stávající, upravím i kontrolní seznam.
3. Pokud jsem vytvořil novou stránku, vytvořím i kontrolní seznam.
4. Ve stránce udělám svislou čáru.
5. Na začátek kontrolního seznamu přidám <Kontrolní seznam>.
6. Na konec kontrolního seznamu přidám </Kontrolní seznam>
7. Odpovím na jednotlivé otázky (doporučuji zkopírovat a nahradit odpovědí)

B. Přijímání nového člena do projektu

Wiki stránka projektu, součást sekce pro vedoucí projektu (viditelná pouze pro vedoucí)

Dlouho jsme si lámali hlavu s tím, jak přijímat nové členy týmu. První rok našeho vedení jsme začali většinou rovnou zadáním bez zkoumání, zda úroveň zadání vůbec odpovídá úrovni nového týmu.

Další rok jsme vytvořili sadu otázek, které jsou určeny k pohovoru s týmem, nebo jednotlivcem, který přijímá zadání v projektu. Vedoucímu projektu tyto otázky pomohou odhalit možná rizika nebo nedostatky. Novému členovi týmu by pak měly pomoci k určení oblastí, kde by měl ještě doplnit informace. V případě nedostatečných znalostí celého týmu je možné udělat školení.

Tento pohovor nám výrazně pomohl k utvoření si obrázku o schopnostech týmu a pomohl nám vybrat nejlepšího zájemce v případě více zájemců o stejné zadání.

Studenti bakalářského studia

Práce v týmu

- Jaká je struktura pracovního týmu? (členové, role) Popište Váš tým.
- Jak je rozdělena odpovědnost v projektu? (celý projekt, části projektu, týmy...)
- Jak a kdo plánuje jednotlivé úkoly a průběh celého projektu? (A kontroluje jednotlivé úkoly?)
- Jak v týmu komunikovat?
- Co byste udělali v situaci, kdyby se jeden člen teamu vyhýbal zadaným úkolům a přesouval by je na ostatní?
- Co uděláte, když Vás kolega požádá o pomoc při něčem, co není Vaší pracovní náplní?

O projektu

- Co to je projekt Space Traffic, čím se zabývá?
- K čemu v projektu potřebujeme dokumentaci?
- Proč projekt testujeme? Proč píšeme automatické testy?
- Máte představu, proč může projekt skončit neúspěšně?
- Jaký je cíl projektu?
- Jaké jsou požadavky na zadavatele projektu?
- Učíte se rádi nové věci?
- Čím nám můžete (Váš tým) přispět? Proč máme vybrat právě vás?
- Co by se ještě hodilo k vašemu zadání? (čím byste zadání vylepšili)
- Kdybyste se měl porovnat se svými současnými kolegy, v čem jste lepší/v čem horší, kde byste měl/a na sobě více zapracovat?

Technické

- Jaký je rozdíl mezi private, public a protected metodou?
- Jaký je rozdíl mezi private, public a protected parametrem?
- K čemu slouží getter a setter?
- Znáte návrhové vzory? K čemu slouží návrhový vzor přepravka? (box)
- Jaká je podle Vás dobrá délka metody? (počet řádek)
- Jaká je podle Vás dobrá délka třídy? (počet řádek)
- Víte, jakým způsobem se ukládají hesla v databázi a proč?
- Co to je .NET? Programovali jste někdy v .NET?
- Co to je MVC architektura a jak funguje?
- Víte, jak funguje http komunikace –požadavek a odpověď?
- K čemu používáme verzovací nástroje? (řešili jste někdy konflikty?)

Studenti navazujícího studia

- Předchozí otázky

O projektu

- K čemu je v projektu potřebujeme znalostní bázi?
- Jak zajišťujeme v projektu kvalitu dodaného produktu?
- Jaká jsou kritéria úspěšnosti Vašeho projektu?
- Jaká jsou rizika projektu?

Technické

- MVC architektura podrobněji.
- Je dobrý nápad pushovat do mastera?
- Co vám říká GIT/Github?
- Jak byste správně pracovali s dokumentací v projektu?
- Jak si myslíte, že funguje hra? (nekonečná smyčka na serveru, která zajišťuje simulaci, počítá s časem, řeší, jak se mění svět)
- Jak fungují ajaxové požadavky a k čemu se v projektu používají?
- Co to je ORM? (mapování řádků tabulky na objekty)
- Jak komentujeme kód?
- Jak si myslíte, že funguje virtuální server projektu? (OS, technologie, jestli jim dojde, že když je to .net tak že to je Microsoft server...)
- Co uděláte, když vám chybí součást implementace, kterou potřebujete, ale není hotová a je nad rámec vaší práce. (NE TO DO, nebo to alespoň někomu říct, udělat rozhraní,...)
- Jaké webové technologie umíte používat?
- Co uděláte, když ve vašem, nebo cizím kódu najdete zakomentované bloky?

C. Analýza firemních projektů

Okruhy otázek, které byly použité k analýze v navštívených firmách.

Obecné informace o projektu:
1. Čím se projekt zabývá?
2. Studie proveditelnosti, finanční plán a analýza projektu.
3. V jaké fázi se projekt nachází? (zahajovací, střední, závěrečná, milníky-mezníky projektu)
4. Jaké máte vazby na prostředí? Širší rámec, do kterého projekt zapadá. (s čím projekt souvisí)
5. Podniková metodika (Jednotné normy pro konkrétní projekt, nebo pro všechny projekty společnosti)
6. Jak a kdo rozhodne, že je produkt možné nasadit? (Jak se produkt dostane z vývojové větve na produkci?)
Cíle a plán projektu:
7. Jaké jsou cíle projektu?
8. Jak měříte mezicíle projektu (Jak zjistíte, jak si 'vedete'?)
9. Jaká jsou kritéria úspěšného ukončení projektu?
10. Jaký je plán projektu? (zdroje, čas, náklady...)
Zdroje projektu:
11. Jaké jsou požadavky na zdroje projektu?
Lidé a komunikace v projektu:
12. Jaké jsou požadavky na členy pracovního týmu?
13. Jaká je struktura pracovního týmu? (členové, role)
14. Jaké jsou požadavky na zadavatele projektu?
15. Funguje projektová komunikace? (a jak?)
16. Jak řešíte spory v týmu?
17. Jak a kdo plánuje jednotlivé úkoly a průběh celého projektu? (A kontroluje jednotlivé úkoly?)
18. Odpovědnost v projektu. (celý projekt, části projektu, týmy...)
Nástroje projektu:
19. Jaké nástroje používáte pro komunikaci v týmu. Jak?
20. Jak probíhá zálohování projektu?
21. Jak pracujete s dokumentací? (ukládání, práce s ní) Vytváříte znalostní bázi?
Měření projektu:
22. Jaká jsou omezení projektu? (finanční, časová, ..)
23. Jaká jsou kritéria úspěšnosti projektu?
24. Jak identifikujete a hodnotíte rizika projektu?
25. Jak zajišťujete kvalitu projektu? (Meziproduktů)
26. Jak probíhá měření projektu - zpětná vazba? (náklady, pracnost, doba trvání, produktivita, kvalita výstupů, spokojenost zadavatele s dodávkou)
27. Zpětná vazba. (metriky, náklady na měření)

D. Analýza projektu Space Traffic při přebírání projektu

Analýza projektu Space Traffic stejnými okruhy otázek, jako v navštívených firmách

Čím se projekt zabývá?	Jde o webovou hru naučného charakteru, která má sloužit k propagaci Katedry informatiky a výpočetní techniky ZČU Plzeň. Dále tento projekt umožňuje studentům vyzkoušet si práci na velkém projektu v rámci semestrálních a klasifikačních prací.
Studie proveditelnosti, finanční plán a analýza projektu.	Tyto informace jsou roztržštěné a momentálně nedohádatelné.
V jaké fázi se projekt nachází? (zahajovací, střední, závěrečná, milníky-mezníky projektu)	Projekt ST se nachází ve střední fázi. (IOC-stále probíhá implementace-construction), beta verze zatím neexistuje.
Jaké máte vazby na prostředí? Širší rámec, do kterého projekt zapadá. (s čím projekt souvisí)	Projekt není funkčně navázán na jiné prostředí.
Podniková metodika (Jednotné normy pro konkrétní projekt, nebo pro všechny projekty společnosti)	Nelze zodpovědět. Projekt má nejasně definované konvence.
Jak a kdo rozhodne, že je produkt možné nasadit? (Jak se produkt dostane z vývojové větve na produkci?)	O nasazení pravděpodobně rozhodne hlavní programátor projektu, ale zatím nevíme, na který server se bude nová verze nasazovat.
Jaké jsou cíle projektu?	Dlouhodobým cílem projektu je vytvořit hratelnou verzi webové hry. Střednědobým cílem je splnit plány na tento akademický rok. Krátkodobým cílem je připravit projekt na den otevřených dveří katedry do prezentovatelné podoby.
Jak měříte mezicíle projektu (Jak zjistíte, jak si 'vedete'?)	Mezicíle v ST jsou pravděpodobně jednotlivá ukončení různých semestrálních prací. Kvalita prací není nikde měřena a není možné dohledat, jak která práce dopadla.
Jaká jsou kritéria úspěšného ukončení projektu?	Hratelná verze hry.
Jaký je plán projektu? (zdroje, čas, náklady...)	Zdroje projektu jsou omezené počtem

	studentů, kteří se rozhodnou k projektu přidat. Čas věnovaný projektu je omezený časem studentů, kteří se projektu věnují v rámci semestrálních prací. V projektu se o finančních nákladech neuvažuje. (řeší se pouze časové zdroje)
Jaké jsou požadavky na zdroje projektu?	Vzhledem k tomu, že jde o studentský projekt, jsou zdroje vyjádřené počtem hodin v rámci semestrálních prací.
Jaké jsou požadavky na členy pracovního týmu?	Požadavky na členy pracovního týmu nejsou blíže specifikované.
Jaká je struktura pracovního týmu? (členové, role)	Struktura pracovního týmu se bude pravděpodobně v průběhu projektu měnit. Nyní je zde jeden vedoucí a hlavní programátor projektu. Také jsou zadána dvě zadání na úpravu znalostní báze projektu.
Jaké jsou požadavky na zadavatele projektu?	Není specifikováno.
Funguje projektová komunikace?	Projektová komunikace funguje zatím dobře, rychle a bez problémů. Pro rychlou komunikaci byla založena skupina na sociální síti Facebook.
Jak řešíte spory v týmu?	Ke sporům zatím nedošlo.
Jak a kdo plánuje jednotlivé úkoly a průběh celého projektu? Kdo kontroluje jednotlivé úkoly?	Jednotlivé úkoly nejsou plánovány, jsou plánována zadání v rámci semestrálních prací.
Odpovědnost v projektu (celý projekt, části projektu, týmy...)	Celý projekt má na starosti vedoucí projektu, který rozděljuje jednotlivé práce.
Jaké nástroje používáte pro komunikaci v týmu. Jak?	Ke komunikaci je používána sociální síť Facebook a byly zavedeny pravidelné osobní schůzky.
Jak probíhá zálohování projektu?	Zatím není zjištěno.
Jak pracujete s dokumentací? (ukládání, práce s ní)	Ano, ale znalostníchází existuje několik,

Vytváříte znalostní bázi?	snažíme se o konsolidaci.
Jaká jsou omezení projektu? (finanční, časová, ..)	Finanční omezení projekt nemá. Je limitován rozsahem semestrálních prací, které zadáváme.
Jaká jsou kritéria úspěšnosti projektu?	Plnění plánů a cílů.
Jak identifikujete a hodnotíte rizika projektu?	Nalezení rizik by mělo být součástí této práce, zatím s nimi nepracujeme.
Jak zajišťujete kvalitu projektu? Meziproduktů	Zatím není zjištěno, pravděpodobně kvalitu projekt neřeší.
Jak probíhá měření projektu - zpětná vazba? (náklady, pracnost, doba trvání, produktivita, kvalita výstupů, spokojenost zadavatele s dodávkou)	Zatím neprobíhá zpětná vazba.
Zpětná vazba (metriky, náklady na měření)	Metriky zatím nejsou určené.

E. Analýza projektu Space Traffic při jeho ukončování

Analýza projektu Space Traffic stejnými okruhy otázek, jako v navštívených firmách

Čím se projekt zabývá?	Jde o webovou hru naučného charakteru, která má sloužit k propagaci Katedry informatiky a výpočetní techniky ZČU Plzeň. Dále tento projekt umožňuje studentům vyzkoušet si práci na velkém projektu v rámci semestrálních a klasifikačních prací.
Studie proveditelnosti, finanční plán a analýza projektu.	Projekt neobsahuje tyto dokumenty, veškeré informace o projektu jsou obsaženy v projektové wiki.
V jaké fázi se projekt nachází? (zahajovací, střední, závěrečná, milníky-mezníky projektu)	Projekt ST má svou první funkční demo verzi. (PR- existuje první demoverze hry)
Jaké máte vazby na prostředí? Širší rámec, do kterého projekt zapadá. (s čím projekt souvisí)	Projekt není funkčně navázán na jiné prostředí.
Podniková metodika (Jednotné normy pro konkrétní projekt, nebo pro všechny projekty společnosti)	Projekt není součástí dalšího prostředí, má jasně definované konvence.
Jak a kdo rozhodne, že je produkt možné nasadit? (Jak se produkt dostane z vývojové větve na produkci?)	O nasazení produktu rozhodne hlavní programátor projektu, který novou stabilní verzi nasadí.
Jaké jsou cíle projektu?	Dlouhodobým cílem projektu je jeho propagace na středních školách. Střednědobým cílem je umožnit programovatelnost lodí. Krátkodobým cílem je připravit projekt pro jeho prezentaci v jeho současném stavu.
Jak měříte mezicíle projektu (Jak zjistíte, jak si 'vedete' ?)	Mezicíle v ST jsou pravděpodobně jednotlivá ukončení různých semestrálních prací. Kvalitu práce hodnotíme závěrečným formulářem, podle kterého je studentovi uděleno hodnocení příslušné semestrální práce. Vzhledem k tomu, že máme plány krátkodobé, střednědobé a dlouhodobé, lze odhadnout, zda plány plníme.

Jaká jsou kritéria úspěšného ukončení projektu?	Projekt nemá specifikovaný konec, měl by se neustále vyvíjet v rámci semestrálních prací.
Jaký je plán projektu? (zdroje, čas,náklady...)	Zdroje projektu jsou omezené počtem studentů, kteří se rozhodnou k projektu přidat. Čas věnovaný projektu je omezený časem studentů, kteří se projektu věnují v rámci semestrálních prací. V projektu se o finančních nákladech neuvažuje. (řeší se pouze časové zdroje)
Jaké jsou požadavky na zdroje projektu?	Vzhledem k tomu, že jde o studentský projekt, jsou zdroje vyjádřené počtem hodin v rámci semestrálních prací.
Jaké jsou požadavky na členy pracovního týmu?	Požadavky na členy pracovního týmu nejsou blíže specifikované, ale existuje dotazník pro ověření schopností nových členů týmu. Student v roli vedoucího projektu by měl prokázat zájem a schopnost samostatně zvládat situace, které v projektu nastávají (pokud není dobrý programátor, je nutné, aby si dobrého programátora do týmu opatřil). Pro programování projektu je nutné znát programovací jazyk C# a .NET. Předpokládá se znalost programování minimálně v rozsahu předmětu KIV/PT. Práci s podpůrnými nástroji a znalostní bázi by měl zvládnout student v 2. Ročníku.
Jaká je struktura pracovního týmu? (členové, role)	Nyní jsou všechny práce na projektu ukončeny a projekt připraven pro předání dalšímu vedoucímu. Struktura pracovního týmu se mění. Struktura týmu se vždy vytvoří na každý semestr zvlášť. Jednotliví studenti jsou v rámci

	semestrálních prací vedení přímo vedoucím projektu, pokročilejší týmy (ASWI) pak vedení zvládají samostatně a vedoucí projektu je zde pouze v roli zadavatele.
Jaké jsou požadavky na zadavatele projektu?	Projekt nemá zadavatele, má mentora. Na vizi projektu se podílí jednotliví studenti a vedoucí, kteří na projektu pracují.
Funguje projektová komunikace?	Projektová komunikace fungovala výborně. Pro rychlou komunikaci byla založena skupina na sociální síti Facebook.
Jak řešíte spory v týmu?	Ke sporům v týmu nedošlo. Problémy s týmy, které neplnily svou práci byly delegovány výš.
Jak a kdo plánuje jednotlivé úkoly a průběh celého projektu? Kdo kontroluje jednotlivé úkoly?	Jednotlivé úkoly nejsou plánovány, jsou plánována zadání v rámci semestrálních prací. Zadání semestrálních prací plánuje vedoucí projektu.
Odpovědnost v projektu (celý projekt, části projektu, týmy...)	Celý projekt má na starosti vedoucí projektu, který rozděluje jednotlivé práce.
Jaké nástroje používáte pro komunikaci v týmu. Jak?	Ke komunikaci je používána sociální síť Facebook a byly zavedeny pravidelné osobní schůzky.
Jak probíhá zálohování projektu?	Zálohujeme databázi wiki projektu, GitHub sice záloha není, ale prakticky se tak dá použít. V podstatě tedy zálohujeme veškerou svou práci. (kromě nastavení serveru)
Jak pracujete s dokumentací? (ukládání, práce s ní) Vytváříte znalostní bázi?	Existuje jedna přehledná znalostní báze projektu.
Jaká jsou omezení projektu? (finanční, časová, ..)	Finanční omezení projekt nemá. Je limitován rozsahem semestrálních prací, které zadáváme.
Jaká jsou kritéria úspěšnosti projektu?	Plnění plánů a cílů.
Jak identifikujete a hodnotíte rizika projektu?	Hlavní hrozby a rizika projektu byly odhaleny pomocí SWOT analýzy a byly navržena jejich

	řešení.
Jak zajišťujete kvalitu projektu?	Projekt není testován. Pro zvýšení kvality výstupů byly zavedeny kontrolní seznamy.
Jak probíhá měření projektu - zpětná vazba? (náklady, pracnost, doba trvání, produktivita, kvalita výstupů, spokojenost zadavatele s dodávkou)	Procesu vývoje bylo přidáno kontrolní patro, kdy je vedoucí projektu kontrolovaný shora v průběhu prezentací, které se musí připravit.
Zpětná vazba (metriky, náklady na měření)	Zpětná vazba probíhá pomocí kontrolních seznamů.

F. Zkrácená verze analýzy projektu:

Zkrácená verze nejdůležitějších otázek, které byly použité k analýze v navštívených firmách.

Tato zkrácená verze analýzy projektu byla použita v případě, že ve firmách nezbýval čas na dlouhou verzi analýzy. Také byla používána pro upřesnění některých získaných informací.

- Jak řešíte ztrátu, či změnu zdrojů (práce, lidi, čas)
- Metriky úspěšnosti (jak zjistíte, jak si projekt vede, ověřování kvality mezivýsledku)
- Klíčové podpůrné nástroje (vývoj, komunikace)
- Postupy používané pro řízení projektu
- Měření dosavadních výsledků a východisek.
- Jak řešit uvedení nového člena týmu do projektu (co nejrychleji)
- Změna týmu, změna vedení, jak předávat znalosti, změny plánování (jak se vyrovnat se změnou)
- Metriky pro kvalitu
- Jak zajišťovat, aby v rychle měnícím se prostředí netrpěl produkt, jak udržovat kvalitu projektu při těchto změnách

G. komunikace: (krátký dotazník členům týmu v průběhu práce)

Wiki stránka projektu, součást sekce pro vedoucí projektu (viditelná pouze pro vedoucí)

Tento dotazník byl sestaven pro zjištění efektivity komunikace v týmu. Členům týmu byl dáván v případě problémů. Také tyto dotazníky byly dány oběma diplomantům, kteří pracovali souběžně s touto diplomovou prací. Jejich obsah je součástí wiki, sekce pro vedoucí.

1. Jste spokojeni s rolími v týmu, nebo navrhuje změnu?
2. Co vás motivuje k práci na projektu?
3. Je vám při práci umožněno kreativní myšlení? Využíváte toho?
4. Jaké má podle vás projekt přinést výsledky?
5. Jaká jsou vaše očekávání od projektu?
6. Jak byste charakterizoval svou výkonnost v týmu?
7. Jakou formou probíhá diskuze? (problémy, nové nápady, porada s problémem)
8. Jak v týmu řešíte konflikty a krize?
9. Považujete se za spolehlivého člena týmu?
10. Považujete ostatní členy týmu za spolehlivé?
11. Jaký je váš osobní pocit z této práce?
12. Jak jste spokojen s plánováním projektu a úkolů?
13. Jste spokojen s odměnou, kterou za práci dostáváte?
14. Máte prostor k osobnímu rozvoji a zlepšení?
15. Cítíte se dostatečně oceněn?
16. Jak hodnotíte celkovou spolupráci s vedoucími
17. Jak vám práci usnadnily/ztížily checklisty?
18. K čemu podle Vás checklisty slouží? Byla to podle vás ztráta času?
19. Jak hodnotíte celkovou spolupráci na projektu.
20. Máte z práce na projektu dobrý pocit?
21. Když jste měli s něčím problémy, dovolali jste se pomoci?
22. Je něco, co nám chcete říct?
23. Bylo plánování a pravidelné schůzky pro vás přínosné?
24. Jaká byla užitečnost plánovaných schůzek a kontrol práce?

H. komunikace: (krátký dotazník členům týmu na konci práce)

Wiki stránka projektu, součást sekce pro vedoucí projektu (viditelná pouze pro vedoucí)

Tento dotazník byl dáván členům projektového týmu Space Traffic v době odevzdávání jejich semestrálních prací.

1. Byli jste spokojeni s rolmi v týmu, nebo byste něco změnili?
2. Co vás motivovalo k práci na projektu? Byla motivace dostatečná?
3. Bylo vám při práci umožněno kreativní myšlení? Využívali jste toho?
4. Jaké měl podle vás projekt přinést výsledky?
5. Jaká jsou vaše očekávání od projektu?
6. Jak byste charakterizoval svou výkonnost v týmu?
7. Jakou formou probíhala diskuze? (problémy, nové nápady, porada s problémem)
8. Jak jste v týmu řešili konflikty a krize?
9. Považujete se za spolehlivého člena týmu?
10. Považujete ostatní členy týmu za spolehlivé?
11. Jaký je váš osobní pocit z této práce?
12. Jak jste byl spokojen s plánováním projektu a úkolů?
13. Byl jste spokojen s odměnou, kterou jste za práci dostal?
14. Měl jste prostor k osobnímu rozvoji a zlepšení?
15. Cítíte se dostatečně oceněn?
16. Jak hodnotíte celkovou spolupráci s vedoucími
17. Jak vám práci usnadnily/ztížily checklisty?
18. K čemu podle vás checklisty slouží? Byla to podle vás ztráta času?
19. Jak hodnotíte celkovou spolupráci na projektu.
20. Máte z práce na projektu dobrý pocit?
21. Když jste měli s něčím problémy, dovolali jste se pomoci?
22. Je něco, co nám chcete říct?
23. Bylo plánování a pravidelné schůzky pro vás přínosné?
24. Jaká byla užitečnost plánovaných schůzek a kontrol práce?

I. Komunikace: (vyhodnocení spolupráce týmem Th3 developers)

Wiki stránka projektu, součást sekce pro vedoucí projektu (viditelná pouze pro vedoucí)

Tato wiki stránka obsahuje odpovědi na dotazník o komunikaci od týmu Th3 developers po ukončení semestrální práce Webové rozhraní hry Space Traffic v rámci semestrální práce z předmětu KIV/ZSWI v akademickém roce 2015/2016.

Jiří Šebor:

1. Ano, byl. Role jsme si rozdělili tak, aby to všem vyhovovalo.
2. Především získání zápočtu a můj osobní zájem o webové technologie. Ano, motivace byla dostatečná.
3. Ano, bylo a při některých úkolech jsem ho značně využil.
4. Vyzkoušet si práci v týmu, přidat novou funkcionalitu do hry Space Traffic.
5. Příklad se novými technologiím a práci v týmu.
6. Všechnu svou práci jsem odvedl jak nejlépe jsem uměl.
7. Schůzky s týmem a se zadavateli. Diskuze na Facebooku.
8. Diskuzí s členy týmu.
9. Ano.
10. Ano.
11. Dobrý, s prací jsem spokojen.
12. Velmi spokojen. Úkoly byly jasně dané a přidělené.
13. Ano.
14. Ano.
15. Ano.
16. Skvělá. Měli jsme jasné zadání a v případě potíží či nejasností jsme vše rychle vyřešili.
17. Ze začátku práci zpomalovaly, ale po chvíli jsem si na ně zvykl a pomohly mi práci si po sobě zkontrolovat.
18. K tomu, abych si po sobě zkontroloval odvedenou práci a zhodnotil její provedení. Doufám, že to ztráta času nebyla a pomůže to budoucím spolupracovníkům na projektu.
19. Velmi kladně. Spolupráce jak se zadavateli, tak s ostatními členy projektu probíhala bezproblémově.
20. Ano, mám. Práce na projektu mě bavila.
21. Ano, mohli jsme využít pravidelné schůzky se zadavateli nebo komunikaci pomocí Facebooku.
22. Dobře se mi s vámi spolupracovalo a těším se, až si hru zahraju.
23. Ano.
24. Velmi užitečné.

Marek Kotrč

1. Byl jsem spokojen. Role jsme si rozdělili podle našich schopností a potřeb projektu.
2. Především zápočet, následovaný užitečností projektu.
3. V mé práci jsem kreativitu používal minimálně, jelikož bylo potřeba dodržet některé zavedené mechanismy projektu.
4. V rámci samotného Space Traffic nejspíše zlepšení uživatelské přívětivosti a zprovoznění některé funkcionality. Projekt jako takový nám měl umožnit získání zkušeností s prací v rámci týmu.
5. Rozšíření mých znalostí v oboru a s prací v týmu.
6. Myslím, že svou práci jsem dělal nejlépe, jak jsem mohl.
7. Nejčastěji komunikací přes internet a větší problémy byly řešeny na společných schůzkách.
8. Středověký souboj s volbou zbraně dle libosti.
9. Ano.
10. Ano.
11. Práci jsme odvedli dostatečně kvalitně, proto jsou mé pocity převážně pozitivní.
12. Velmi spokojen.
13. Ano. Zápočet jsme obdrželi s vysokým počtem bodů a navíc jsme byli i pochváleni.
14. Ano.
15. Ano.
16. Velmi pozitivně. Instrukce byly jasně zadány a v případě problémů jsme kdykoliv mohli požádat o radu.

17. Z počátku působily nadbytečnou administrativní zátěž, ale postupně se z nich stala běžná rutina. Myslím, že největší přínos budou mít pro budoucí práci na projektu.
18. Nutily nás k zamýšlení nad svojí prací a svojí strukturou také podněcují k dělení práce na menší části.
19. Velmi pozitivně. Všechny případné problémy byly vyřešeny rychle a bez komplikací.
20. Ano.
21. Ano.
22. Chtěl bych poděkovat za možnost pracovat na tomto projektu a za váš pozitivní a přátelský přístup.
23. Ano.
24. Velmi užitečné.

Kolja Matuševský

1. Velmi spokojen. Myslím, že díky povaze projektu jsme si mohli rozdělit práci tak, aby vyhovovala každému z nás.
2. Kromě zápočtu i fakt, že mi tento projekt přišel jako dobrý nápad a jako užitečný zdroj práce pro další studenty.
3. Vzhledem k tomu, v jakém stavu byly některé části projektu, tak myslím, že se kreativně vyhnout nedalo. Já měl na starosti hlavně grafiku, takže kreativita byla nutná.
4. Dle mého názoru si hlavně vyzkoušet práci v týmu a ověřit si, zda je týmová práce něco pro mě, či nikoliv. Dále od samotného Space Traffic to bylo přivést projekt do hratelné podoby a přilákat nové hráče.
5. Vyzkoušení si práci v týmu, popřípadě se v týmu naučit pracovat a řešit problémy.
6. Dle mého názoru jsem dělal vše, jak nejlépe jsem mohl.
7. Většinou schůzkami ať už s vlastním týmem, nebo se zadavatelem, popřípadě společný chat na Facebooku.
8. Pokud to šlo, tak jsme se daný problém snažili řešit společně, popřípadě se zadavatelem. Hodně taky pomohl Google.
9. Ano.
10. Ano, za celou dobu práce v našem týmu se nám nestalo, že by někdo nesplnil to, co mu bylo zadáno.
11. Pozitivní, myslím, že jsme práci odvedli dobře.
12. Spokojen. Myslím, že úkoly jsme plánovali logicky a systematicky tak, aby nedošlo k žádným zmatkům.
13. Zápočet, pochvala a pocit, že tento projekt k něčemu byl. Co víc si přát.
14. Ano.
15. Ano.
16. Velmi dobrá. Dostali jsme jasné zadání, pokud jsme něčemu nerozuměli, tak jsme měli možnost se kdykoliv zeptat a každé dva týdny jsme měli schůzku, kde jsme probrali vše potřebné.
17. Ze začátku to bylo poněkud zmatečné a spíše to práci zpomalovalo, ale po chvíli si na to člověk zvykl. Nejsem si ale jistý, jestli to práci nějak zpřehlednilo. Ale možná, že pro budoucí studenty, kteří budou v práci pokračovat, to bude přínos.
18. K tomu, abychom si po sobě práci zkontrolovali, popřípadě pro přehlednost pro budoucí studenty, kteří budou na projektu pracovat. Pro programovací úkoly to byl celkem dobrý nápad, ale pro věci, které se netýkají programování (schůzky se zadavatelem, vytváření grafiky...) to nebylo nejlepší řešení a zpomalovalo to práci, protože jsme pořádně nevěděli, jak je vyplnit. Pro tyto neprogramovací věci by asi bylo vhodné vytvořit nový druh checklistů, nebo je vůbec nevyplňovat.
19. Myslím, že jako tým jsme pracovali velmi dobře a neměli jsme žádné personální problémy. I spolupráce se zadavatelem byla kvalitní.
20. Ano, myslím, že to pomůže ostatním studentům.
21. Ano, jakýkoliv problém jsme mohli řešit se zadavatelem.
22. Dobrá práce s vedením projektu, dobře se nám s vámi spolupracovalo.
23. Ano.
24. Velmi užitečné.