

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ
FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

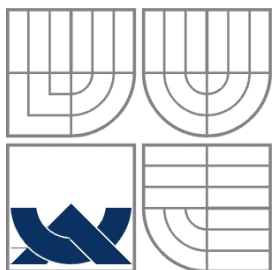
PROGRAMOVÁ PODPORA DEFINOVÁNÍ PROJEKTU
S VYUŽITÍM LFM (LOGICAL FRAME METHOD)

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

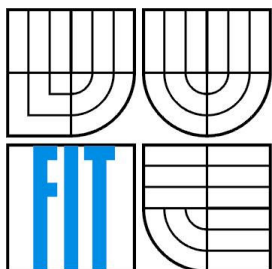
AUTOR PRÁCE
AUTHOR

BC. MARTIN VIKTORIN

BRNO 2008



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

PROGRAMOVÁ PODPORA DEFINOVÁNÍ PROJEKTU S VYUŽITÍM LFM (LOGICAL FRAME METHOD)

SOFTWARE SUPPORT FOR PROJECT DEFINITION WITH USING OF LOGICAL FRAME
METHOD

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

BC. MARTIN VIKTORIN

VEDOUCÍ PRÁCE

SUPERVISOR

ING. ŠÁRKA KVĚTOŇOVÁ

BRNO 2008

Abstrakt

Cílem diplomové práce je návrh a realizace programové podpory definování projektu s využitím LFM (Logical Frame Method). První část pojednává o projektovém řízení a nutnosti jeho použití při řešení projektů. Dále jsou popsány fáze projektu z hlediska projektového řízení, podrobně je popsán způsob definování projektu metodou logického rámce. V druhé části je pak uvedena analýza požadavků, návrh pomocí UML diagramů a zajímavé části implementace programu, který byl vytvořen ve vývojovém prostředí Microsoft Visual Studio 2005 v jazyku C#. V závěru je provedeno zhodnocení, srovnání s existujícími aplikacemi a jsou navržena možná rozšíření.

Klíčová slova

Řízení projektů, projekt, logický rámec, LFM, projektová fiše, metoda logického rámce, fáze projektu, definování, plánování, Microsoft Project, C#, Visual Studio, .NET Framework.

Abstract

The purpose of this work is to design and implement software support for Project Definition of LFM (Logical Frame Method). The first part is about project management and about necessity to use it for solving projects. There is description of phases of project and detailed description of logical frame method. In the second part there is system requirements analysis, description of the application via UML and implementation of application which was created in development environment Microsoft Visual Studio 2005 in C# language. There is a short summary, possible extensions and comparing with existing applications in the end of the work.

Keywords

Project management, project, logical frame, LFM, project fiche, Logical frame method, project phases, definition, planning, Microsoft Project, C#, Visual Studio, .NET Framework.

Citace

Viktorin Martin: Programová podpora definování projektu s využitím LFM (Logical Frame Method). Brno, 2008, diplomová práce, FIT VUT v Brně.

Programová podpora definování projektu s využitím LFM (Logical Frame Method)

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně pod vedením Ing. Šárky Květoňové. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Martin Viktorin
19. 5. 2008

Poděkování

Tímto bych chtěl poděkovat mé vedoucí Ing. Šárce Květoňové za pomoc a rady při řešení diplomové práce.

© Martin Viktorin, 2008.

Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.

Obsah

1	Úvod.....	4
2	Projekt.....	5
2.1	Základní pojmy	5
2.2	Kritéria projektu	5
2.3	Charakterické rysy.....	7
2.3.1	Cíl projektu	7
2.3.2	Jedinečnost.....	8
2.3.3	Zdroje.....	9
2.3.4	Organizace	9
2.4	Typické problémy	10
3	Řízení projektu.....	11
3.1	Význam řízení projektu	11
3.2	Fáze projektu z hlediska projektového řízení.....	11
3.2.1	Definování	12
3.2.2	Plánování	14
3.2.3	Organizování.....	20
3.2.4	Provádění/kontrola.....	21
3.2.5	Uzavření.....	22
3.3	Programová podpora	23
3.3.1	Microsoft Project	23
3.4	Řízení projektu v oblasti IT.....	24
3.5	Vyspělost projektového řízení v IT	25
4	Metoda logického rámce	27
4.1	Listina základního vymezení projektu	27
4.2	Význam metody	27
4.3	Struktura logického rámce.....	28
4.3.1	Hlavička logického rámce.....	29
4.3.2	Vertikální logika	29
4.3.3	Objektivně ověřitelné ukazatele.....	30
4.3.4	Zdroje (informací) k ověření.....	30
4.3.5	Rizika/předpoklady	30
4.4	Proces vytváření a čtení.....	30
4.4.1	Vytváření	30
4.4.2	Čtení.....	31

4.5	Projektová fiše.....	32
5	Program realizující podporu definování pomocí metody logického rámce	33
5.1	Analýza požadavků	33
5.1.1	Základní funkce	33
5.1.2	Uživatelské rozhraní	34
6	Použité technické prostředky	36
6.1	Microsoft Visual C#.....	36
6.1.1	Historie.....	36
6.1.2	Popis.....	36
6.2	Microsoft Visual Studio 2005	37
6.3	.NET Framework.....	37
7	Návrh programu	38
7.1	Diagram případů použití.....	38
7.1.1	Popis důležitých případů použití.....	38
7.2	Diagram tříd	39
7.2.1	Datový model.....	39
7.2.2	Model uživatelského rozhraní.....	41
8	Implementace	43
8.1	Uživatelské rozhraní.....	43
8.2	Zobrazení logického rámce	44
8.3	Uložení logického rámce.....	45
8.4	Vyplňování rámce	46
8.5	Průvodce vyplněním.....	47
8.6	Úprava vzhledu	48
8.7	Šablony	48
8.8	Export.....	50
8.8.1	MPX.....	50
8.8.2	XML.....	51
8.8.3	HTML	52
8.9	Nápověda.....	52
8.10	Ukázka zpracování projektu	53
9	Srovnání s existujícími aplikacemi	54
9.1	Logframe for Windows	54
9.1.1	Možnosti exportu logického rámce.....	54
9.1.2	Kvalita systému nápovědy	54
9.1.3	Ostatní vlastnosti.....	55
9.1.4	Rozšířenost aplikace Logframe.....	55

9.2	Internetová aplikace Dotačního informačního systému MMR	55
10	Možná rozšíření	56
	Závěr	57
	Literatura	58
	Přílohy	60
A	Zpracování projektu metodou logického rámce	60
B	Přiložené CD	66

1 Úvod

V současné době se jednorázové akce firem nebo organizací řeší v podobě projektů. Pokud chce být nějaká firma zaměřená na projekty úspěšná, musí fungovat systematicky, podle nějakých přesně stanovených pravidel a postupů. Důvodem je rychlá dynamika v obchodním sektoru a velká síla konkurence.

Nezbytností pro každou projektovou organizaci je řízení projektů. Jde o specifické postupy plánování a řízení a to s důrazem na čas, kvalitu a omezené zdroje. Jako příklady typických projektových organizací můžeme uvést stavební a technické firmy nebo státní investory. Pomocí řízení projektů mohou být projekty prováděny efektivně, což zvýší konkurenceschopnost podniku a naplní tak hlavní cíle každé organizace nebo firmy, tedy maximalizuje její zisky.

Nejdůležitějšími částmi řízení projektu jsou fáze definování a plánování. Při definování projektu je nutno určit přesně jeho cíle, vytvořit odhady časového plánu a nákladů a rozhodnout tak, jestli bude projekt opravdu schválen a proveden. Právě v této fázi je vhodné použít metodu logického rámce. Logický rámec je ale dále užitečný i při implementaci a hodnocení.

Hlavním cílem této diplomové práce je vytvoření aplikace, realizující podporu definování projektu s využitím LFM (Logical Frame Method).

Tato práce je rozdělena na dvě části. V první je uveden teoretický základ pro pochopení metody logického rámce. Vysvětlen je samotný pojem projekt, jsou uvedena kritéria a charakteristické rysy a také typické problémy, které mohou nastat při řešení. Dále se pojednává o řízení projektu, jeho významu, fázích projektu, softwarové podpoře s důrazem na Microsoft Project a potřebu projektového řízení v oblasti IT. V závěru první části jsou uvedeny standardy úrovně vyspělosti projektového řízení a je popsána metoda logického rámce, její význam a způsob jak vytvářet a číst logický rámec.

V druhé části je uvedena analýza požadavků a návrh samotné aplikace pomocí diagramů UML. Jsou zmíněny důležité části implementace. Na závěr je provedeno shrnutí a srovnání s existujícími aplikacemi a jsou diskutována možná rozšíření.

První část práce, konkrétně kapitoly 2-4, využívají výsledků získaných v Semestrálním projektu. V něm byl zpracován teoretický základ řízení projektů a metody logického rámce. Ten je pak využit při analýze požadavků a návrhu programu, tedy v kapitolách 5 a 7. Vhodné části jsou také použity v nápovědě aplikace.

2 Projekt

V dnešní době se jednorázové akce ve firmách a jiných podnicích provádí formou projektu. Tyto projekty jsou rozhodující součástí strategického řízení podniku. Může se jednat o jakoukoli akci, libovolného rozsahu, od těch, které mají za úkol uspokojit potřebu zákazníka, přes akce, které řeší vlastní potřeby podniku až po ty nejvíce specifické. Jako příklad můžeme uvést vytvoření výrobku, uvedení výrobku na trh, propagaci nového informačního systému, instalaci nových strojů v továrně nebo průzkum spokojenosti zaměstnanců ve firmě. Všechny tyto akce mají společné rysy. Někjaký cíl, kterého se snaží dosáhnout v omezeném čase a za použití omezených nákladů a zdrojů. Tyto rysy jsou charakteristické pro všechny projekty [1].

2.1 Základní pojmy

V této kapitole si uvedeme pouze některé nejzákladnější pojmy ohledně projektů a jejich řízení. Další budou podle potřeby definovány dále v textu.

Jak je tedy definován *projekt*? Je to jedinečný proces sestávající z řady koordinovaných a řízených činností s daty zahájení a ukončení, prováděný pro dosažení cíle, který vyhovuje specifickým požadavkům, včetně omezení daných časem, náklady a zdroji. (*ČSN ISO 10006, Management jakosti – Směrnice jakosti v managementu projektu, 1999*) [3].

Dalším důležitým pojmem je *řízení projektu*. Je to strategie při řízení projektu, která vyžaduje specifické postupy plánování a řízení, zejména s důrazem na čas, kvalitu a omezené zdroje.

Projektové řízení je uplatnění znalostí, dovedností, nástroj a technik v projektových činnostech s cílem splnit nebo překročit potřeby zájmových skupin a jejich očekávání od projektu. Projektové řízení je tedy uplatnění znalosti řízení projektu.

Při definování projektu jsme použili slovo *proces*. Proces je soubor vzájemně propojených zdrojů a činností, které přeměňují vstupy na výstupy.

Vedoucí projektu je osoba zodpovědná za řízení projektu. Projektový tým se skládá z lidí, kteří pracují na projektu a jsou přímo i nepřímo podřízeni vedoucímu projektu.

Posledním pojmem, který si na úvod uvedeme jsou *Zdroje*. Tímto pojmem rozumíme pracovníky, vybavení, materiál, dodávky, software, hardware atd., které jsou potřeba k realizaci projektových činností [5].

2.2 Kritéria projektu

Aby se dal proces nazvat projektem, musí splňovat určitá kritéria [3]. Ty jsou pak možným zdrojem problémů a rizik, které si můžeme stručně nastínit.

Aktivity se týkají více než dvou osob

Na projektu se podílí více lidí. Typický projekt totiž vyžaduje znalosti a dovednosti z mnoha oborů. Proto se na projektu podílejí odborníci a sjednocují svoje úsilí a dovednosti v zájmu splnění cíle projektu. To vyžaduje jejich organizaci. Musí se vytvořit tým, rozdělit role a přijmout veškerá opatření, aby v týmu nenastaly žádné problémy. Takovými problémy může být například komunikace v týmu, během projektu mohou vznikat mezi lidmi konflikty nebo může dojít ke změně počtu lidí, kteří se na projektu podílí.

Aktivity vyžadují více než dva týdny úsilí osob

Projekt trvá typicky delší časové období a je tudíž nezbytné dobře naplánovat všechny činnosti tak, aby byl včas hotov. Po dobu provádění projektu je nutno kontrolovat dodržování časového plánu. Lze předpokládat, že námi navržený časový plán nebude zcela přesně dodržen. V průběhu provádění projektu se totiž vyskytují problémy, které lze předem jen velmi těžko předvídat.

Budou vyžadovány nové postupy nebo nové technologie

Každý projekt je jedinečný. Ke splnění jeho cíle bude tedy nutno použít nové postupy nebo technologie. To sebou přináší riziko potíží, protože pracovníci mohou mít s těmito postupy nebo technologiemi jen malé nebo žádné zkušenosti. Může tak dojít například k nedodržení časového plánu, protože se ukáže, že pracovníci potřebují ke zvládnutí nové technologie více času, než se původně předpokládalo.

Rozpočet je těsný

Projekt má předem definovaný rozpočet, který je velmi těsný. Je to dáno snahou odhadnout náklady na projekt co nejpřesněji již na samém začátku. Odhadovaná suma pak bývá často příliš nízká, kvůli nutnosti přesvědčit zákazníka, aby si vybral právě naše řešení. Toto podbíživé řešení s sebou ovšem přináší riziko příliš malých nebo dokonce žádných rezerv v rozpočtu, což může mít za následek jeho překročení.

Některé úkoly jsou závislé na dokončení jiných úkolů

Jak již bylo řečeno, projekt je proces, sestávající z řady dílčích činností. Tyto činnosti na sobě mohou být závislé. Znamená to například, že k zahájení nějakého úkolu B potřebuji informace, které jsou výsledkem jiného úkolu A. Problém pak nastane, když se dokončení úkolu A zpozdí a úkol B tak nemůže začít včas.

Vývoj produktu se skládá z několika fází, které se musí koordinovat

Vývoj produktu se skládá z pěti základních fází: definování, plánování, organizování, provádění/kontrola a uzavření [3]. Kvalita provedení jedné fáze může ovlivňovat kvalitu fází dalších.

Každá fáze by měla produkovat výstup, který by měl být ověřitelný. Mělo by se podle něho tedy dát objektivně zjistit, jestli je výstup shodný s tím, co bylo požadováno.

Je požadováno měření kvality

Výstupy jednotlivých fází i cíl celého projektu by měl odpovídat požadavkům, které byly zadány na samotném začátku. Kvalita by se měla hodnotit pomocí předem definovaných a měřitelných vlastností. Pokud výsledná kvalita neodpovídá požadavkům, musí se obě zainteresované strany dohodnout na řešení. Dodavatel například přislíbí, že produkt vylepší za vlastní náklady, ovšem za cenu pozdního dodání.

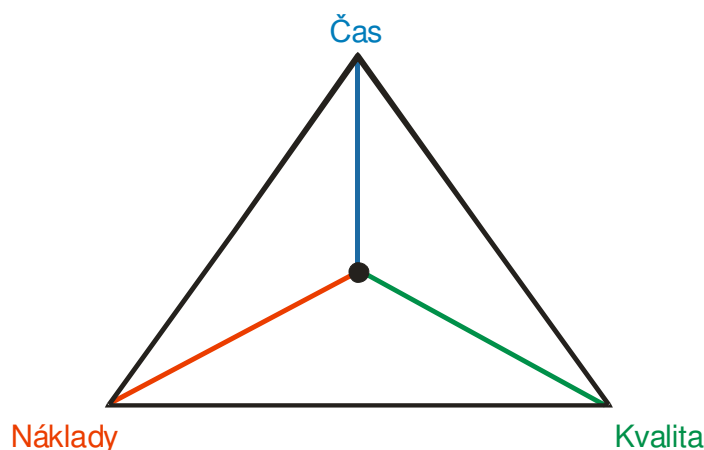
2.3 Charakterické rysy

Existují čtyři základní rysy, které odlišují řízení projektu od jiných manažerských činností. Těmito rysy jsou cíl projektu, jeho jedinečnost, potřebné zdroje pro jeho vykonání a organizace, v rámci které se projekt vypracovává [1].

2.3.1 Cíl projektu

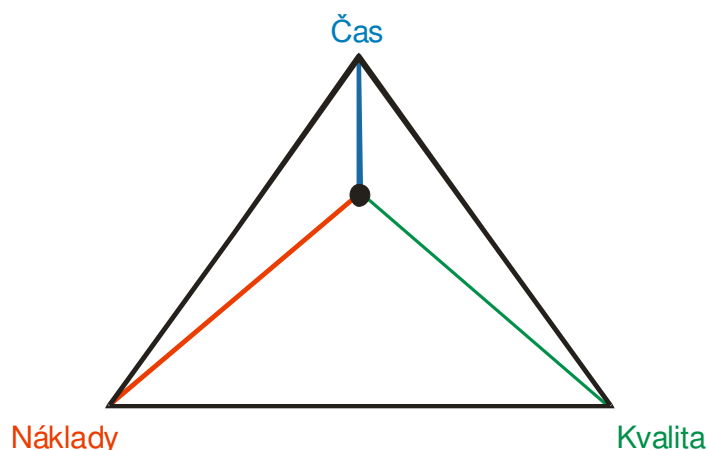
Cíl se skládá ze tří proměnných projektu, které musí být splněny současně a navíc se navzájem ovlivňují. Těmito proměnnými jsou čas, kvalita a náklady. Dohromady se označují jako trojimperativ. Trojimperativ definuje projekt a jeho splnění je hlavním cílem projektu.

Skutečnost, jak se proměnné trojimperativu navzájem ovlivňují ilustrují obrázek 2.1. a obrázek 2.2 [5].



Obrázek 2.1: Trojimperativ [5]

Parametry trojimperativu mohou být nastaveny různě. Záleží to vždy na konkrétních požadavcích zadavatele projektu. Zatímco někdo může požadovat produkt vysoké kvality za vysokou cenu a akceptuje dlouhou dobu trvání projektu, pro druhého může být rozhodující právě doba, za kterou bude produkt dodán s tím, že se spokojí i s nižší kvalitou.



Obrázek 2.2: Změna parametrů trojimperativu [5]

Trojimperativ je základem celého procesu plánování a sledování projektu a důsledky různých projektových výstupů jsou posuzovány právě z hlediska splnění jeho parametrů. Jejich splnění ovšem stěžuje mnoho problémů, které se během průběhu projektu mohou objevit. Více se o těchto problémech zmíníme v kapitole 2.4.

Parametry trojimperativu se mohou měnit i v průběhu řešení projektu. Pokud se zadavatel například rozhodne, že požaduje vyšší výslednou kvalitu, než bylo požadováno původně, bude patrně potřeba najmout na projekt víc pracovníků nebo zakoupit lepší vybavení, aby byl projekt splněn včas. To si vyžádá zvýšení nákladů.

2.3.2 Jedinečnost

Každý projekt je jedinečný. Provádí se jenom jednou, většinou ho řeší různé skupiny lidí a má omezené trvání.

I když se může zdát, že postavení dvou stejných domů je projekt stejný, je velice nepravděpodobné, že se budou stavět ve stejný čas, stejně dlouho a bude na něm pracovat stejná skupina pracovníků. Stejně tak domy nebudou zcela stejné a i drobné změny, které jsou vynuceny například jinou polohou domu, způsobí to, že budou během stavby provedeny různé činnosti.

Tyto odlišnosti mohou zapříčinit vznik neočekávaných a hlavně předtím neznámých problémů, se kterými se musí v každém projektu počítat.

Problémy způsobuje také omezená doba trvání. Po tuto dobu musíme zajistit potřebné zdroje a mít jistotu, že je budeme mít k dispozici až do ukončení projektu. Poté se musí vyřešit, zda se zdroje přesunou na jiný projekt nebo již nebudou potřeba. Tato situace by se dala přirovnat k dočasnému založení firmy, která by musela zajistit přijmutí lidí na jednorázovou zakázku a poté celou firmu nechat rozpustit.

2.3.3 Zdroje

Řešení projektů zahrnuje jak materiální tak i lidské zdroje. Důležité je potřebu zdrojů dobře odhadnout a naplánovat. Nedostatek může způsobit prodloužení doby trvání projektu, nadbytek se dá považovat za plýtvání penězi.

Hlavně lidské zdroje vyžadují zvýšenou pozornost. Manažer projektu by měl mít velké komunikační schopnosti a měl by být schopen a ochoten porozumět svým pracovníkům. Řídit projekt totiž znamená hlavně řídit lidi, kteří na něm pracují. Velkými nedostatky v tomto směru trpí technicky vzdělaní a zaměření manažeři. Jsou zvyklí řešit problémy, které mají jednoznačné a přesné řešení a nejsou tak schopni řešit konflikty v týmu, které vyžadují cit pro lidské chování.

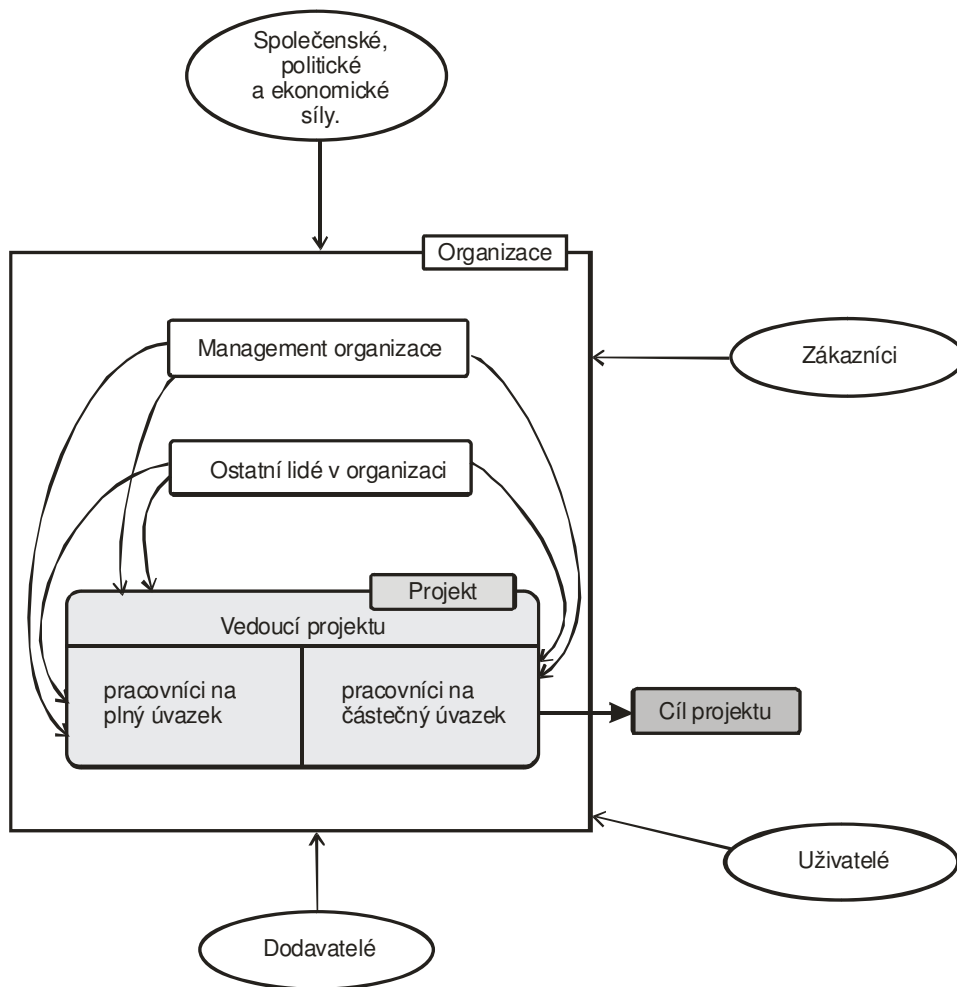
Dalším problémem je přetěžování zdrojů. Pokud jde o ty materiální, jde o problém závažný. Ovšem v případě lidských zdrojů je jejich přetěžování ještě větším problémem. Pracovník, který bude nucen dělat dvojnásobek práce a pracovat přesčas jistě nebude mít takovou chuť plnit všechny úkoly svědomitě a také budoucí spolupráci na dalších projektech bude pečlivě zvažovat.

2.3.4 Organizace

Projekty se často řeší ve velkých organizacích, které sledují značné množství cílů. Je to dáno rozmanitým složením organizace a také vlivem zákazníků a uživatelů. Dalším problémem je, že se v organizaci často řeší mnoho projektů najednou.

Vedoucí projektu a jeho tým se pak může často cítit zmatený a bezmocný, protože cíle se v průběhu řešení projektu mění a navíc si často i protirečí a navzájem se vylučují.

Vlivy na organizaci a projektový tým jsou znázorněny na obrázku 2.3 [1]. Na celou organizaci působí zákazníci, uživatelé, dodavatelé a společenské, politické a ekonomické síly. Tyto vlivy, spolu s dalšími cíly, pak pomocí managementu i ostatních lidí v organizaci působí i na projektový tým.



Obrázek 2.3: Různé vlivy na organizaci a projektový tým [1]

2.4 Typické problémy

Hlavním problémem, který se vyskytuje u velkého počtu projektů je překročení odhadovaných nákladů nebo času trvání. Důvodem může být špatné porozumění cílům projektu již při jejich definování na samém začátku nebo problémy, které se vyskytují během řešení. Další problémy může vytvářet organizace projektového týmu. Často je vedením pověřena osoba, která má sice výborné technické znalosti, ale neumí dostatečně komunikovat s lidmi a řešit různé konflikty.

Předcházet všem předvídatelným problémům a umět řešit ty nepředvídatelné tak, aby byl splněn trojimperativ, je úkolem řízení projektu.

3 Řízení projektu

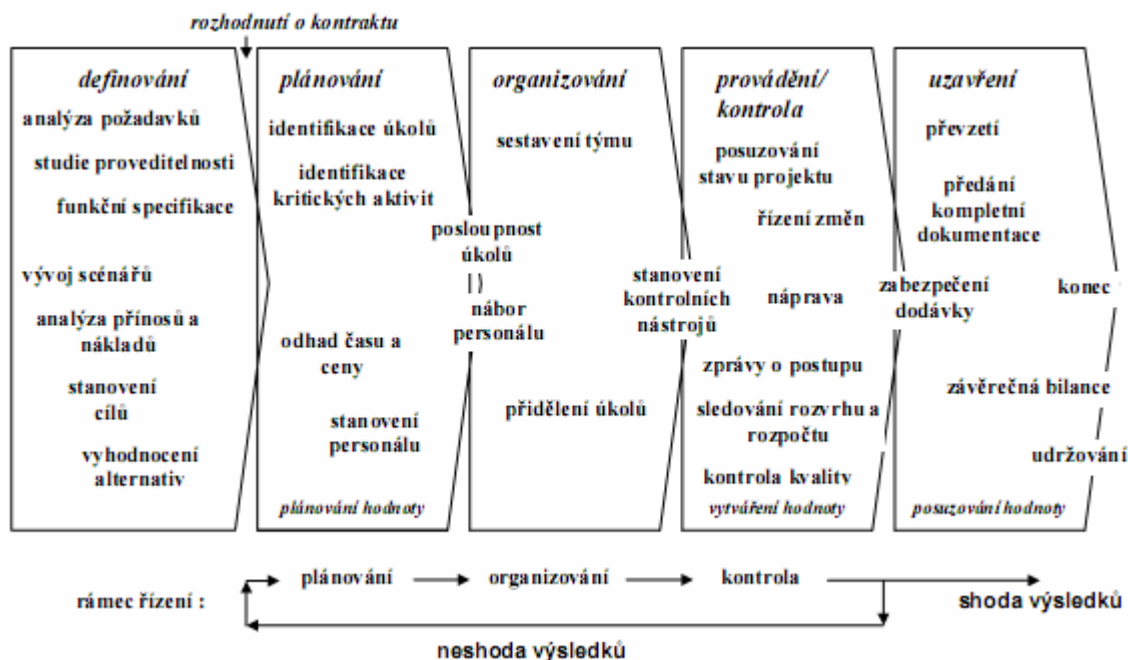
3.1 Význam řízení projektu

Pokud chce být dnes nějaká firma zaměřená na projekty úspěšná, je nezbytné, aby fungovala systematicky, podle určitých přesně stanovených pravidel a postupů. Je to způsobeno zvětšující se dynamikou v obchodním sektoru. Proto by se mělo jednat velmi rychle, jinak může zaváhání využít konkurence a připravit firmu o významné zisky. Na druhou stranu je dnes také požadován vysoký stupeň kvality produktů. A navíc, vzhledem k velké konkurenci, očekávají zákazníci co možná nejnižší cenu služeb i produktů. Tyto všechny požadavky není jednoduché splnit. Ulehčení jejich splnění umožňuje právě řízení projektu.

Hlavním významem řízení projektu je maximalizovat zisky společnosti. To znamená pokud možno co nejvíc snížit náklady a zvýšit tržby.

3.2 Fáze projektu z hlediska projektového řízení

Projekt se z hlediska projektového řízení dělí na pět základních fází. Jsou jimi definování, plánování, organizování, provádění/kontrola a uzavření. Souvislosti fází ilustruje obrázek 3.1 [3].



Obrázek 3.1: Vztah fází projektu z hlediska projektového řízení [3]

Každá etapa se vyznačuje dokončením jednoho nebo více předmětů dodávky. Předmět dodávky by měl být hmotný a ověřitelný. Při uzavření každé fáze je nutné přezkoumat a zhodnotit klíčové

předměty dodávek a plnění úkolu a určit tak, zda je možné přejít do další fáze, popřípadě zjistit a opravit chyby [3].

3.2.1 Definování

Hlavním cílem fáze definování je vytvoření cíle projektu, tedy projektového trojimperativu. Tím se definují tři hlavní proměnné projektu - čas, náklady a kvalita. Trojimperativ, nástroj k definování a hodnocení projektu, je zmíněn již v kapitole 2.3.1.

Ještě před zahájením prací na projektu je nutno zanalyzovat, zda je projekt pro firmu, která uvažuje o jeho realizaci, vhodný. Tato analýza by měla brát v potaz dlouhodobé cíle firmy a současné nebo v krátké době dostupné zdroje. Firma nebo organizace by se tedy neměla ucházet o projekt, který nezapadá do koncepce dlouhodobých cílů, i když může být finančně zajímavý. Takový projekt by mohl bránit rozvoji firmy předem určeným směrem a to by mohlo mít za následek problémy v budoucnosti. Nedostupnost zdrojů by byla velkou překážkou pro úspěšné dokončení projektu. Zdroje by se sháněly narychlo nebo by se museli využít externí dodavatelé, což by mohlo mít za následek zpoždění nebo prodražení projektu.

Mnoho organizací se také zbytečně zabývá přípravou ztrátových projektů, které nemohou obstát v soutěži nebo jejich výstup bude nepříznivý z hlediska trojimperativu. Důležité tedy je, soustředit se na smysluplné perspektivní projekty, které má organizace šanci získat [1].

Pokud se firma rozhodne o projekt ucházet, musí předložit svůj návrh, ve kterém definuje a popisuje svoje řešení projektu. Hlavní části návrhu tvoří definice cílů a rozsahu a předběžný plán.

3.2.1.1 Logický rámec

Na začátku celého plánovacího procesu by měl být sestaven logický rámec projektu. Ten tvoří základ pro přípravu jednotlivých aktivit a rozvoj monitorovacího systému. Není důležitý jen ve fázi přípravy projektu, ale je klíčovým nástrojem také při implementaci a hodnocení [8].

Metoda logického rámce je podrobně popsána v kapitole 4.

3.2.1.2 Definice cílů a rozsahu

Definice cílů a rozsahu musí popisovat, co má být uděláno. Také by měla obsahovat všechny specifikace, které budou použity. Měla by určovat měřitelná, hmotná a ověřitelná akceptační kritéria, která budou použita pro posouzení přijatelnosti konečného výstupu projektu [1].

Velmi častým problémem jsou nejasné a nepřesné cíle zadavatele. Pokud se ve fázi definice nevěnuje dostatečná pozornost úplnému porozumění zadání a upřesnění všech nejasností, vede to k nesprávnému pochopení celého projektu, sestavení chybného plánu a ztrátám organizace. Možností, jak definovat přesně cíle a rozsah projektu, je více. Mezi tradiční patří rozhovor se zákazníkem. Tato metoda patří mezi primární. Obvykle mívá dvě části – strukturovanou a nestrukturovanou. Ve

strukturované části jsou zákazníkovi podávány předem připravené otázky. Druhá část probíhá spíše formou neformálního setkání, ve které jde o získání podrobnějších informací.

Při rozhovoru je důležité pokládat otázky tak, aby jim zákazník rozuměl. Problémem může být používání odborných nebo slangových výrazů. V tomto případě je nutností výrazy předem vysvětlit, popřípadě omezit jejich používání [5].

Další možností je získání požadavků formou dotazníku. V tomto případě jsou předem připraveny testové otázky s možnostmi. Dotazník může zákazník vyplňovat sám a může být odeslán například poštou. Pokud ovšem nějaké otázce nerozumí dostatečně, nemá možnost se zeptat. Řešením je přítomnost pracovníka firmy, který může vysvětlit nejasné otázky.

Mezi moderní metody patří například brainstorming a speciálně u softwarových projektů prototypování nebo společný vývoj aplikací.

Brainstorming je konferenční technika pro hledání nových myšlenek a hledání možných řešení problému. Používá se v případech, kdy cíle definuje více osob najednou a těžko se mezi sebou dohodnou. Celou akci vede moderátor, který položí „spouštěcí“ otázku. Každý z účastníků pak napíše na papír anonymně svůj názor. Papíry pak kolují mezi účastníky. Poté se názory čtou, diskutuje se a hlasuje se o nich. Nápad s nejvíce hlasy je pak přijat jako řešení.

Prototypování je nejpoužívanější moderní metodou získávání požadavků. Po získání základních požadavků se vyrobí prototyp softwaru, který simuluje skutečný softwarový produkt. Zákazník má možnost si prototyp odzkoušet a přijít tak na nové požadavky, které by jinak nebylo možno zjistit.

Společný vývoj (JAD) je technika podobná brainstormingu a je také založena na skupinové spolupráci. Diskuze se účastní moderátor, který diskusi řídí, písař, který vše zaznamenává, zákazníci, kteří diskutují o požadavcích a přijímají rozhodnutí a vývojáři, kteří spíše poslouchají a ujasňují si informace [5].

3.2.1.3 Předběžný plán

Ve fázi definování je nutností vytvořit předběžný plán projektu. Ten slouží jako simulace celého projektu a pomůže tak odhadnout cenu a čas projektu. Čím podrobnější plán, tím by měly být odhady přesnější. Příliš podrobný plán ovšem zabere více času a stojí větší množství peněz. Musí se tedy najít vhodný kompromis.

Již ve fázi definování by měl být kladen důraz na reálnost vytvořeného plánu. Jestliže bude plán nesplnitelný již na papíře, nebude ho možné dodržet ani v praxi.

Plán má tři dimenze, protože sleduje splnění trojimperativu. Pro splnění parametrů kvality je nutné si nejprve sestavit hierarchickou strukturu činností (WBS). Ta popisuje dílčí úkoly, které se budou řešit. Z takto vyznačených prvků WBS je pak možné sestavit síťový graf, který určí vzájemné logické vztahy mezi činnostmi a umožní odhadnout časovou dimenzi. Po odhadnutí času lze přiřazením nákladů na jednotlivé zdroje odhadnout náklady celého projektu [1].

Pokud je návrh projektu přijat a na projektu se začne pracovat, musí být plán doplněn a upraven na vyhovující stupeň podrobností.

3.2.2 Plánování

Plánovací činnosti jsou pro řízení projektu rozhodující. Plánování je simulací průběhu projektu a pomáhá rozpoznat možné problémy a předcházet jim. Jak již bylo zmíněno v kapitole 3.2.1.3, plán sleduje tři proměnné projektu: kvalitu, čas a náklady. Proto jsou projektové plány ve skutečnosti tři: jeden pro dimenzi provedení (hierarchická struktura činností – WBS), jeden pro dimenzi času (sít'ový diagram, seznam milníků, úsečkový diagram) a jeden pro dimenzi nákladů (finanční rozpočet) [1].

Plán je možné začít tvořit až, když před sebou máme určitý cíl. Popisuje cestu, jak cíle dosáhnout s ohledem na všechna možná omezení, jakými jsou například zdroje nebo termíny.

Efektivní projektový plán má následující vlastnosti:

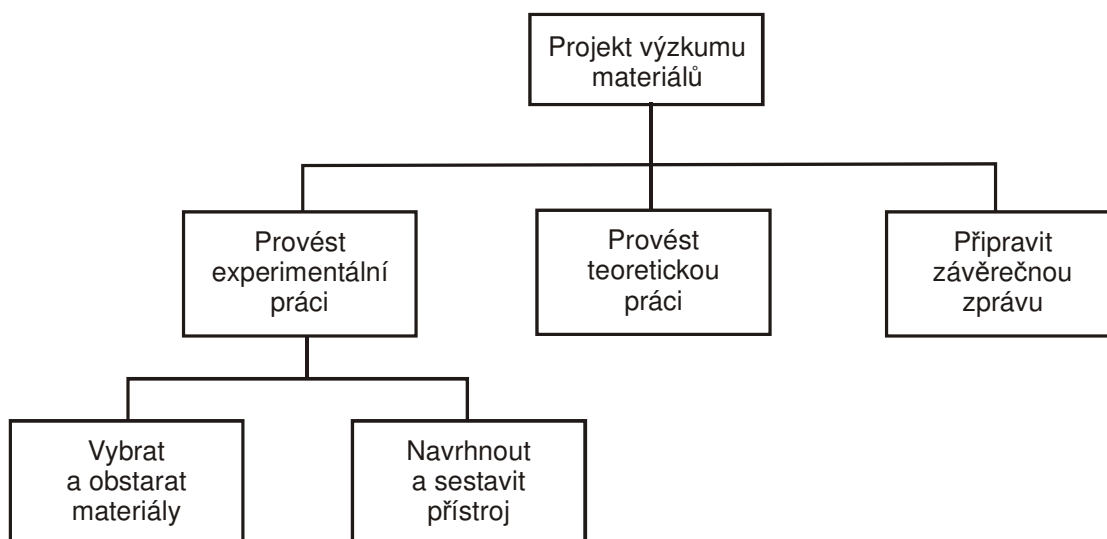
- Identifikuje vše, co je zapotřebí k úspěšnému dokončení projektu.
- Obsahuje harmonogram pro načasování těchto úkolů a souvisejících milníků.
- Definuje potřebné zdroje se zárukou jejich dostupnosti v určitou dobu a zohledňuje nasazení a jejich řízení
- Má rozpočet nákladů pro každý úkol
- Obsahuje rezervu rozpočtu pro nepředvídatelné události a problémy
- Je reálný a věrohodný jak pro předpokládané realizátory, tak pro management [1].

Důležité také je, aby plán nesestavoval pouze vedoucí projektu, ale podíleli se na něm pracovníci, kteří ho budou realizovat. Takto se dají do plánu zanést informace, které by samotného manažera nenapadly. Navíc je takový plán reálnější a pracovník bude k jeho provedení více motivován.

Nutné je také počítat s tím, že plán se bude v průběhu realizace projektu měnit. Vlivem nepředvídatelných událostí a vzniklých problémů bude docházet ke zpoždění některých úkolů a manažer na tyto okolnosti bude muset reagovat. Nejčastěji bude muset pozměnit projektový plán tak, aby byla zachována podmínka splnění trojimperativu. Plán by měl být také stále aktuální a měl by odrážet všechny změny, které v průběhu realizace projektu nastanou.

3.2.2.1 Hierarchická struktura činností (WBS)

Hierarchická struktura činností je seskupení prvků projektu podle předmětů dodávek a definuje celkový rozsah prací projektu. Činnost, která není zahrnuta do WBS nepatří do rozsahu prací projektu. Často se používá pro stanovení nebo potvrzení společného chápání rozsahu prací projektu. Každé položce prací je ve WBS obvykle přidělen jednoznačný identifikátor (účetový rozvrh). Položky na nejnižší úrovni WBS jsou často označovány jako balíky prací [3].



Obrázek 3.2: Ukázka Hierarchické struktury činností (WBS) [1]

WBS tedy pomáhá rozložit velké úkoly na menší díly a to hierarchicky až na nejnižší úroveň, která definuje již dobře zvládnutelné činnosti. Tímto způsobem lze identifikovat a logicky propojit všechny požadované projektové činnosti, což snižuje pravděpodobnost, že se na nějakou zapomene.

Je vhodné uvážit velikost WBS. Detailně propracovaná struktura umožní lepší naplánování projektu, stejně jako zpřesní odhad času a ceny. Musíme ovšem brát v potaz náklady na její vytvoření. Mnohdy mohou převyšovat náklady na řešení problémů, které by se vyskytly z důvodu méně propracované struktury. Stejně tak není vhodné brát celý projekt jako jeden úkol. Ten bude příliš velký a nákladný. Řešením je zlatá střední cesta. V tomto případě záleží na zkušenostech manažera projektu. Pomoci může také použití WBS z nějakého předešlého podobného projektu [1].

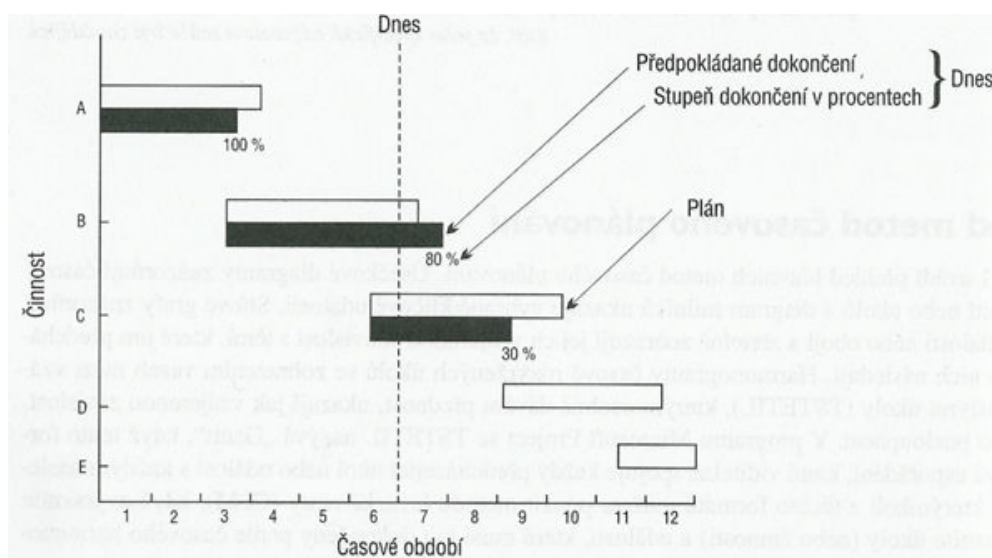
3.2.2.2 Nástroje časového plánování

Existují čtyři základní nástroje, které se používají k plánování času. Jsou to úsečkové diagramy, seznamy úkolů nebo milníků, síťové diagramy a nejpoužívanější TSTETIL diagramy, což jsou časově rozvržené úkoly s viditelnými vazbami vzájemně závislých úkolů [1]. V následujícím textu si je stručně popíšeme a uvedeme jejich hlavní výhody a nevýhody.

Jako vstup při tvorbě diagramů by měla sloužit hierarchická struktura činností (WBS) a do diagramu by měl být zanesen každý její prvek.

Úsečkové diagramy

Úsečkové diagramy jsou také často nazývány Ganttovy diagramy, podle H. L. Gantta, který je zavedl během první světové války. Činnosti se zakreslují do diagramu jako vodorovné úsečky [1]. Vodorovná osa zobrazuje čas a úsečka tak vypovídá o začátku a konci činnosti. Na úsečce lze též vyznačit aktuální stav průběhu činnosti, například číslem, které udává, kolik procent činnosti je již hotovo.



Obrázek 3.3: Úsečkový diagram [1]

Úsečkové diagramy lze snadno sestavit, pochopit a měnit. Graficky znázorňují, které činnosti jsou oproti plánu v předstihu a které jsou opožděny. Nevýhodou jsou však chybějící vztahy mezi činnosti a diagram tím pádem nic nevypovídá o aktuálním stavu celého projektu, pouze dílčích činností. Nelze ho tedy použít pro řízení projektu [1].

Seznam úkolů nebo milníků

Milník je kontrolním bodem projektu, který v něm představuje významnou událost. Označuje konec nějaké aktivity procesu a používá se k monitorování průběhu projektu. Milník je používán zpravidla k reprezentaci kompletnosti projektové fáze (provedení je typickým výstupem uzavírací fáze) [5].

Časový plán milníků zaznamenává klíčové události, tedy milníky, na kalendářním úsečkovém diagramu. Důležité je v tomto případě vhodně zvolit počet milníků tak, aby jich nebylo příliš a jejich kontrola nebyla spíše na škodu. Proto musíme vybrat opravdu důležité události.

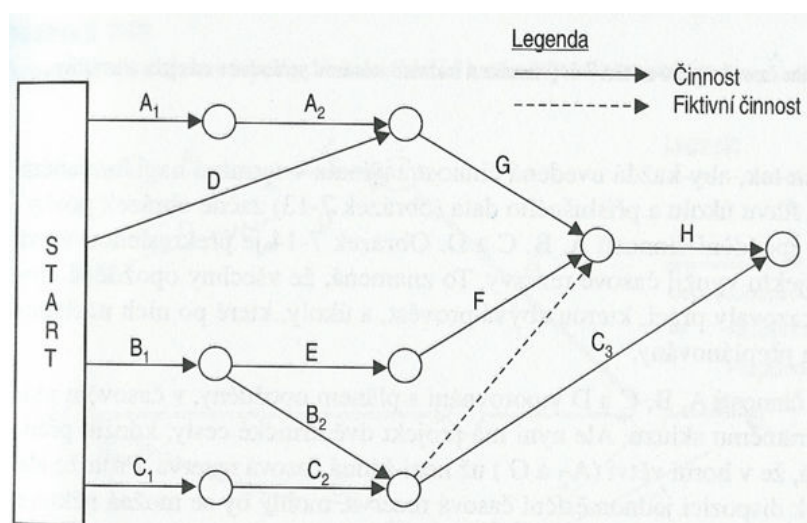
Plány milníků ovšem neposkytují, stejně jako úsečkové diagramy, jasnou představu o vzájemných vazbách mezi úkoly a činnostmi a proto je nutné je použít spolu s jinými nástroji [1].

Sít'ové diagramy

Sít'ový diagram je jakékoli z několika grafických zobrazení, které spojuje činnosti a události s cílem zobrazit jejich vzájemné závislosti. Každá činnost nebo úkol má vzájemné vazby s předcházející, následující či souběžně probíhající činností nebo úkolem [1].

Sít'ové diagramy ukazují nejdůležitější vzájemné vazby mezi činnostmi. Jsou doporučenými postupy plánování časové dimenze pro jakýkoli projekt.

Sít'ových diagramů existuje celá řada. Mezi nejoblíbenější patří sít'ový graf logického sledu činností PERT, uzlově orientovaný sít'ový graf PDM a hranově orientovaný sít'ový graf ADM. Liší se hlavně významem hran a uzlů grafu. Některé mají činnosti přiřazenou hraně, některé uzlu a další mají v uzlu událost [1].

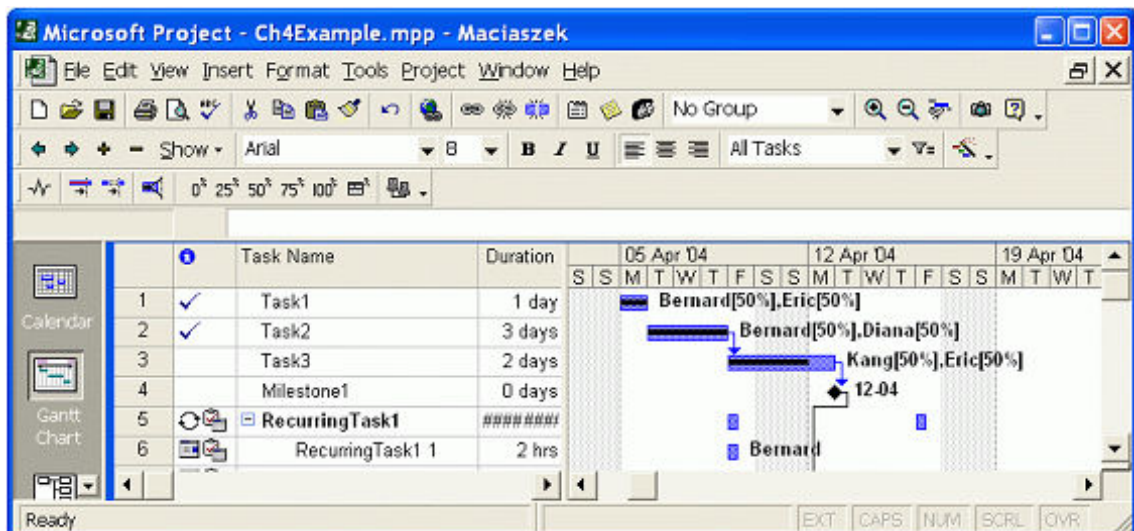


Obrázek 3.4: Sít'ový diagram [1]

Největší výhodou sít'ových diagramů je to, že nám dávají velmi dobrý přehled o stavu celého projektu. Vlivem provázanosti činností přehledně vidíme jejich závislosti a hlavně jak zpoždění jedné činnosti ovlivňuje zpoždění celého projektu.

TSTETIL diagramy

TSTETIL diagram je v podstatě sít'ový diagram s činností na hraně, ke kterému přidáme časovou osu. Tím udává vodorovná vzdálenost, kterou rozpětí hrany pokrývá, časový plán a přitom jsou pomocí spojů jasně znázorněni předchůdci a následovníci. Tento graf podporuje také software pro řízení projektů Microsoft Project a to pod názvem Ganttův diagram. Tento diagram je nejpoužívanější a poskytuje přehledně o projektu nejvíce informací.



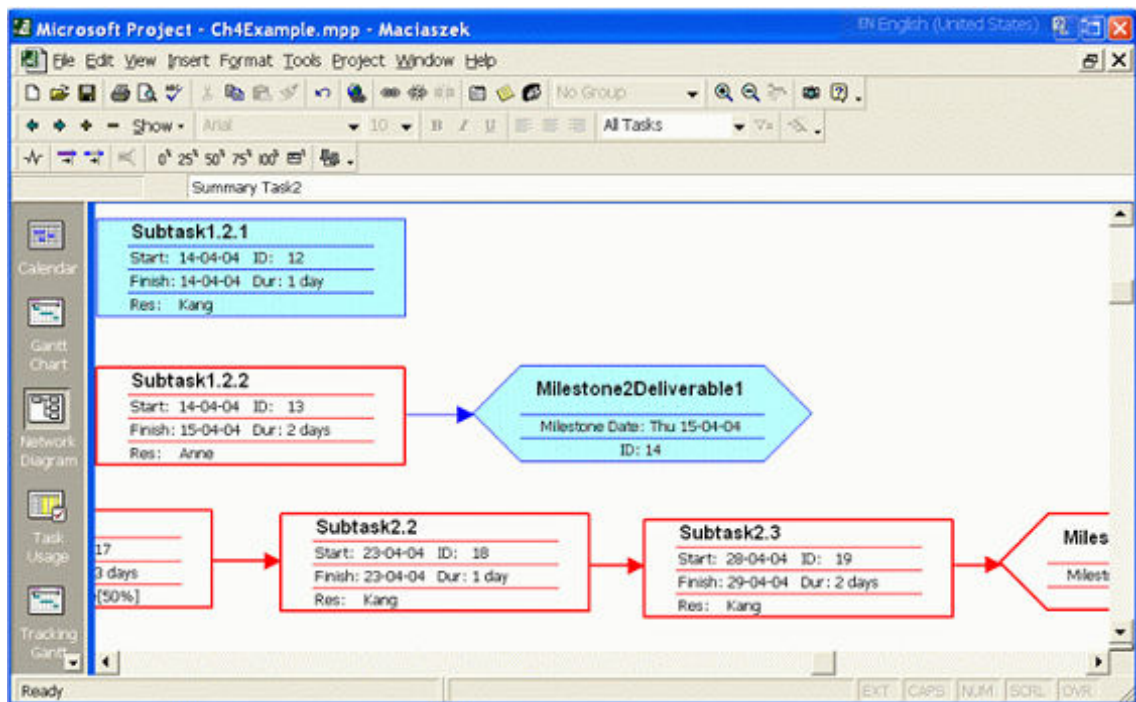
Obrázek 3.5: Ganttův diagram v programu Microsoft Project [5]

Kritická cesta

V každém z popisovaných diagramů lze znázornit kritickou cestu projektu. Kritická cesta je sekvence vzájemně propojených úkolů v projektu, která musí být splněna včas, aby mohl projekt proběhnout v rámci daného časového plánu – do určitého termínu. Zpoždění průběhu nějakého úkolu, který leží na kritické cestě, způsobí prodloužení celkové doby trvání projektu [5].

Kritická cesta tedy udává nejkratší možnou dobu, za kterou lze projekt dokončit a nemá žádnou časovou rezervu [1].

Nejvhodnějším diagramem pro znázornění kritické cesty je síťový diagram. Kritickou cestu lze také dobře znázornit v diagramu TSTETIL. Oba diagramy i nalezení kritické cesty podporuje také většina softwaru pro řízení projektů, včetně Microsoft Project.



Obrázek 3.6: Kritická cesta v síťovém diagramu programu Microsoft Project [5]

3.2.2.3 Plánování rozpočtu

Plán nákladů pomáhá zabránit situaci, kdy jsou skutečné náklady na projekt vyšší než odhadované nebo kdy není projekt schválen z důvodu nadhodnoceného rozpočtu.

Nutnými vstupy pro vytvoření plánů nákladů je hierarchická struktura činností (WBS) a nějaký síťový diagram. Bez nich nemá smysl náklady odhadovat. Plán nákladů bude nejpřesnější, pokud se provede odhad zvláště pro každý pracovní balík WBS. Tyto úkoly jsou malé a náklady se tak odhadnou přesněji, než u úkolů větších.

Plán rozpočtu pak lze vytvořit v běžném tabulkovém procesoru. Do plánu se zaznamená odhadované množství pracovních hodin, nepracovní výdaje pro každou pracovní skupinu nebo jednotlivce a hodinová sazba pracovníků. Pomocí těchto údajů lze rozpočet odhadnout.

Velké usnadnění práce v tomto případě znamená použití specializovaného softwaru pro řízení projektu. Tomu lze při vytváření struktury činností ke každé zadat informace o potřebných zdrojích, hodinové mzdě a nepracovních nákladech a software na základě těchto informací plán nákladů vytvoří sám.

Existují také zkrácené postupy odhadování nákladů. Jedním z nich je metoda „zdola nahoru“. Při ní se rozdělí projekt na malé balíky, u kterých je možné provést přesný odhad. Výsledné náklady jsou pak součtem těchto odhadů. Tuto metodu je dobré zkontrolovat rychlým odhadem „shora dolů“. Pokud se tyto odhady výrazně liší, je dobré odhad „zdola nahoru“ překontrolovat [1].

Z dalších metod můžete zmínit odhadování nákladů na základě podobnosti s nějakým již proběhnutým projektem nebo parametrické modelování, které využívá charakteristik (parametrů) projektu v matematickém modelu pro odhadování nákladů projektu [3].

3.2.3 Organizování

K řízení a organizování projektu může být použita projektová, funkcionální nebo maticová struktura. Funkcionální topologie organizace je z pohledu řízení projektu tou nejméně vhodnou. Pro dlouhodobé a rozsáhlé projekty je nejvhodnější projektová topologie a maticová topologie je vhodná zejména pro organizace s velkým množstvím menších projektů [1].

3.2.3.1 Projektový tým

Projektový tým se skládá z lidí, kteří jsou organizačně podřízeni vedoucímu projektu. Těmto lidem může vedoucí přiřadit pracovní balíky bez nutnosti jednání s jinými manažery. Lidé v projektovém týmu mohou pracovat výhradně na projektu nebo mohou mít také jiné úkoly. Stejně tak mohou být v týmu po celou dobu provádění projektu nebo jen po nějakou část.

Kromě projektového týmu se účastní prací na projektu také tým podpůrný. Členové tohoto týmu jsou podřízeni jinému manažerovi, avšak na projektu se mohou podílet stejnou měrou jako pracovníci projektového týmu.

Vhodnými zdroji pracovníků projektového týmu jsou členové návrhového týmu, kteří připravovali návrh projektu. Ti jsou do dané problematiky již zainteresováni a znají dobře návrh i požadavky zákazníka.

Dalším zdrojem pracovníků jsou zaměstnanci organizace. Ti sice nemusí mít znalosti ohledně projektu, ale ví, jak organizace funguje, jaké jsou hlavní cíle a manažer navíc často zná jejich charakter a dovednosti. Navíc je pravděpodobné, že spolu na nějakém projektu již pracovali a tak se znají i navzájem.

Často je nutné přijmout i lidi mimo organizaci. Může se jednat o poradce, externí pracovníky nebo subdodavatele. Při jejich najímání je nutné provést různé pohovory, získat informace z jejich portfolia nebo prostudovat reference od minulých zaměstnanců aby manažer projektu poznal alespoň základní vlastnosti a schopnosti [5]. Poté je nutné pracovníka zaškolit a seznámit ho se zvyklostmi organizace.

Manažer projektu musí počítat s faktem, že ne vždy bude mít k dispozici pracovníky podle své představy a mnohdy bude muset obsadit nějaké místo člověkem, jehož schopnosti a vlastnosti mu ne zcela vyhovují.

Většina této kapitoly čerpá z [1].

3.2.3.2 Komunikace v týmu

Důležitou součástí organizace projektového týmu je komunikace. Pro většinu pozic v týmu je schopnost komunikace dokonce důležitější, než například technické dovednosti.

Řízení komunikace v rámci projektu zahrnuje procesy požadované pro včasné zajištění odpovídající tvorby, sběru, šíření a uchování informací o projektu a konečné nakládání s těmito informacemi.

Komunikaci nalezneme v celé řadě forem. Mezi nejčastější patří neformální komunikace mezi lidmi, telefonická konverzace, komunikace přes internet nebo pomocí pošty, ať už klasické nebo internetové.

Komunikace v týmu musí být řízena a kontrolována. Musíme být schopní docílit toho, aby se informace dostala ke všem zainteresovaným stranám, kterých se týká [5].

3.2.3.3 Role vedoucího projektu

Vedoucí projektu zodpovídá za splnění trojimperativu a tím i za úspěch celého projektu.

Vedoucí musí mít specifické schopnosti a vlastnosti. Musí rozumět lidem, mít schopnost naslouchat jejich přáním a potřebám, umět motivovat a řídit kolektiv. Protože projekty nejsou rutinní záležitostmi, musí často přistupovat k řešení úkolů a problémů tvůrčím způsobem. Vedoucí projektu by měl být součástí týmu a ne jenom autorita, která tým řídí [1].

3.2.4 Provádění/kontrola

V průběhu realizace projektu je nutno provádět kontroly. Kontrolami se zjišťuje, zda projekt probíhá podle plánu. Plán se musí kontrolovat pro všechny tři složky trojimperativu. Jak by měla realizace projektu probíhat ukazuje hierarchická struktura činností (WBS), síťový graf, obsahující každý prvek WBS a plán nákladů pro každou činnost. Jakákoli odchylka od plánu signalizuje možný problém a vyžaduje opravnou akci.

Kontroly lze provádět tak, že se zavedou omezující kontrolní nástroje. Pracovníci musí například manažera žádat o poskytnutí zdrojů, jejichž náklady přesahují 30 000 korun. Manažer má tak všechny náklady pod kontrolou, navíc je schopen podle plánu rozpoznat například zpoždění, pokud zjistí, že pracovníci některé skupiny měli požadovat souhlas a nestalo se tak.

Další možností je vložit všechnu důvěru ohledně kontroly do člověka, provádějící konkrétní úkol. To ovšem předpokládá, že je vybraný člověk schopen rozpoznat odchylky od plánu, uvědomuje si závažný problém a je schopen ho jasně formulovat. Protože všechny tyto vlastnosti má jen malé procento pracovníků, nelze tuto metodu většinou použít.

3.2.4.1 Zprávy

Nejlépeším způsobem kontroly je zjišťovat stavy projektových úkolů. Informace o stavech lze předkládat členy týmů, kteří na úkolech pracují, v podobě zpráv nebo hlášení na kontrolních schůzkách. Zprávy by se měly týkat všech tří dimenzí trojimperativu. Manažer projektu ovšem nemůže spoléhat pouze na zprávy. Mohou totiž obsahovat řadu nepřesností. Zavádějící je také posuzovat stav projektu na základě procent dokončenosti jednotlivých úkolů. Tvrzení, že úkol je hotov z 80% nemusí zdaleka znamenat, že bude brzo dokončen. Manažer by se tak měl soustředit hlavně na ukončené a dodané výstupy úkolů.

3.2.4.2 Kontrolní schůzky

Dalším způsobem kontroly je pořádání kontrolních schůzek. Existují dva druhy. Periodické a tematické. Periodické jsou naplánovány v pravidelných intervalech a hlavním účelem je zjištění postupu prací na projektu. Schůzky mohou často rozpoznat problém, který by se mohl v budoucnu stát katastrofou. Schůzky jsou také velmi dobrou příležitostí informovat o stavu projektu jak pracovníky, tak vedení nebo zákazníka.

Tematické schůzky jsou pořádány nepravidelně a jejich obsahem může být například předběžný návrh, výroba, kompletace před odesláním nebo zákazník. Tyto schůzky jsou většinou zaneseny do smlouvy se zákazníkem a měly by být zaneseny i do plánu projektu.

Většina této kapitoly čerpá z [1].

3.2.5 Uzavření

Projekt může skončit ještě před svým řádným uzavřením, například odebráním zdrojů na jiný projekt nebo úplným zrušením vlivem nedodržení například časového nebo rozpočtového plánu.

Řádné a připravované uzavření projektu vyžaduje zvýšené úsilí. Je možné, že se budou muset konat častěji kontrolní schůzky nebo budou vlivem zpoždění přetěžovány zdroje.

3.2.5.1 Akceptační test

Při předání produktu zákazníkovi se provádí akceptační test. Ten kontroluje splnění všech požadavků, které byly zaneseny do smlouvy ve fázi uzavření kontraktu. Je důležité, aby všechny tyto požadavky byly objektivní a tím pádem i jednoznačně testovatelné. Také samotný akceptační test by měl být jasně definován již ve smlouvě.

V projektu by se také nikdy neměly vyskytovat služby, které nejsou specifikovány. Pro takové služby totiž neexistují jasná akceptační kritéria a při předání by s nimi zákazník nemusel být spokojen.

Projekt může končit dodávkou produktu. V mnoha případech je ovšem požadována podpora nebo záruka i po dodání. Tato skutečnost také musí být zachycena ve smlouvách již na začátku projektu [1].

3.2.5.2 Dokumentace

Ve velkém množství případů je požadována k výslednému produktu dokumentace například ve formě závěrečné zprávy, manuálů nebo instruktážních příruček. Tvorba dokumentace je časově náročná a tak by se měla ve smlouvě definovat jako samostatná položka.

Tvorba dokumentace na konci projektu je obtížná. Je to z důvodu neochoty technických odborníků dokumentaci sepisovat a také se může stát, že pracovníci, kteří mají potřebné informace, mohli být v průběhu projektu přeřazeni na jinou práci a na projektu již nepracují.

Řešením je vytvoření všech potřebných šablon pro dokumentaci již na samotném začátku a jejich postupné tvoření během projektu [1].

3.2.5.3 Závěrečné zhodnocení

Každý projektový vedoucí by měl po ukončení projektu dát dohromady jeho testament.

To znamená prozkoumat celý průběh projektu a sepsat o něm podrobnou zprávu. Účelem této činnosti je zaznamenat všechny užitečné informace pro pozdější použití u dalších projektů a hlavně se poučit ze situací, které během projektu nastaly. Kvůli objektivnosti by měla toho hodnocení provádět nějaká nezainteresovaná osoba, která se na projektu příliš nepodílela [1].

3.3 Programová podpora

Software pro podporu řízení projektu dokáže ušetřit spoustu práce a času. Většina obsahuje nástroje pro plánování, sledování a řízení. Na trhu je celá řada produktů různé kvality. Mezi nejznámější patří Microsoft Project, SureTrack Project Manager od firmy Primavera nebo Imendio Planner, který je zdarma pro operační systém Linux.

3.3.1 Microsoft Project

Nejpoužívanějším je Microsoft Project. Důvodů je několik. Je to například jednoduché a přehledné uživatelské rozhraní shodné s produkty řady Microsoft Office, podpora velkých projektových týmů, možnost spolupráce s ostatními programy firmy Microsoft, velké množství kvalitní dokumentace a mnoho užitečných funkcí.

Microsoft Project verze 2003 je vydáván ve čtyřech edicích. Jsou to Project Standard, Project Professional, Project Server a Project Web Access.

Project Standard neobsahuje nástroje pro týmovou spolupráci, jako je například zasílání e-mailových zpráv nebo podpora systému Project Server. Slouží tedy výhradně pro práci jednotlivce. Jinak ale nabízí všechny potřebné funkce pro plánování, sledování a řízení projektu.

Project Professional spolu s Project Server nabízí vlastnosti pro týmy a velké organizace. Project Server je serverový produkt, který je centrem informací systému Project. Díky němu jsou informace dostupné všem potřebným osobám.

Project Web Access umožňuje pohled na informace reprezentované systémem Project Server a to pomocí webového rozhraní, dostupného z internetového prohlížeče.

Microsoft Project nám umožňuje snadno vytvářet hierarchickou strukturu činností, generuje z ní různé časové diagramy, na kterých také zobrazuje průběh projektu. Pomocí možnosti zadání nákladů a zdrojů na úkoly odhaduje náklady celého projektu. Informace o projektu umí publikovat na webu a vytváří nastavitelné tiskové sestavy.

Pro usnadnění zadávání údajů a porozumění správnému způsobu zadávání obsahuje od verze 2003 prvky jako jsou Průvodce projektem, které práci výrazně zjednoduší a vedou uživatele správnou cestou [11].

3.4 Řízení projektu v oblasti IT

Řízení projektu v oblasti IT je obzvlášť důležitou činností. Na počátcích vývoje softwaru nebyla žádná taková činnost ani potřeba. Programy byly malého rozsahu a nebyly obzvlášť složité. S velkým rozvojem oblasti IT se ale začaly i zvětšovat programy a bez systematického řízení vývoje se tato oblast již neobešla. Převážné množství větších projektů v 60. letech se potýkalo s problémy ohledně včasného dokončení, překročení nákladů nebo nedostatečné kvality výsledného produktu. Tato neúnosná situace vyústila v softwarovou krizi. Opatřením proti krizi bylo zavedení softwarového inženýrství. Softwarové inženýrství je systematický přístup k vývoji, nasazení a údržbě softwaru. Zahrnuje tři hlavní oblasti: management projektu, techniky (pro analýzu, návrh, programování, testování a údržbu softwaru) a výchovu softwarového inženýra [4].

O přetrvávající softwarové krizi ovšem hovoří například studie Standish Group Report, která byla zveřejněna v USA v roce 1995. Vybrané výsledky jsou zobrazeny v tabulkách 3.1, 3.2 a 3.3 [4].

Překročení nákladů o	Projektů
méně než 20%	15,5%
21-50%	31,5%
51-100%	29,6%
101-200%	10,2%
201-400%	8,8%
více než 400%	4,4%

Tabulka 3.1: Překračování nákladů softwarových projektů [4]

Překročení času o	Projektů
méně než 20%	13,9%
21-50%	18,3%
51-100%	20,0%
101-200%	35,5%
201-400%	11,2%
více než 400%	1,1%

Tabulka 3.2: Překračování času softwarových projektů [4]

Výsledná funkčnost	Projektů
méně než 25%	4,6%
25-49%	27,2%
50-74%	21,8%
75-99%	39,1%
100%	7,3%

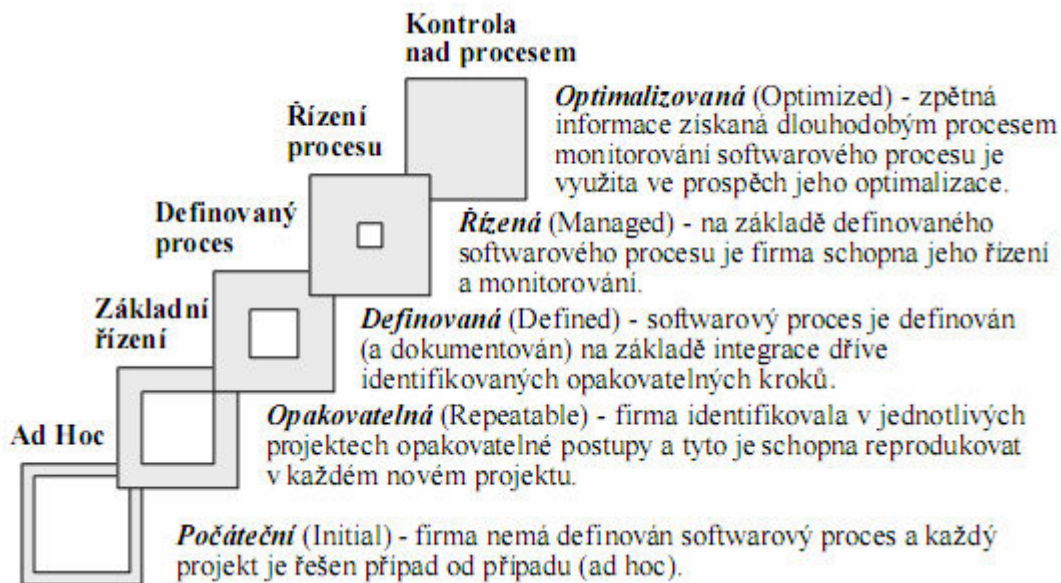
Tabulka 3.3: Dosažená funkčnost softwarových projektů [4]

3.5 Vypělost projektového řízení v IT

Vypělost projektového řízení souvisí s řízením kvality. Tato činnost prolíná většinu procesů projektového řízení. Řízení kvality by ovšem mělo být odděleno od projektového řízení. Mělo by mít svůj vlastní plán i rozpočet.

Velké úsilí bylo věnováno standardizaci řízení kvality. V USA vznikl model CMM – Capability Maturity Model. Byl vyvinut v 80. letech na žádost Ministerstva obrany, aby pomohl organizacím určit stupeň procesní vypělosti, nejkritičtější problémy v oblasti kvality softwaru a zvolit si tak správné řešení pro zvýšení kvality na požadovaný stupeň [5]. Úrovní vypělosti projektového řízení definuje celkem pět. Od nejnižší po nejvyšší. Tento model se brzy rozšířil po celém světě a stal se tak prakticky standardem pro celý proces řízení kvality. Je také používán pro vládní projekty po celém světě. V současnosti požadují některé země pro vládní zakázky dosažení alespoň třetí úrovně [6].

Model také obsahuje sadu doporučených technik v klíčových procesních oblastech, jejíž použití má za následek zvyšování úrovně vypělosti vývoje softwaru. Slouží jako návod tvůrcům a dodavatelům softwarových produktů, jak dostat pod kontrolu určité procesy vývoje a údržby softwaru a tedy jak zvýšit úroveň procesní vypělosti [5].



Obrázek 3.7: Úrovně vspělosti modelu CCM [5]

Model CMM byl vyvíjen od roku 1987 do roku 1997. V roce 2001 byl nahrazen novým modelem CMMI. Ten je v současné době ve verzi 1.2. Hlavním cílem nového modelu je vylepšit použitelnost CMM a spojit mnoho různých modelů do jednoho rámce [7].

Mezinárodní organizace pro standardizaci přijala za vlastní sérii standardů ISO 9000. Ty se týkají všech průmyslových odvětví. V současnosti jsou standardy sjednoceny do jediného označovaného jako ISO 9001:2000.

4 Metoda logického rámce

Metoda logického rámce tvoří základ jak pro analýzu problémů, tak i pro definování cílů a stanovení konkrétních aktivit k řešení těchto problémů. Přehledně zobrazuje cíle (účel), konkrétní výstupy z projektu, činnosti a prostředky umožňující jejich naplnění. Dále také ukazuje ukazatele, pomocí kterých je možné výsledky projektu vyjádřit a podmínky a rizika, které realizaci projektu ovlivní, případně ohrozí [3]. Spolu s projektovou fiší tvoří listinu základního vymezení projektu, která je základní dokumentací projektu.

4.1 Listina základního vymezení projektu

Listina základního vymezení projektu je základní dokumentací a může být tvořena dvěma dokumenty:

- Logický rámec projektu
- Projektová fiše

Popis a význam obou dokumentů, stejně jako postup pro jejich vytvoření, je uveden v následujících kapitolách.

4.2 Význam metody

Metoda logického rámce je metoda, jak přehledně zmapovat záměry, očekávání a uvést je do souladu s konkrétními činnostmi a výstupy při realizaci projektu.

Logický rámec není důležitý jen ve fázi přípravy projektu, ale je klíčovým nástrojem také při implementaci a hodnocení. Je využíván pro svoji jednoduchost, jednoznačnost a hlavně jednotnost popisu všech projektů. Usnadňuje také práci hodnotitelům projektu [8].

Proces sestavování logického rámce je stejně důležitý jako jeho obsah. Rámec projektu by měl být sestaven již na začátku celého plánovacího procesu a měl by být připravován za účasti všech zúčastněných a dotčených osob nebo institucí. Návrh logického rámce by měl být sestaven ještě před vypracováním projektové fiše. Dokončen by měl být spolu s ní [3].

4.3 Struktura logického rámce

<i>Sloupec 1 – Intervenční (vertikální logika)</i>	<i>Sloupec 2 – Objektivně ověřitelné ukazatele</i>	<i>Sloupec 2 – Zdroje (informací) k ověření</i>	<i>Sloupec 4 – Rizika/ předpoklady</i>
<p>Celkový cíl (1)</p> <p><i>Jaký je celkový širší cíl, k němuž projekt přispěje? Celkový cíl musí přitom odpovídat příslušné prioritě programu.</i></p>	<p>Objektivně ověřitelné ukazatele (2)</p> <p><i>Jaké jsou klíčové ukazatele vztahující se k celkovému cíli?</i></p>	<p>Zdroje ověření (3)</p> <p><i>Jaké jsou zdroje informací pro tyto ukazatele?</i></p>	
<p>Účel projektu (4)</p> <p><i>Jaké jsou specifické cíle, kterých chce projekt dosáhnout?</i></p> <p><i>Proč byl projekt navržen?</i></p> <p><i>Jaký je jeho účel? Jaké jsou předpokládané efekty a přínosy projektu? Jaká zlepšení a změny projekt přinese?</i></p> <p><i>Poznámka: Pokud je účelem projektu něco, co nebude dosaženo jen projektem samotným, je nutné, aby ostatní nezbytné aktivity vně projektu byly uvedeny v oddíle 11.</i></p>	<p>Objektivně ověřitelné ukazatele (5)</p> <p><i>Jaké jsou kvantitativní nebo kvalitativní ukazatele, které ukazují, zda a do jaké míry budou specifické cíle dosaženy? Jakými ukazateli lze měřit, zda a v jaké míře dosáhne projekt předpokládané účinky a efekty?</i></p>	<p>Zdroje ověření (6)</p> <p><i>Jaké jsou zdroje informací, které již existují nebo mohou být soustředěny?</i></p> <p><i>Jaké jsou metody nutné k získání těchto informací?</i></p>	<p>Předpoklady (7)</p> <p><i>Jaké jsou faktory a podmínky mimo přímý vliv projektu, které jsou nutné k dosažení těchto cílů?</i></p> <p><i>Jaká rizika je nutné brát v úvahu?</i></p>
<p>Výsledky (8)</p> <p><i>Jaké jsou konkrétní výsledky, které mají zajistit dosažení specifických cílů?</i></p>	<p>Objektivně ověřitelné ukazatele (9)</p> <p><i>Jaké ukazatele změní, zda projekt dosáhl předpokládaných výsledků (oddíl 8)?</i></p>	<p>Zdroje ověření (10)</p> <p><i>Jaké jsou zdroje informací pro tyto ukazatele?</i></p>	<p>Předpoklady (11)</p> <p><i>Jaké vnější faktory a podmínky se musí navíc uskutečnit kromě výsledků a výstupů projektu (oddíl 8), aby bylo dosaženo účelu projektu (oddíl 4)?</i></p>
<p>Aktivity (12)</p> <p><i>Jaké klíčové aktivity musí být provedeny a v jakém pořadí, aby bylo dosaženo očekávaných výsledků (oddíl 8)?</i></p>	<p>Prostředky (13)</p> <p><i>Jaké prostředky jsou nutné k realizaci těchto aktivit, např.: personál, vybavení, školení, studia, dodávky, provozní zařízení atd.?</i></p>	<p>Zdroje (14)</p> <p><i>Tato část logického rámce se podle některých metodik nevyplňuje. Pokud se vyplní, je možné na tomto místě uvést zdroje, které má žadatel o dotaci před zahájením projektu k dispozici, zatímco v oddíle 13 se pak uvedou jen ty prostředky, jejichž získání je předmětem projektu.</i></p>	<p>Předpoklady (15)</p> <p><i>Jaké podmínky jsou mimo přímý vliv projektu, které musí existovat pro realizaci plánovaných akcí k dosažení výsledků (oddíl 8)?</i></p>
			<p>Předběžné podmínky (16)</p> <p><i>Jaké předběžné podmínky jsou vyžadovány před zahájením projektu?</i></p>

Obrázek 4.1: Struktura logického rámce [3]

Struktura logického rámce je uvedena na obrázku 4.1 [3]. Skládá ze čtyř sloupců, které vyjadřují:

- Vertikální logiku projektu
- Objektivně ověřitelné ukazatele
- Zdroje (informací) k ověření
- Rizika/předpoklady, které podmiňují dosažení výsledků a cílů projektu [9]

4.3.1 Hlavička logického rámce

Hlavička logického rámce není na obrázku 4.1 zobrazena, avšak měla by být uvedena před samotným logickým rámcem a měla by obsahovat název předkládaného projektu, základní informace o předkladateli (název, adresa), označení programu, do kterého bude projekt předkládán a celkové náklady projektu s vyčleněním celkových přijatelných nákladů [8].

4.3.2 Vertikální logika

Vertikální logika je vztahem příčina-důsledek mezi aktivitami a cíli na rozdílných úrovních. Každá úroveň by měla logicky vést k úrovni, která je o jeden stupeň výše. Údaje z tohoto sloupce se uvádějí v projektové fiši.

4.3.2.1 Celkový cíl (1)

Celkový cíl naznačuje, jak projekt zapadá do kontextu programu a měl by odpovídat jeho prioritám. Jde tedy o hlavní cíl, o který se snažíme, ne o cíl projektu.

4.3.2.2 Účel projektu (4)

Účel projektu odpovídá na otázku, proč by měl být projekt realizován. Jsou specifické cíle, kterých chce projekt dosáhnout. Jde o změnu, kterou chceme realizací projektu dosáhnout, i když to není třeba možné pouze prostřednictvím realizace tohoto individuálního projektu. Tuto změnu tedy nemůžeme často ani sami provést, ale o její naplnění se snažíme pomocí výsledků projektu.

4.3.2.3 Výsledky (8)

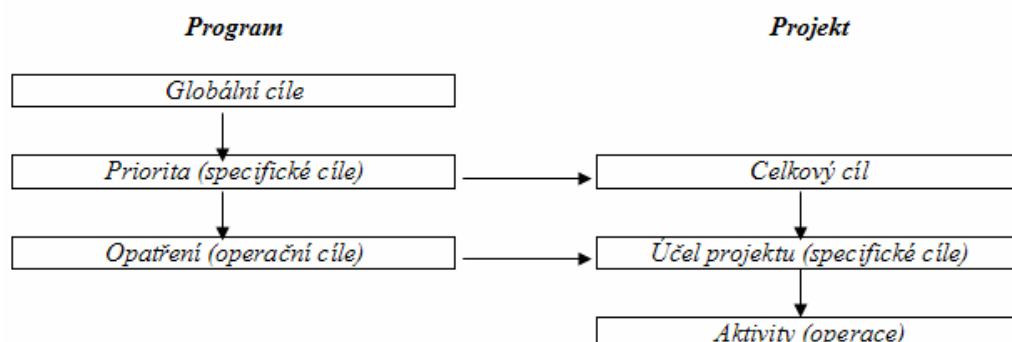
Očekávané výsledky ukazují, čeho fyzická osoba nebo instituce nakonec prostřednictvím projektu dosáhne a za co je také zodpovědná. Jedná se o konkrétní přímé výsledky projektu, které mají pomoci dosáhnout účel projektu. Pokud má projekt více částí, je nutné je na této úrovni popsat a očíslovat.

4.3.2.4 Aktivity (12)

Aktivity ukazují, jak se bude projekt dělat a jaké prostředky jsou k tomu potřebné. Ke každému výsledku stanovíme aktivity, které podle nás povedou k jejich dosažení. Tyto aktivity mohou být základem pro hierarchickou strukturu činností (WBS).

4.3.2.5 Vztah cílů programu a projektu

Při přípravě projektu musíme mít na paměti, že může být součástí programu a musíme tak prokázat soulad s jeho cíli. Návaznost cílů programu a projektu ilustruje obrázek 4.2 [9].



Obrázek 4.2: Návaznost cílů programu a projektu [9]

Tato kapitola čerpala z [3] a [8].

4.3.3 Objektivně ověřitelné ukazatele

Ve druhém sloupci logického rámce se uvádějí objektivně ověřitelné ukazatele celkových cílů, účelu projektu a výsledků. Pro aktivity se uvádějí ukazatele vstupů, tj. prostředků nebo zdrojů. Ukazatele jsou základem pro určení efektivity a účelnosti projektu [9].

4.3.4 Zdroje (informací) k ověření

V tomto sloupci by měly být uvedeny možné zdroje informací k ověření průběhu a výsledků realizace projektu, včetně údajů kde jsou tyto informace uloženy, jaké existují metody k jejich získání atp. [9].

4.3.5 Rizika/předpoklady

Tento sloupec obsahuje stručný popis rizik, jejichž zdroji jsou externí faktory. Na taková rizika nemá projekt vliv, ale mohou ovlivnit jeho realizaci i udržitelnost. V posledním řádku jsou předběžné podmínky, které musí být splněny, aby projekt začal. Je velmi důležité vyplňování tohoto sloupce nepodcenit. Při neúspěchu projektu je totiž možné na tyto uvedené údaje poukázat a požadovat zmírnění sankcí za nesplnění cílů projektu [9].

4.4 Proces vytváření a čtení

4.4.1 Vytváření

Při vytváření logického rámce máme dvě možnosti, jak postupovat. Těmi jsou postup shora dolů nebo zezdola nahoru. Čísla položek v závorkách korespondují s označením v obrázku 4.1.

4.4.1.1 Postup shora dolů

Při tvorbě logického rámce seshora dolů se doporučuje začít položkou účel projektu (4), poté pokračovat očekávanými výsledky (8) a předpoklady/rizika (11) potřebnými k získání účelu projektu (4). Poté následují aktivity (12) a předpoklady/rizika (15) potřebné k získání očekávaných výsledků (8). Dále se vyplní položka celkový cíl (1) a poslední položka předpokladů/rizik (7). Nakonec se doplní u všech úrovní objektivně ověřitelné ukazatele a zdroje (informací) k ověření, tj. prostřední dva sloupce [10].

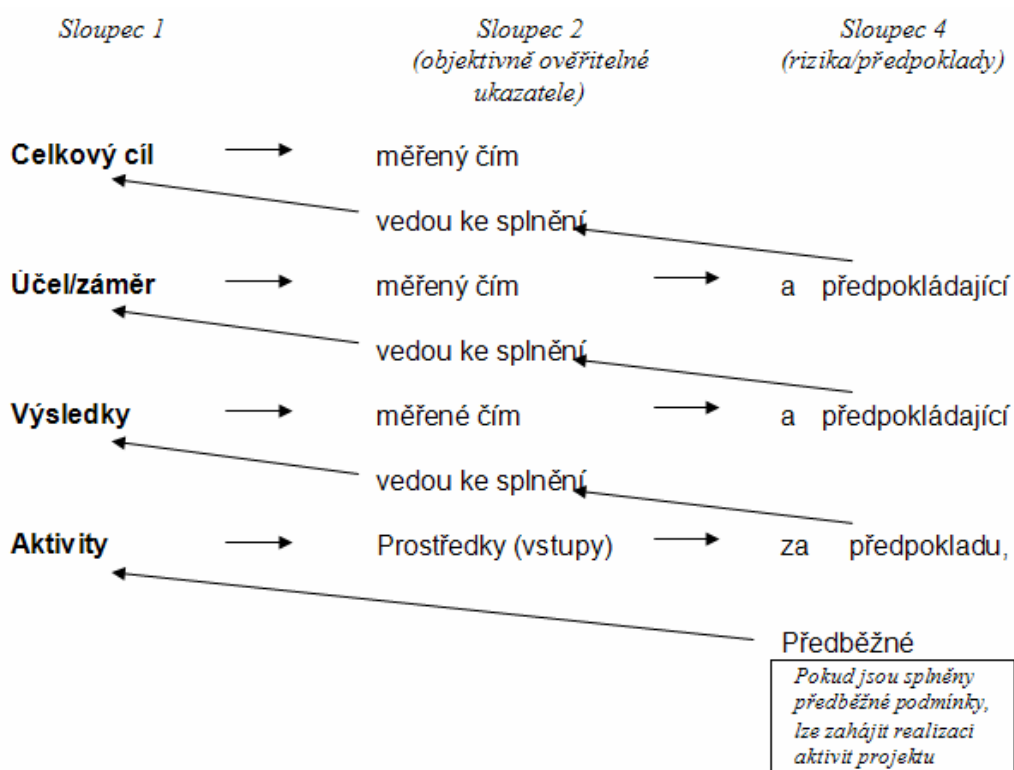
4.4.1.2 Postup zezdola nahoru

Při tvorbě logického rámce zezdola nahoru je nejlepší vyplňovat od položky aktivity (12) k předpokladům/rizikům (15). Tyto předpoklady se vztahují k očekávaným výsledkům (8). Dále se pokračuje stejným způsobem přes účel projektu (4) až k celkovému cíli (1) [10].

4.4.2 Čtení

Logický rámec by měl být čten tak, jak je uvedeno na obrázku 4.3 [9]. Tedy vzestupně ve směru šipek.

Pomocí správného přečtení logického rámce se můžeme přesvědčit i o jeho správném předchozím vytvoření.



Obrázek 4.3: Čtení logického rámce [9]

4.5 Projektová fiše

Projektová fiše tvoří spolu s logickým rámcem základní dokumentaci projektu – listinu základního vymezení projektu. Typická osnova projektové fiše je následující:

- Název projektu
- Lokalizace projektu (stát, okres, obec, region)
- Předkladatel projektu (žadatel) a další partneři podílející se na projektu
- Cíle projektu
- Popis projektu
- Časový harmonogram projektu
- Rozpočet projektu
- Rovné příležitosti
- Posouzení vlivu projektu na životní prostředí
- Ekonomické a finanční vyhodnocení projektu
- Předpoklady a možná rizika realizace projektu
- Přípravenost projektu k realizaci [3]

Přílohou projektové fiše je logický rámec projektu.

5 Program realizující podporu definování pomocí metody logického rámce

Součástí mé diplomové práce je program, realizující podporu definování projektu pomocí metody logického rámce. Mezi jeho základní funkce patří vytvoření a export rámce ve vhodném formátu pro program Microsoft Project. Podrobněji jsou požadavky na program uvedeny v následující kapitole.

5.1 Analýza požadavků

Program umožňuje vytvoření logického rámce, jeho uložení do souboru a opětovné načtení. Dále nabízí export do vhodných formátů, z nichž nejdůležitějším je export činností projektu do souboru aplikace Microsoft Project. Kromě těchto funkcí je možné upravovat vzhled rámce, umožnit vytisknutí nebo provádění kontroly vyplnění pomocí jeho správného přečtení.

5.1.1 Základní funkce

Mezi základní funkce programu patří:

- Vytvoření logického rámce samostatně nebo s průvodcem
- Kontrola vyplnění pomocí metody čtení logického rámce
- Ukládání a načítání logického rámce
- Tisk
- Export do programu Microsoft Project
- Export do dalších formátů
- Vytvoření šablon, definujících vzhled
- Systém nápovědy

5.1.1.1 Vytvoření logického rámce

Vytvoření logického rámce je možné jak přímým vepsáním informace do položky logického rámce, tak pomocí průvodce. Průvodce pomůže uživateli při vyplňování pokládáním vhodných otázek. Otázky jsou pokládány na základě metody vyplňování shora dolů a zezdola nahoru.

5.1.1.2 Kontrola vyplnění

Kontrolu správného vyplnění může uživatel provést pomocí textu, který je vygenerován na základě metody čtení logického rámce a který tak logicky spojí informace z jednotlivých buněk. Přečtením tohoto textu může uživatel zjistit nedostatky a následně je opravit.

5.1.1.3 Ukládání a načítání

Aplikace umožňuje ukládat a načítat informace o logickém rámci do souboru. Soubor musí obsahovat jak textový obsah, tak i vzhled rámce. Uživatel si může zvolit místo i název ukládaného souboru.

5.1.1.4 Tisk

Metoda logického rámce je užitečná pro svou vlastnost přehledného a kompaktního zobrazení informací, a tak je důležitou součástí možnost tisku. Před samotným tiskem je možno nechat zobrazit náhled, vybrat tiskárnu a upravit její nastavení.

5.1.1.5 Export do programu Microsoft Project

Program umožňuje export zadaných projektových činností do formátu MPX, který slouží pro výměnu informací s aplikací Microsoft Project. Vytvoří se tím základ hierarchické struktury činností, který může být dále v aplikaci Microsoft Project upravován a rozšiřován.

5.1.1.6 Export do dalších formátů

Kromě exportu do formátu MPX je umožněn export do dalších formátů, konkrétně XML, který lze použít pro výměnu informací s různými aplikacemi. Dále je to formát HTML, ve kterém je logický rámec exportován včetně jeho vzhledu a který slouží pro jeho prezentaci.

5.1.1.7 Vytvoření šablon, definujících vzhled

Protože vzhled logického rámce, stejně jako popisky položek, se jak v publikacích, tak i organizacích, které ho používají, mírně liší, nabízí program možnost vytvoření šablon, které budou základní vlastnosti a vzhled definovat. Šablony je možno ukládat a poté vlastnosti vybrané šablony aplikovat na aktuální logický rámec.

5.1.1.8 Systém nápovědy

Pro začínající uživatele je důležitý propracovaný systém nápovědy, který spolu s průvodcem při vyplňování slouží ke správnému pochopení metody. Díky těmto pomocným prvkům může sloužit tento program i k výukovým účelům, k pochopení vytváření a čtení logického rámce.

5.1.2 Uživatelské rozhraní

Kvůli snadné práci s programem by mělo být uživatelské rozhraní grafické, jednoduché a přehledné. I když se předpokládá, že s aplikací budou pracovat zejména uživatelé, kteří danou problematiku

ovládají, mělo by kromě hlavních ovládacích prvků obsahovat prvky, usnadňující správné vyplnění rámce, které budou vhodně zobrazovat instrukce pro vyplňování.

Kromě těchto požadavků by mělo uživatelské rozhraní působit dobře i graficky. Protože jsou uživatelé zvyklí na určitý vzhled formulářových aplikací, znepříjemnila by jim velká změna práci s programem. Proto bude vhodné zpříjemnit užívání aplikace doplněním ilustračních obrázků do formulářových oken, ale vzhled standardních prvků by měl zůstat nezměněn.

6 Použité technické prostředky

Aplikace je implementována v jazyku C# ve vývojovém prostředí Microsoft Visual Studio 2005. Důvodem tohoto výběru je možnost snadné tvorby uživatelského rozhraní a výborná podpora práce s XML dokumenty. S výhodou lze v tomto prostředí využívat také obsažené komponenty nebo komponenty vytvořené vývojáři, které jsou volně dostupné na specializovaných internetových stránkách. Dalším velkým důvodem pro tuto volbu je kvalitní dokumentace do firmy Microsoft a početná komunita uživatelů kolem tohoto vývojového prostředí.

Program lze spouštět v rozhraní .NET Framework 2.0 a vyšším v operačním systému Windows.

6.1 Microsoft Visual C#

Jazyk C# je výkonný, objektově orientovaný jazyk od firmy Microsoft. Lze ho využít k vytváření formulářových aplikací ve Windows, webových aplikací a stránek, databázových programů nebo softwaru pro mobilní zařízení.

6.1.1 Historie

Začátky jazyku C# se datují do roku 1999, kdy sestavil Anders Hejlsberg tým, který vytvořil nový jazyk, nazvaný Cool. V roce 2000 byl přejmenován na C# a zveřejněn ve verzi 1.0 spolu s .NET Frameworkem 1.0. V roce 2005 byla zveřejněna verze 2.0 společně s .NET Frameworkem 2.0. Tato verze obsahovala nové vlastnosti, které ještě více podporovaly objektově-orientované programování. Na konci roku 2007 vyšla verze 3.0 a zatím poslední 3.5. Tato verze přináší poměrně mnoho změn a nových vlastností.

6.1.2 Popis

Jazyk C# je jednoduchý, moderní, objektově-orientovaný jazyk. Inspiruje se jazyky C++ a Java. Z nich se snaží využít nejlepších vlastností a naopak odstranit vlastnosti, které programátorům znepříjemňují tvorbu aplikací. Pro usnadnění vývoje v tomto jazyku slouží například detekce překročení hranic pole, detekce použití neinicializovaných proměnných nebo automatická správa paměti a rušení objektů.

Program v jazyku C# je překládán téměř ve všech případech do mezijazyka CIL (Common Intermediate Language). Tento kód se poté spouští v běhovém prostředí. Tím bývá nejčastěji .NET Framework, který zajišťuje běh programů v operačním systému Windows. Nezávisle na .NET Framework jsou vyvíjena prostředí Mono a DotGNU, implementující běhové prostředí pro Unixové systémy.

6.2 Microsoft Visual Studio 2005

Microsoft Visual Studio je integrované vývojové prostředí pro vývoj aplikací a to jak formulářových, tak i webových aplikací a stránek, webových služeb a dalších. Podporuje mnoho programovacích jazyků.

Toto prostředí zahrnuje textový editor s podporou systému IntelliSense, který programátorovi nabízí interaktivní výpisy vlastností a metod objektů. Dále je součástí překladač s možností překladu do binárního kódu nebo mezikódu. Obsahuje také nástroj pro návrh formulářů pro aplikace s grafickým uživatelským rozhraním, pro návrh webových aplikací, schématu databáze a mnoho dalších, které napomáhají efektivnímu a rychlému vývoji aplikací.

6.3 .NET Framework

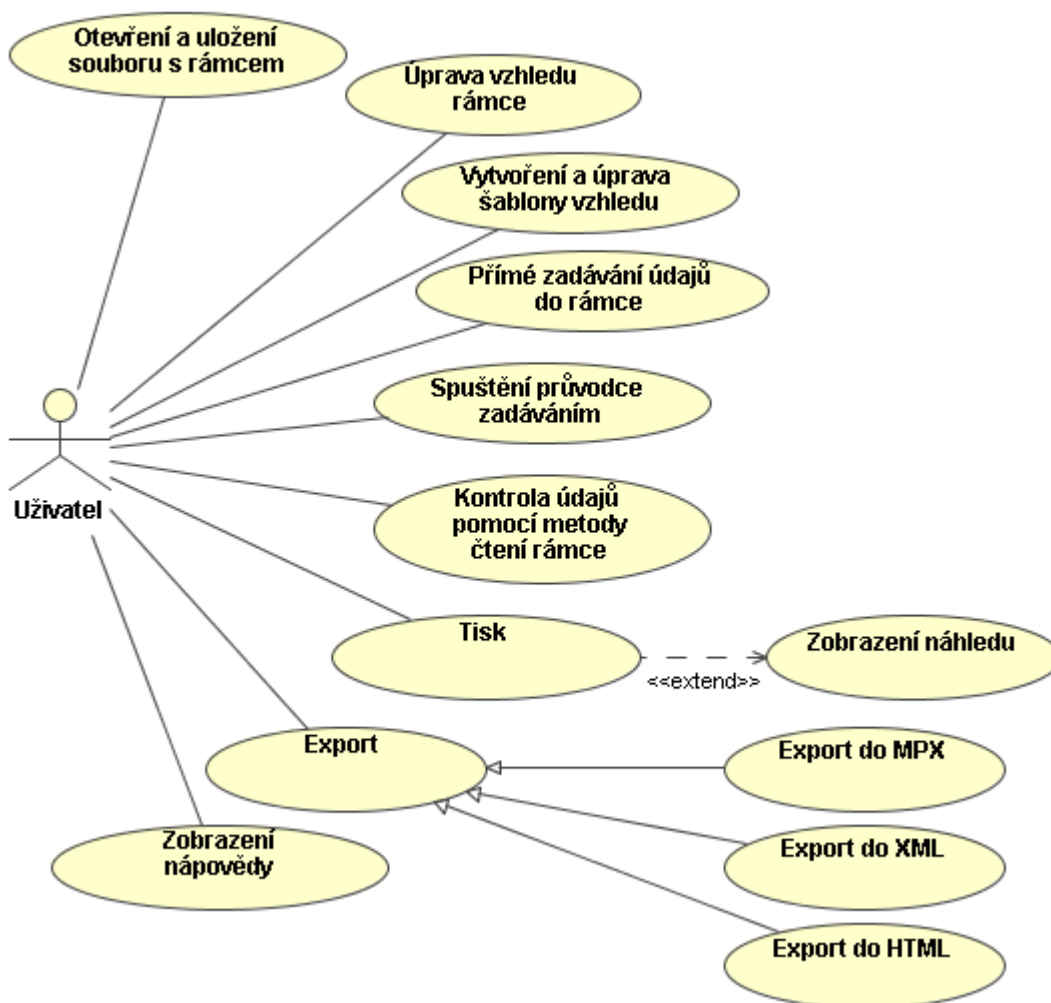
.NET Framework je prostředí od firmy Microsoft, potřebné pro běh .NET aplikací na osobních počítačích s operačním systémem Windows a to od verze Windows 98. Nabízí jak spouštěcí rozhraní, tak všechny potřebné knihovny.

Nejnovější verze 3.5 byla vydána v roce 2007. Pro vývoj .NET aplikací vytvořila firma Microsoft integrované vývojové prostředí Microsoft Visual Studio.

7 Návrh programu

V následujících kapitolách je uveden návrh programu pomocí diagramů jazyka UML, konkrétně diagramu případů použití a diagramu tříd, společně s jejich popisem.

7.1 Diagram případů použití



Obrázek 7.1: Diagram případů použití

7.1.1 Popis důležitých případů použití

Export logického rámce umožňuje jeho uložení ve formátu XML, HTML a MPX. Zvláště důležitým formátem je export pro program Microsoft Project. Ten je ve formátu Aplikace Microsoft Project Exchange MPX. Pomocí něho je v MS Projectu generována hierarchická struktura činností WBS. Při exportu je uživateli nabídnuta možnost zvolit název a umístění exportovaného souboru a v případě

exportu do HTML i možnost exportovat také název rámce, jméno autora a datum vygenerování exportovaného souboru.

Spuštění průvodce zadáváním umožňuje vyplňovat logický rámec pomocí odpovědí na otázky zadané průvodcem. To zajišťuje správné pochopení logického rámce a učí uživatele rámec správně vyplňovat. Po spuštění průvodce je zobrazeno okno, které dává uživateli krok po kroku instrukce, jak správně vyplnit položky logického rámce. Po dokončení posledního kroku jsou změny uloženy do zobrazeného rámce. Uživatel si může zvolit, jakým způsobem ho průvodce při vyplňování povede a to buď metodou zezdola nahoru nebo shora dolů.

Kontrola čtením rámce logicky spojuje vyplněné údaje do souvislého textu, na kterém je možno ověřit, zda byly informace zadány správně. Spojení se provádí na základě metody čtení logického rámce. Po zvolení kontroly je uživateli zobrazen text, jehož přečtení pomůže odhalit nedostatky ve vyplněném rámci.

Vytvoření šablon pro vzhled rámce umožňuje přizpůsobit popisky, názvy a další vlastnosti rámce podle zvyklostí a potřeb uživatele. Šablona se vytváří z vlastností otevřeného logického rámce. Ty si může uživatel upravit podle svých potřeb a potom toto nastavení uloží do šablony pod určitým názvem. Zvolit si může jak název šablony, který pak bude zobrazován při jejím výběru, tak i název souboru, do kterého bude šablona uložena. Uložené šablony si pak může uživatel prohlížet a podle potřeby vzhled vybrané na otevřený rámec aplikovat.

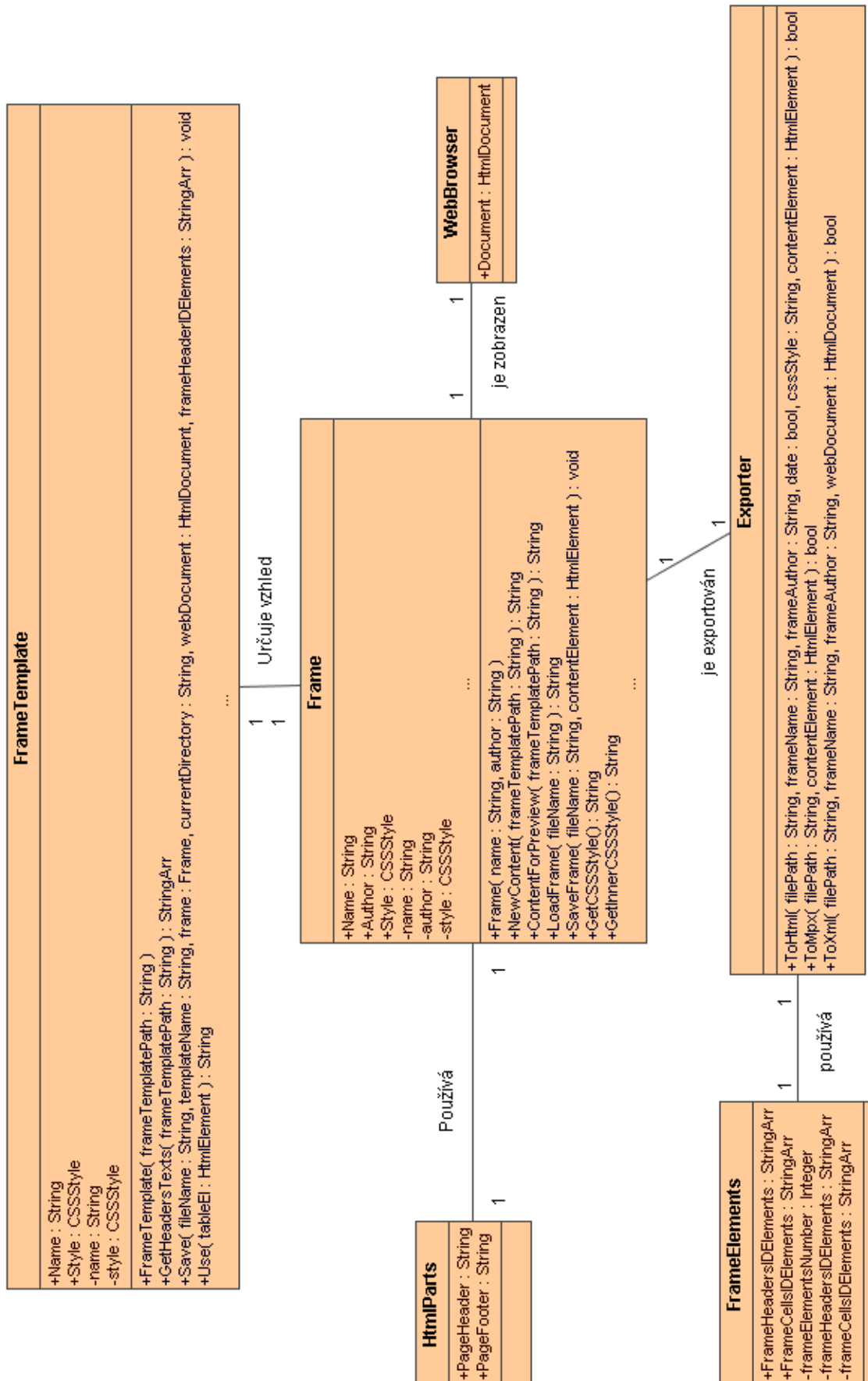
7.2 Diagram tříd

Diagram tříd byl navržen tak, aby od sebe odlišoval dva typy tříd. Prvním typem jsou třídy, které obsahují a zpracovávají data. Druhým typem jsou třídy, které slouží k realizaci uživatelského rozhraní a mají funkci zobrazování dat a obsluhování událostí, vyvolaných uživatelem.

Původní diagram tříd byl v průběhu realizace programu mírně upravován, ve většině případů šlo pouze o přidání atributu nebo metody, několikrát byla přidána třída nová. Základní struktura původního návrhu zůstala ovšem nezměněna.

7.2.1 Datový model

Obrázek 7.2 zobrazuje diagram tříd, obsahující takové třídy, jejichž objekty obsahují a zpracovávají data. Kvůli přehlednosti a srozumitelnosti byly vybrány pouze důležité třídy a jejich atributy a metody.



Obrázek 7.2: Diagram tříd – datový model

7.2.1.1 Popis důležitých tříd

Třída *Frame* je jednou z nejdůležitějších tříd aplikace. V objektech této třídy jsou ukládány informace o logickém rámci, jako jsou jeho název, autor a vzhled. Obsah logického rámce je ukládán v objektu třídy *WebBrowser* ve formátu HTML dokumentu. Třída *Frame* poskytuje základní metody pro práci s rámcem jako je jeho vytvoření, uložení, načtení, vytvoření obsahu pro náhled před tiskem a další.

Třída *FrameTemplate* zajišťuje možnost ukládat vzhled logického rámce do šablony a poté aplikovat vlastnosti šablony na otevřený rámeček. Do šablony lze ukládat vlastnosti písma a pozadí jak pro hlavičky, tak pro položky logického rámce a texty hlaviček.

Exporter je třídou, realizující export rámce do formátů MPX, XML a HTML.

Třídy *HtmlParts* a *FrameElements* jsou pomocné třídy, poskytující opakovaně používané a potřebné informace, jako je hlavička a patička html stránky nebo názvy identifikátorů buněk logického rámce.

7.2.2 Model uživatelského rozhraní

Obrázek 7.3 zobrazuje diagram tříd, realizujících grafické uživatelské rozhraní. Objekty těchto tříd zobrazují data uživateli a obsluhují události. Obrázek zobrazuje pouze nejdůležitější třídy a vybrané atributy a metody. Kompletní model by byl značně nepřehledný a nesrozumitelný.

7.2.2.1 Popis důležitých tříd

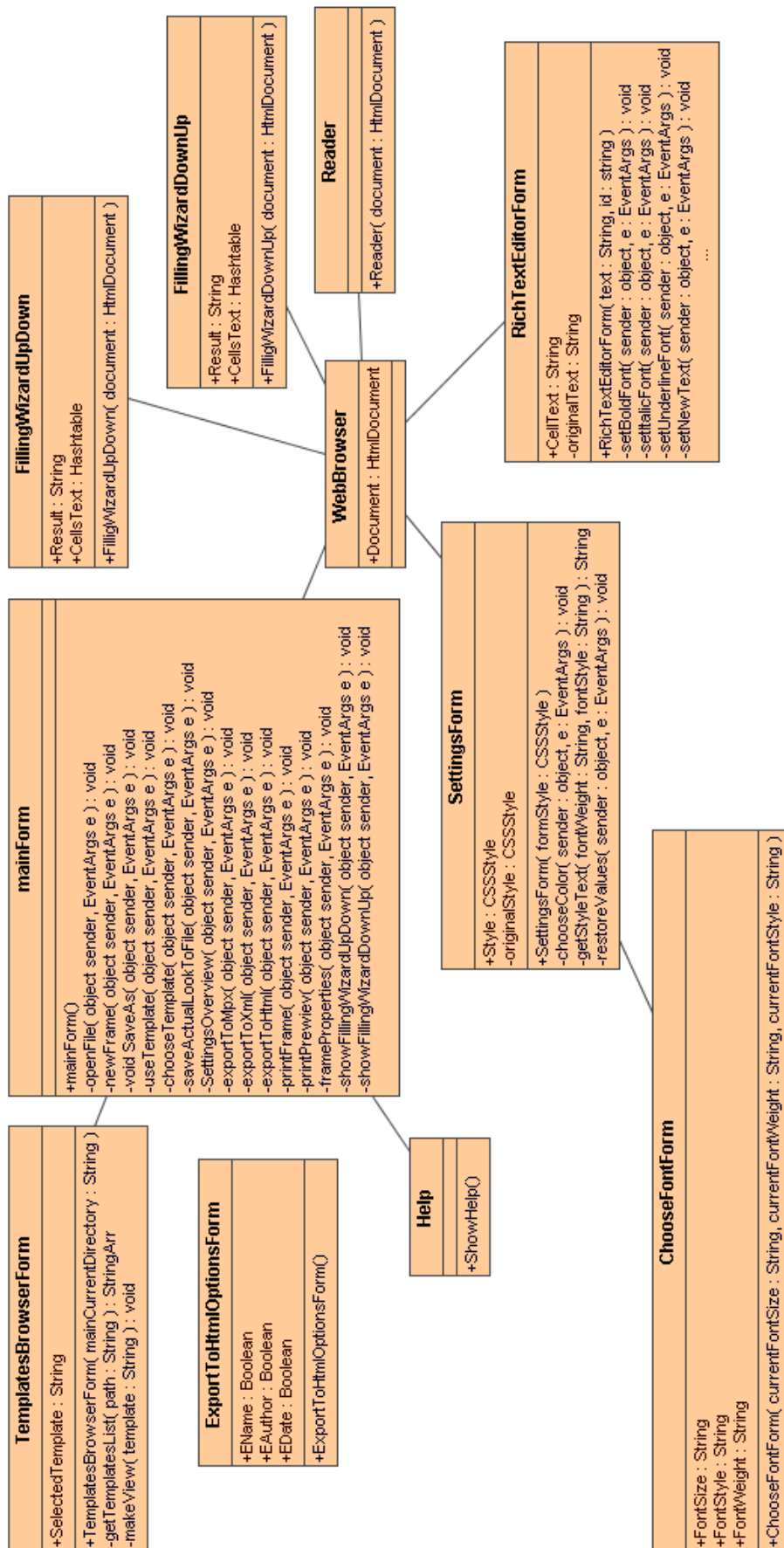
Objekt třídy *MainForm* tvoří hlavní formulář aplikace. Obsahuje hlavní menu, ze kterého lze volat všechny důležité funkce programu. V tomto objektu se nachází objekt třídy *WebBrowser*, který zobrazuje logický rámeček a je v něm uchováván jeho obsah.

Třídy *FillingWizardUpDown* a *FillingWizardDownUp* realizují průvodce zadáváním. Průvodce je realizován sledem kroků, ve kterých je uživateli usnadněno vyplňování rámce pomocí kladení vhodných otázek k příslušné položce. Třída *FillingWizardUpDown* realizuje metodu vyplňování shora dolů, *FillingWizardDownUp* metodu zezdola nahoru.

Třída *RichTextEditorForm* realizuje editor pro přímou editaci obsahu položky logického rámce. Nabízí také možnost základního formátování písma.

TemplatesBrowserForm umožňuje procházení uložených šablon vzhledu logického rámce, včetně zobrazení jejich náhledu. Libovolnou šablonu lze nastavit jako implicitní a použít její vlastnosti na otevřený logický rámeček.

Třída *SettingsForm* realizuje formulář pro nastavení vlastností otevřeného logického rámce. Nastavovat lze vlastnosti písma a barvu pozadí položek.



Obrázek 7.3: Diagram tříd –model uživatelského rozhraní

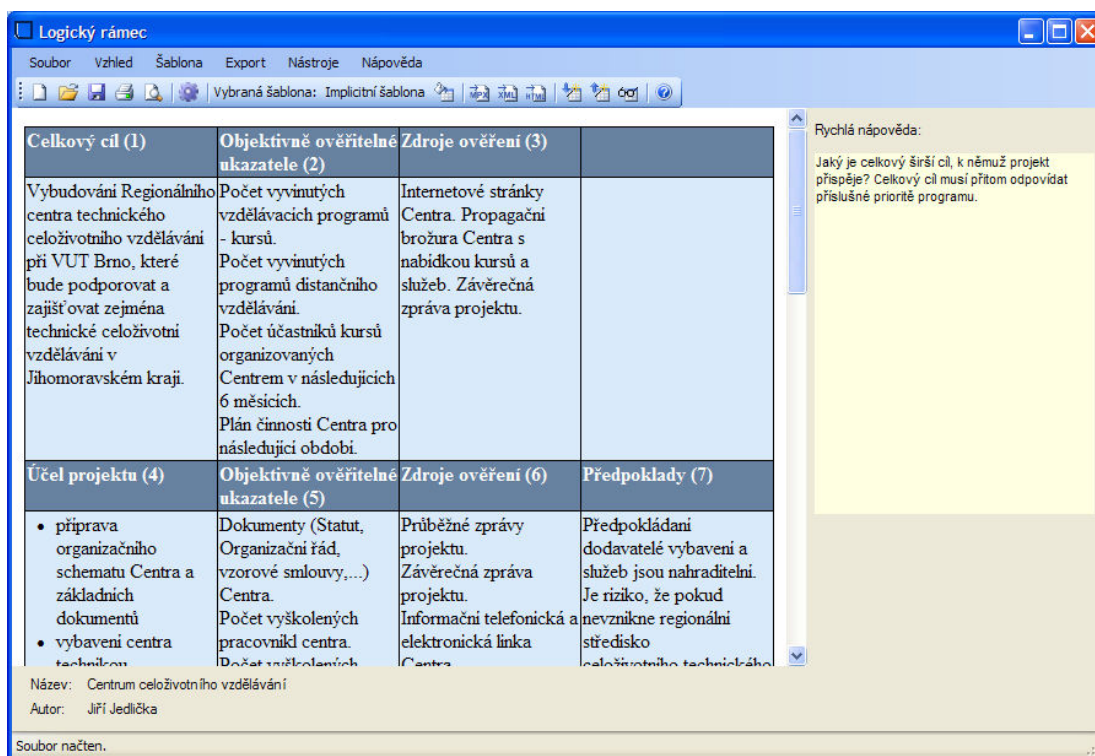
8 Implementace

Před samotnou implementací bylo nutno vymyslet vzhled a vhodné rozmístění prvků tak, aby byla práce pro uživatele co nejvíce intuitivní a pohodlná. Další otázkou bylo vhodné zobrazení logického rámce a způsob jeho uložení.

Při implementaci bylo snahou co nejvíce využít vlastností vývojového prostředí Visual Studio, které nabízí velkou podporu pro práci s XML a HTML formáty. S výhodou se také daly využít již hotové komponenty, které nabízí jak samotné Visual Studio, tak internetové stránky a fóra vývojářů v jazyku C#.

8.1 Uživatelské rozhraní

Uživatelské rozhraní bylo vytvořeno pomocí návrháře Visual Studia. Ten umožňuje rychlé a pohodlné vytvoření všech potřebných ovládacích prvků. Nový prvek lze vytvořit pouhým výběrem ze seznamu dostupných komponent a jeho přetažením do požadovaného formuláře. Stejně snadno lze prvky také rozmístit nebo jejich rozmístění zajistit pomocí umístění do vhodného kontejneru.



Obrázek 8.1: Hlavní okno aplikace

Hlavní okno aplikace obsahuje hlavní menu, ze kterého je možno vyvolat všechny funkce. Menu je členěno do položek, které logicky seskupují funkce podle jejich účelu.

Pod hlavním menu se nachází nástrojová lišta, která umožňuje spouštět často používané funkce jednoduše kliknutím na určitou ikonu. Obrázky ikon byly vytvořeny tak, aby znázorňovaly co nejuvýstižněji danou funkci. Pro standardní funkce, jako je otevření, vytvoření nebo uložení souboru, bylo použito obrázků, známých z většiny aplikací operačního systému Windows. Uživatelé jsou na jejich význam zvyklí a odlišný obrázek by jim ztěžil ovládání aplikace.

Pod nástrojovou lištou je okno rozděleno na dvě části. V levé části se nachází webový prohlížeč, který slouží pro zobrazení logického rámce. Je vytvořen pomocí komponenty *WebBrowser*. V pravé části je pak prvek, který zobrazuje textovou nápovědu pro jednotlivé položky logického rámce. Jeho obsah se mění podle toho, nad kterou položkou logického rámce se nachází kurzor myši. Šířka obou částí je nastavena tak, aby byla zobrazena celá šíře pomocného textu v pravé části. Lze ji však upravit podle uživatelových představ.

Ve spodní části hlavního okna jsou zobrazeny informace o názvu a autorovi rámce a stavový řádek, informující o výsledku proběhlé akce.

8.2 Zobrazení logického rámce

Při rozhodování o vhodném prvku pro zobrazení logického rámce byly podstatné vlastnosti, které logický rámeček má. Bývá zobrazován ve formě tabulky, jednotlivé buňky mohou obsahovat formátovaný text a zobrazit by měla jít také libovolná barva a to jak textu, tak i pozadí buněk. V prostředí Visual Studia se z tohoto hlediska zdály být vhodné dvě komponenty a to *RichTextBox* a *WebBrowser*. Obě umí zobrazit tabulku i formátovaný text. Nakonec byla vybrána komponenta *WebBrowser* a to hlavně kvůli srozumitelnosti zdrojového HTML kódu, který umožňuje snadnější změny obsahu a export do dalších formátů, jako je například XML. Visual Studio navíc nabízí mnoho funkcí pro práci s HTML kódem.

Logický rámeček je tedy zobrazen jako HTML tabulka, která má čtyři sloupce a pět řádků, přičemž každý řádek má své záhlaví. Záhlaví u řádků je řešeno pomocí vložení řádku s označením `class=head`, jehož vzhled je nadefinován pomocí CSS stylu.

Buňky záhlaví jsou označeny identifikátory, začínající písmenem h, po něm následuje číselné označení pořadí buňky tak, jak je zvykem. Ostatní buňky jsou označeny písmenem c, taktéž následovaným číslem pořadí. Identifikátory buněk slouží pro jejich snadné vyhledání a úpravy pomocí DOM modelu. Jeden řádek zobrazeného logického rámce má tedy tuto strukturu:

```
<!-- first row -->
<TR class=head id=firstHead>
  <TD id=h1>Celkový cíl (1)</TD>
  <TD id=h2>Objektivně ověřitelné ukazatele (2)</TD>
  <TD id=h3>Zdroje ověření (3)</TD>
  <TD>&nbsp;</TD>
</TR>
<TR>
  <TD id=c1>Položka 1</TD>
```



```

<TD id=c2>Položka 2</TD>
<TD id=c3>Položka 3</TD>
<TD>&nbsp; </TD>
</TR>

```

Vzhled tabulky zajišťuje CSS styl, který je vygenerován podle nastavení vzhledu logického rámce. Je umístěn na začátku HTML stránky.

Celkový cíl (1)	Objektivně ověřitelné ukazatele (2)	Zdroje ověření (3)	
Položka 1	Položka 2	Položka 3	
Účel projektu (4)	Objektivně ověřitelné ukazatele (5)	Zdroje ověření (6)	Předpoklady (7)
Položka 4	Položka 5	Položka 6	Položka 7
Výsledky (8)	Objektivně ověřitelné ukazatele (9)	Zdroje ověření (10)	Předpoklady (11)
Položka 8	Položka 9	Položka 10	Položka 11
Aktivity (12)	Prostředky (13)	Zdroje (14)	Předpoklady (15)
<ul style="list-style-type: none"> • Aktivita 1 • Aktivita 2 • Aktivita 3 • Aktivita 4 	Položka 13	Položka 14	Položka 15
			Předběžné podmínky (16)
			Položka 16

Obrázek 8.2: Zobrazení logického rámce

8.3 Uložení logického rámce

Logický rámec je ukládán do souboru s příponou lff. Přípona byla zvolena jako zkratka typu souboru Logical Frame File. Jméno a umístění souboru si může uživatel zvolit v dialogu pro uložení souboru. Data jsou uložena ve formátu XML, který je pro ukládání a předávání informací vhodný a je rozšířen v mnoha jiných aplikacích. Do souboru jsou ukládány informace jak o textovém obsahu, tak o vzhledu. Hlavním elementem je *logicalFrame*, který obsahuje elementy *name* s názvem rámce, *author* se jménem autora, *headersStyle* s informacemi o vzhledu záhlaví a *cellsStyle* s informacemi o vzhledu ostatních buněk. Posledním elementem je *content*, který obsahuje HTML podobu tabulky logického rámce.

Soubor má následující strukturu:

```

<?xml version="1.0" encoding="utf-8"?>
<logicalFrame>
  <name>Název logického rámce</name>
  <author>Jméno autora</author>
  <headersStyle>
    <font-size>velikost písma v bodech</font-size>
    <font-style>styl písma - normal nebo italic</font-style>
    <font-weight>tučnost písma - normal nebo bold</font-weight>
    <color>barva písma v hexadecimálním tvaru</color>
    <background-color>barva pozadí v hexadecimálním tvaru</background-color>
  </headersStyle>
  <cellsStyle>
    <font-size>velikost písma v bodech</font-size>

```

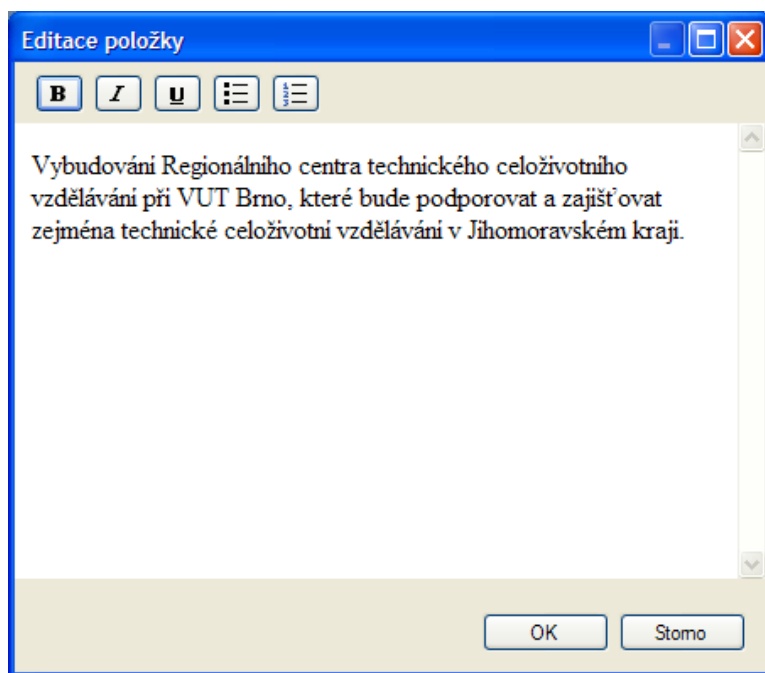
```

<font-style>styl písma - normal nebo italic</font-style>
<font-weight>tučnost písma - normal nebo bold</font-weight>
<color>barva písma v hexadecimálním tvaru</color>
<background-color>barva pozadí v hexadecimálním tvaru
</background-color>
</cellsStyle>
<content>
    Tabulka logického rámce v HTML
</content>
</logicalFrame>

```

8.4 Vyplňování rámce

Zadávání údajů do logického rámce se provádí pomocí dvojitého poklepání na položku, kterou chce uživatel upravovat nebo přes kontextové menu vyvolané stisknutím levého tlačítka myši na příslušné položce. Tato akce vyvolá otevření okna s WYSIWYG HTML editorem, ve kterém může uživatel pohodlně zadat nebo upravit požadovaný text i jeho základní formátování.



Obrázek 8.3: Editace položky

HTML editor je realizován pomocí volně dostupné komponenty *HtmlEditor*, která je k dispozici na adrese <http://www.itwriting.com/htmleditor>. Jedná se o C# wrapper Mshtml COM komponenty firmy Microsoft. Mshtml je částí internetového prohlížeče Internet Explorer, která zajišťuje render HTML. Tento HTML editor umožňuje jak prohlížení, tak editaci HTML kódu. Obsahuje také funkce pro změnu základních vlastností písma. Pro účely této aplikace tedy poskytuje vše, co je potřebné.

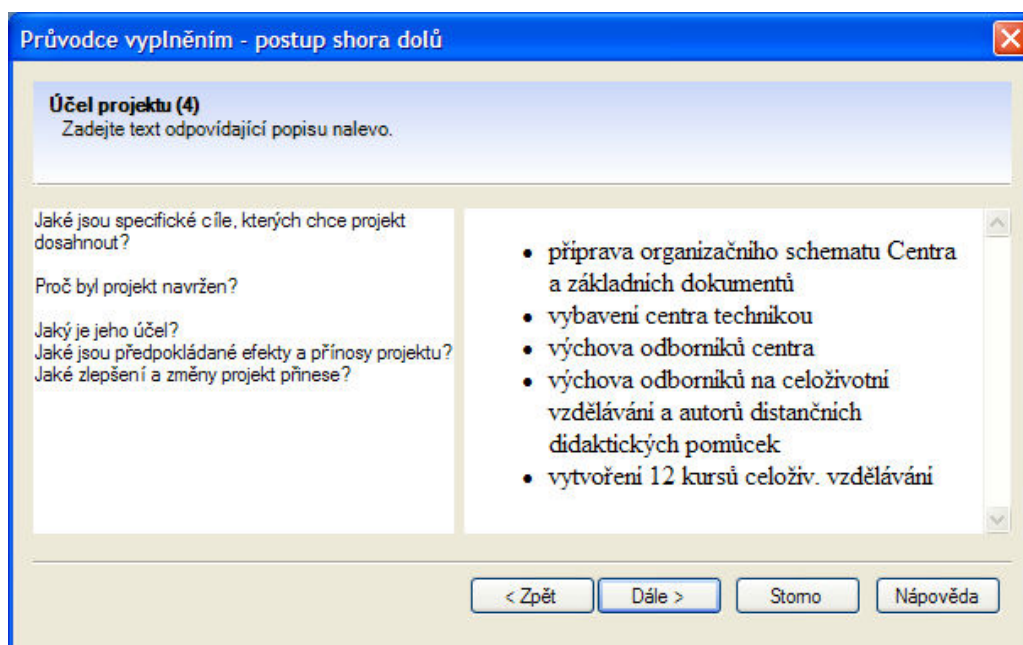
Jediná nevýhoda používání této komponenty je nedodržování kompatibility s XHTML. Při změně kódu dochází například k nahrazování malých písmen názvů elementů za velké, hodnoty parametrů nejsou uzavírány do uvozovek. Toto chování je zapříčiněno používáním Mshtml COM

komponenty, která nezaručuje zachování kódu. To ovšem není pro tuto aplikaci problémem. Internet Explorer, který je pro zobrazení rámce použit, zobrazuje i takto pozměněný HTML kód správně.

8.5 Průvodce vyplněním

Pro usnadnění vyplňování rámce nabízí aplikace možnost spustit průvodce vyplněním. Pro jeho realizaci bylo využito komponenty *Simple Wizard Control for .Net 2.0* od indického vývojáře Manish Ranjan Kumara. Tuto komponentu lze stáhnout na adrese <http://www.codeproject.com/KB/cs/WizardDemo.aspx>.

Autor získal za článek s jejím uveřejněním cenu Best C# article of Apr 2007 serveru codeproject.com [12].



Obrázek 8.4: Průvodce vyplněním

Komponenta umožňuje vytvoření průvodce během krátkého času. Obsahuje vlastního návrháře, pomocí kterého lze nadefinovat všechny potřebné vlastnosti. Nabízí možnost automatického zobrazení navigačních tlačítek pro přechod na další nebo předchozí krok, stejně jako tlačítka Storno a Dokončit, které je zobrazeno v posledním kroku. Tlačítka mají rovněž definované události, na které lze navázat metody.

Do komponenty lze také přidávat další prvky a návrhář umožňuje měnit popisky, barvy a obrázky na pozadí.

Průvodce je definován posloupností několika kroků. V prvním kroku jsou zobrazeny obecné informace a pokyny. V dalších pak dochází k postupnému vyplňování rámce v závislosti na vybrané metodě vyplňování. V posledním kroku je uživatel seznámen s výsledkem a vhodným postupem po ukončení průvodce.

8.6 Úprava vzhledu

Aby bylo možné aplikaci přizpůsobit zvyklostem uživatele, lze měnit vzhled i popisky logického rámce. Popisky se upravují dvojitým poklepáním na vybranou hlavičku položky nebo pomocí kontextového menu, vyvolaného stisknutím levého tlačítka myši na příslušné položce. Zobrazí se okno pro editaci textu, ve kterém mohou být jednoduše provedeny změny.

Vlastnosti písma a barvy pozadí se nastavují pomocí formuláře s nastavením. Informace o vzhledu se ukládají do speciálně vytvořené struktury *CSSStyle*. Z té je podle potřeby generován CSS styl pro HTML kód.

Pro výběr barvy je použita komponenta *ColorDialog*, umožňující vybrat předem připravené základní barvy nebo zvolit barvu libovolnou vlastní. Pro úpravu vlastností písma byl vytvořen dialog, umožňující změnu velikosti, stylu a tučnosti.

Mimo možnosti nastavit vlastnosti textu globálně pro celý logický rámec, lze měnit styl a tučnost také v okně pro editaci obsahu položky jen pro libovolný vybraný text.

Ačkoli si může uživatel vybrat jakoukoli barvu písma i pozadí, je toto nastavení aplikováno vždy pro všechny položky rámce a jejich hlavičky. Důvodem pro toto chování je dodržení jednotného barevného vzhledu příslušných položek.

8.7 Šablony

Do šablony je umožněno ukládat aktuální vzhled logického rámce. Šablona se ukládá do souboru s příponou *lft*. Přípona je zkratkou typu souboru Logical Frame Template. Údaje o šabloně jsou uloženy ve formátu XML. Ukládají se informace o vzhledu položek a popiscích hlaviček. Hlavním elementem je *logicalFrameTemplate*, který dále obsahuje element *name* s názvem šablony, *headersText* s popisky hlaviček, *headresStyle* a *cellsStyle* se vzhledem položek.

Soubor se ukládá do adresáře */templates* v adresáři aplikace. Jeho jméno, stejně jako jméno šablony si může uživatel libovolně zvolit.

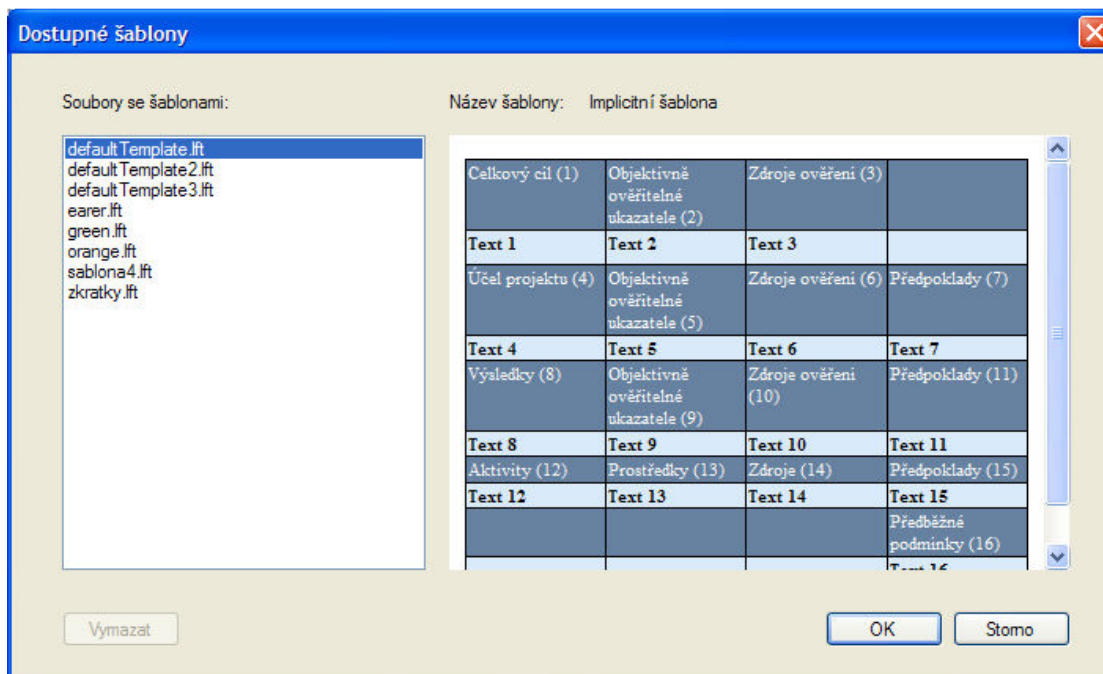
Šablony v uvedeném adresáři pak může uživatel v aplikaci procházet a nechat si zobrazovat jejich náhled. Vybranou šablonu pak lze nastavit jako implicitní a kdykoli aplikovat její vzhled na otevřený logický rámec. Po instalaci aplikace je pro uživatele připraveno několik nadefinovaných šablon.

Soubor šablony má následující strukturu:

```

<?xml version="1.0" encoding="utf-8" ?>
<logicalFrameTemplate>
  <name>Jméno šablony</name>
  <headersText>
    <cell id="1">Popis hlavičky (1)</cell>
    <cell id="2">Popis hlavičky (2)</cell>
    <cell id="3">Popis hlavičky (3)</cell>
    .
    .
    <cell id="16">Popis hlavičky (16)</cell>
  </headersText>
  <headersStyle>
    <font-size>velikost písma v bodech</font-size>
    <font-style>styl písma - normal nebo italic</font-style>
    <font-weight>tučnost písma - normal nebo bold</font-weight>
    <color>barva písma v hexadecimálním tvaru</color>
    <background-color>barva pozadí v hexadecimálním tvaru
  </background-color>
  </headersStyle>
  <cellsStyle>
    <font-size>velikost písma v bodech</font-size>
    <font-style>styl písma - normal nebo italic</font-style>
    <font-weight>tučnost písma - normal nebo bold</font-weight>
    <color>barva písma v hexadecimálním tvaru</color>
    <background-color>barva pozadí v hexadecimálním tvaru
  </background-color>
  </cellsStyle>
</logicalFrameTemplate>

```



Obrázek 8.5: Výběr šablony

8.8 Export

Aplikace nabízí export do tří formátů a to MPX, XML a HTML.

8.8.1 MPX

Tento formát slouží pro výměnu informací mezi různými verzemi programu Microsoft Project nebo ostatními aplikacemi, podporující tento formát. V aplikaci je použita verze MPX 4.0.

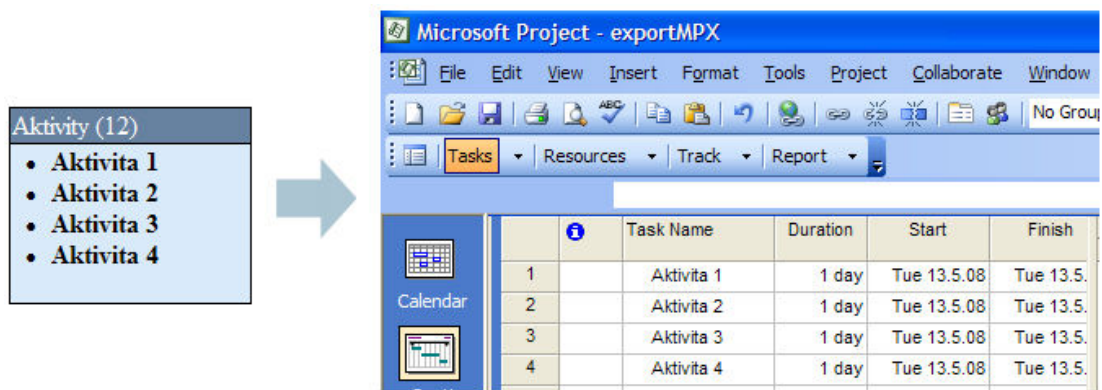
Jedná se o textový soubor a lze pomocí něj přenášet různé informace o projektu jako je přiřazení zdrojů, časový plán nebo aktivity. Údaje jsou zapisovány do záznamů, které jsou oddělené novým řádkem. Každý záznam má své číslo a řádek jím musí začínat. Po čísle záznamu následuje pole příslušných hodnot, které jsou oddělené libovolným oddělovačem. Ten však musí být pro celý soubor jednotný a musí být na začátku definován. Počet a pořadí hodnot je dáno typem konkrétního záznamu. Nutné je také dodržovat pořadí záznamů. To udává tabulka typů záznamů, které je dostupná na internetových stránkách technické podpory firmy Microsoft. Zde lze také nalézt přehled typů záznamů a příslušná pole hodnot [13].

V našem případě se z logického rámce exportují informace o aktivitách. Proto je použit typ záznamu Numerická definice tabulky aktivit, která má číslo 61 a určuje, které informace o aktivitách se budou zaznamenávat. Pole hodnot tohoto záznamu může obsahovat více údajů, například informace o začátku, konci, nákladech nebo procentu dokončení aktivity. Pro náš účel postačí zadávat pouze ID a jméno aktivity, ostatní informace nejsou v logickém rámci obsaženy. Tyto záznamy mají čísla 90 a 1.

Záznam pro export seznamu aktivit pak vypadá například takto:

```
MPX; Microsoft Project; 4.0
61;90;1
70;1;"Aktivita 1"
70;2;"Aktivita 2"
70;3;"Aktivita 3"
70;4;"Aktivita 4"
```

První řádek záznamu definuje verzi souboru MPX a oddělovač hodnot, kterým je zvolen středník. Druhý řádek označuje numerickou definici tabulky aktivit (číslo 61) a hodnoty, které se do ní budou zaznamenávat, konkrétně ID (číslo 90) a jméno aktivity (číslo 1). Další řádky jsou již záznamem samotných aktivit. Číslo 70 udává, že se jedná o aktivitu, následuje její ID a jméno v uvozovkách.



Obrázek 8.6: Vygenerování kostry hierarchické struktury činností v aplikaci Microsoft Project

Po otevření vyexportovaného souboru v aplikaci Microsoft Project se vytvoří kostra hierarchické struktury činností, kterou lze dále upravovat, přidávat další činnosti nebo jim přiřazovat časové úseky.

Verzi MPX 4.0 lze otevřít v libovolné verzi programu Microsoft Project.

8.8.2 XML

Export do formátu XML dovolí zpracovat obsažené informace i ostatními programy. Exportují se pouze textové informace, obsažené v logickém rámci bez informací o jeho vzhledu. Jména elementů jsou v češtině, protože se předpokládá, že aplikaci budou používat převážně čeští uživatelé a význam jednotlivých elementů jim tak bude zřejmý již z názvu.

Hlavním element má název *logickyRamec*. Obsahuje elementy *autor*, *nazev* a *obsahLogickehoRamce*. Element *obsahLogickehoRamce* obsahuje informace o položkách logického rámce, které jsou umístěny v elementu *polozka*, který obsahuje dva elementy a to *nazev* a *obsah*.

Vyexportovaný soubor XML má následující strukturu:

```
<?xml version="1.0" encoding="utf-8"?>
<logickyRamec>
  <nazev>Název rámce</nazev>
  <autor>Autor rámce</autor>
  <obsahLogickehoRamce>
    <polozka>
      <nazev>Popis hlavičky (1)</nazev><obsah>Položka 1</obsah>
    </polozka>
    <polozka>
      <nazev>Popis hlavičky (2)</nazev><obsah>Položka 2</obsah>
    </polozka>
    <polozka>
      <nazev>Popis hlavičky (3)</nazev><obsah>Položka 3</obsah>
    </polozka>
    <polozka>
      <nazev>Popis hlavičky (4)</nazev><obsah>Položka 4</obsah>
    </polozka>
  </obsahLogickehoRamce>
</logickyRamec>
```

```

    <polozka>
      <nazev>Popis hlavičky (16)</nazev><obsah>Položka 16</obsah>
    </polozka>
  </obsahLogickehoRamce>
</logickyRamec>

```

8.8.3 HTML

Export do HTML umožňuje vytvořit soubor s logickým rámcem, který je možno otevřít a prohlížet na všech počítačích vybavených internetovým prohlížečem. Takových počítačů je v dnešní době již drtivá většina a tak je export do tohoto formátu určen hlavně pro prezentaci vytvořeného logického rámce.

Exportuje se jak obsah, tak i vzhled. Volitelně je možno nechat nad tabulku logického rámce vyexportovat i název rámce, jeho autora a datum vytvoření exportovaného souboru.

Struktura vyexportovaného HTML souboru je následující:

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="cs" lang="cs">
<head>
  <meta name="language" content="cs" />
  <meta http-equiv="content-type" content="text/html; charset=utf-8" />
  <title>Logický rámec</title>
</head>
<body>
  <style type="text/css">Vygenerovaný CSS styl</style>
  <h1>Logický rámec</h1>
  <h2>Název</h2>
  <h3>Autor</h3>
  <p>Datum vytvoření</p>
  <p>
    HTML kód tabulky logického rámce
  </p>
</body>
</html>

```

8.9 Nápořěda

Nápořěda je vytvořena programem firmy Microsoft *Html Help Workshop*. Pomocí tohoto nástroje lze vytvořit HTML Help, což je standard pro systémy nápořědy v operačních systémech Windows. Ke spuštění je potřeba internetový prohlížeč Internet Explorer 4.01 a vyšší a Windows 98 a vyšší. Na rozdíl od starého standardu WinHelp se zobrazuje text nápořědy i obsah najednou a uživatel má možnost vyhledávání.

Text nápořědy byl uložen do html souborů. Programem *Html Help Workshop* byly html soubory uspořádaný, byl vytvořen obsah a výsledek je zkomprimován do souboru HTML Help s příponou chm, který je umístěn v adresáři aplikace.

Nápořěda je v aplikaci vyvolána pomocí komponenty *helpProvider*, která má ve vlastnosti *HelpNamespace* zadanou cestu k vytvořenému souboru.

Nápověda poskytuje jak obecné informace o metodě logického rámce, tak i konkrétní návody pro práci s aplikací.

8.10 Ukázka zpracování projektu

Ukázka zpracování projektu metodou logického rámce je uvedena v příloze A. V ní jsou zachyceny obrazovky, postupně znázorňující vytvoření nového logického rámce, jeho vyplnění, změnu vzhledu, kontrolu pomocí metody čtení, uložení, náhled tisku a export do formátů HTML, XML a MPX.

9 Srovnání s existujícími aplikacemi

Při hledání aplikací, umožňující práci s logickým rámcem, jsem objevil pouze dvě. Jednou z nich je program s názvem Logframe for Windows od české firmy Team.cz. Tato společnost se zabývá vzděláváním a rozvojem pracovníků, směřovaný ke zvýšení individuální a týmové výkonnosti v rámci organizací a podniků. Zaměřuje se také na pomoc a podporu při budování systémů pro řízení projektů a zavádění plánovacích postupů s podporou softwaru [14].

Druhým programem, zabývajícím se touto problematikou, je internetová aplikace Dotačního informačního systému Ministerstva pro místní rozvoj (MMR).

9.1 Logframe for Windows

Produkt Logframe pro Windows poskytuje podobné funkce jako aplikace řešená v rámci této diplomové práce. Nabízí možnost vytvoření logického rámce, jeho uložení a načtení. Lze také měnit vzhled rámce, včetně popisu hlaviček položek.

Umožňuje uložit aktuální vzhled rámce. Ten se v aplikaci Logframe nazývá pohled a lze ho kdykoli opět načíst.

9.1.1 Možnosti exportu logického rámce

Logframe také nabízí možnost exportu do formátů MPX a HTML. Navíc umožňuje export do formátu RTF, který může být otevřen textovým editorem Microsoft Word, a dokumentu aplikace Microsoft Excel. To umožňuje jeho další úpravy. Naopak chybí export do dnes již velmi rozšířeného formátu XML.

Kromě exportu do jiných formátů nabízí aplikace Logframe navíc ještě možnost otevřít více logických rámců najednou a spojovat je do programů. V mé aplikaci lze pracovat pouze s jedním otevřeným rámcem.

9.1.2 Kvalita systému nápovědy

Autoři programu Logframe zřejmě předpokládali, že budou s aplikací pracovat uživatelé, kteří metodu logického rámce dobře znají. Nápověda je velmi stručná a zobrazování kontextové nápovědy během zobrazování není možné. Z tohoto pohledu se jeví moje aplikace s obsáhlou nápovědou a zobrazováním pokynů při vyplňování rámce vhodná pro uživatele, kteří metodu neovládají příliš dobře nebo se její používání teprve učí. Kvalitní nápověda může ovšem pomoci i uživateli, který metodu dobře zná.

S tím souvisí i možnost spouštění průvodců vyplňováním v mé aplikaci. Průvodce pomáhá vyplňovat krok po kroku logický rámec na základě dvou metod vyplňování. Takováto podpora v aplikaci Logframe chybí.

9.1.3 Ostatní vlastnosti

Nešťastným řešením aplikace Logframe je podle mě volba definovat všechny položky rámce jako číslovaný seznam. Ten má smysl pouze v případě výčtu aktivit v položce Aktivita. V ostatních případech se jedná spíše o souvislý text a seznam může znepříjemnit vyplňování položky.

Zajímavou funkcí Logframe je možnost měnit pořadí sloupců a řádků. Díky této funkci může uživatel téměř libovolně měnit strukturu rámce. Na druhou stranu má logický rámec strukturu poměrně přesně danou a takováto funkce proto může být v některých případech dokonce nežádoucí.

9.1.4 Rozšířenost aplikace Logframe

Při snaze o nalezení zahraničního produktu pro práci s logickým rámcem jsem zjistil, že český program Logframe je nabízen v anglické verzi také zahraničními společnostmi, zabývajícími se podporou managementu projektů. Jako příklad můžeme uvést americkou společnost Maizemoor International, která pomáhá s velkými projekty po celém světě. Na jejich internetových stránkách lze objednat program Logframe za cenu \$69. Na stránkách českého prodejce je cena jedné licence 1904 Kč s DPH.

9.2 Internetová aplikace Dotačního informačního systému MMR

Tato aplikace se používá v případě podání elektronické žádosti o dotaci od Ministerstva pro místní rozvoj. Aplikace umožňuje vytvořit podle současných znalostí žadatele logický rámec, který se poté stane součástí žádosti. Vytvořený rámec lze také exportovat do formátu dokumentu MS Word nebo ho lze vytisknout [15].

Tuto aplikaci lze použít pouze při zadávání žádosti o dotaci a tak jsem její funkčnost neměl možnost blíže prozkoumat. Z nápovědy k Dotačnímu informačnímu systému však vyplývá, že aplikace umožňuje pouze vyplnění příslušných položek a nenabízí další funkce jako je úprava vzhledu nebo export do dalších formátů kromě formátu dokumentu MS Word. Slouží jen pro účely Ministerstva pro místní rozvoj a tak není rozšířená funkčnost aplikace ani nutná [16].

10 Možná rozšíření

Možných rozšíření, která by zlepšila vlastnosti aplikace, je více.

Uživatel by jistě přivítal export logického rámce i do jiných běžných formátů jako jsou dokument Microsoft Word nebo Excel. Tyto formáty mají složitější zápis zdrojového kódu, než je tomu například u HTML. Existují .NET komponenty pro snadný převod z HTML do formátu RTF, se kterým umí aplikace MS Word pracovat, avšak tato komponenta stojí přibližně \$240. Proto by bylo možné v budoucnu rozšířit kód aplikace o vlastní převod z HTML do RTF, popřípadě dokumentu aplikace MS Excel.

Dalším vhodným rozšířením je přidání funkce, která by umožnila vrácení úprav o několik kroků zpět. Pokud by uživatel omylem přepsal některé údaje v logickém rámci nebo by si provedené úpravy rozmyslel, mohl by se tímto způsobem vrátit do předchozího požadovaného stavu dokumentu.

Výhodou by také byla možnost mít otevřeno více logických rámců najednou, které by bylo možno uložit do jednoho souboru. Tato funkce by se uplatnila při vytváření projektů v rámci velkého programu.

Pro zlepšení exportu do aplikace Microsoft Project by bylo vhodné zadávat u aktivit data jejich předpokládaného zahájení a ukončení. Exportovaný soubor by tak kromě základu hierarchické struktury činností obsahoval i základ časového plánu.

Programů pro vytváření logického rámce je málo a přitom se tato metoda používá po celém světě. Důležitou se stává hlavně v Evropské Unii, kvůli nutnosti vytvoření rámce při žádosti o dotace z jejich fondů. Vhodné by tak bylo i vytvoření dalších jazykových verzí, minimálně anglické, včetně překladu nápovědy.

Protože rozšíření se nabízí hned několik a jejich případná implementace bude postupná, bylo by vhodné vytvořit internetové stránky, na kterých by bylo možno stáhnout aktuální verzi programu spolu s popisem nových vlastností, případně úprav a oprav. Na stránkách by bylo také možno stáhnout další šablony, definující vzhled rámce. Nechyběla by ani aktualizovaná dokumentace, případně diskusní fórum pro řešení různých problémů.

Závěr

V první části byla detailně popsána metoda logického rámce a její vztah k řízení projektu. Vhodné části tohoto textu se pak objevují v nápovědě aplikace. Uživatel se tak může dozvědět potřebné informace o logickém rámci, což mu může pomoci lépe zvládnout proces jeho vytváření.

Aplikace se podařila navrhnout a realizovat tak, aby umožňovala uživateli snadné vytvoření logického rámce a jeho export do několika formátů. Implementovány byly také prvky, pomáhající pochopení procesu tvorby a čtení rámce. Aplikace tak může sloužit jak zkušeným uživatelům, kteří metodu logického rámce znají, tak také k účelům výukovým. Z tohoto důvodu byly implementovány průvodci vyplněním a kontrola vyplnění pomocí metody čtení logického rámce.

Aplikace je kompletně dokončena a lze jí používat. Vzhledem k potřebě sestavování logického rámce v případě žádosti o granty z fondů Evropské Unie je velký předpoklad, že aplikace najde praktické uplatnění u širokého spektra uživatelů. Tomu nahrává i skutečnost, že je v současné době dostupný pouze jeden program, specializující se na vytvoření logického rámce. Z tohoto důvodu bude aplikace patrně rozšiřována o nové funkce, zmíněné v kapitole 10 a její vývoj tak bude nadále pokračovat.

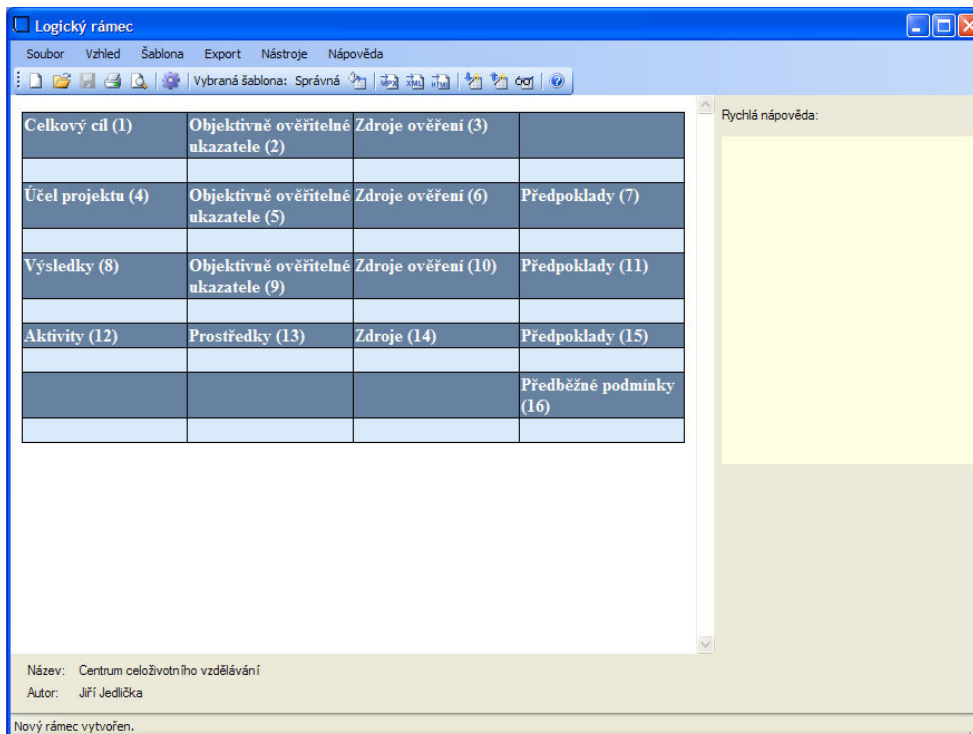
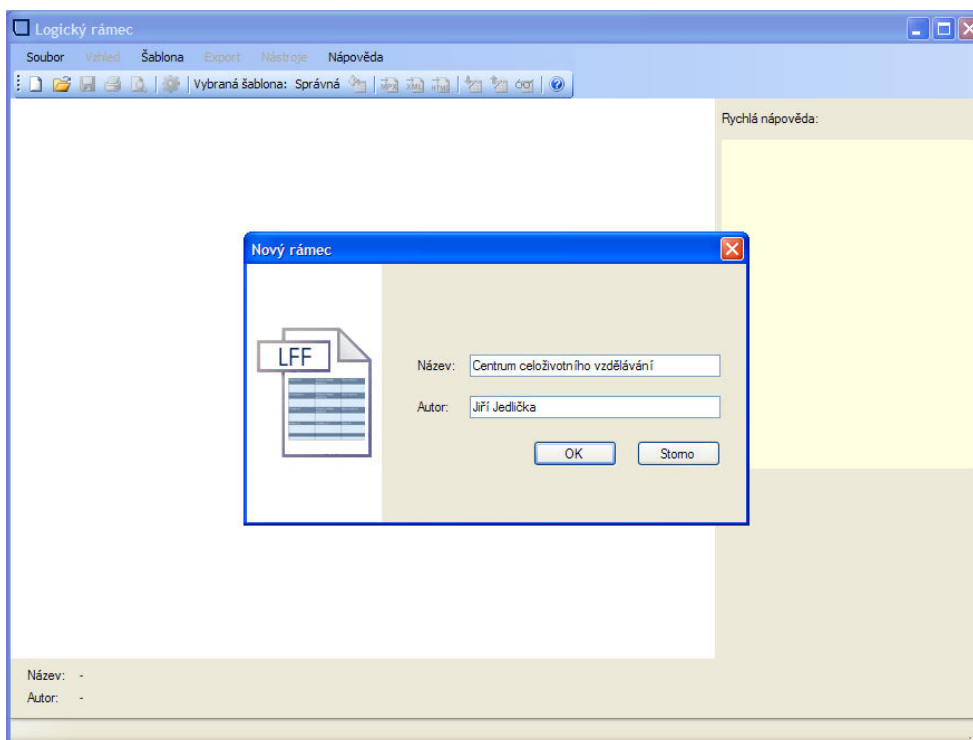
Literatura

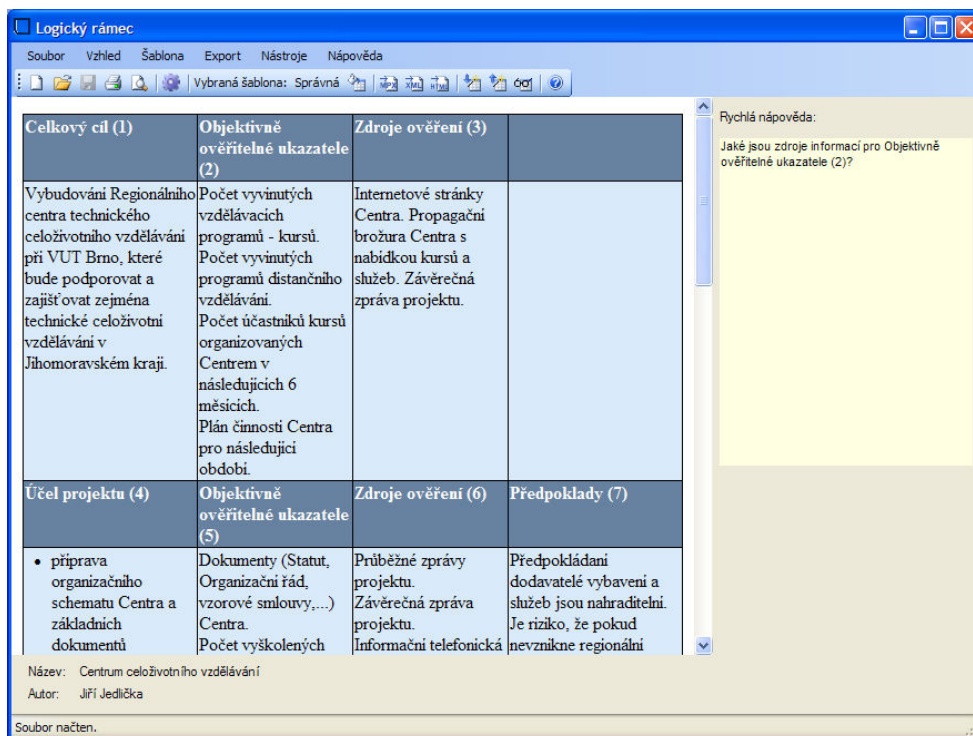
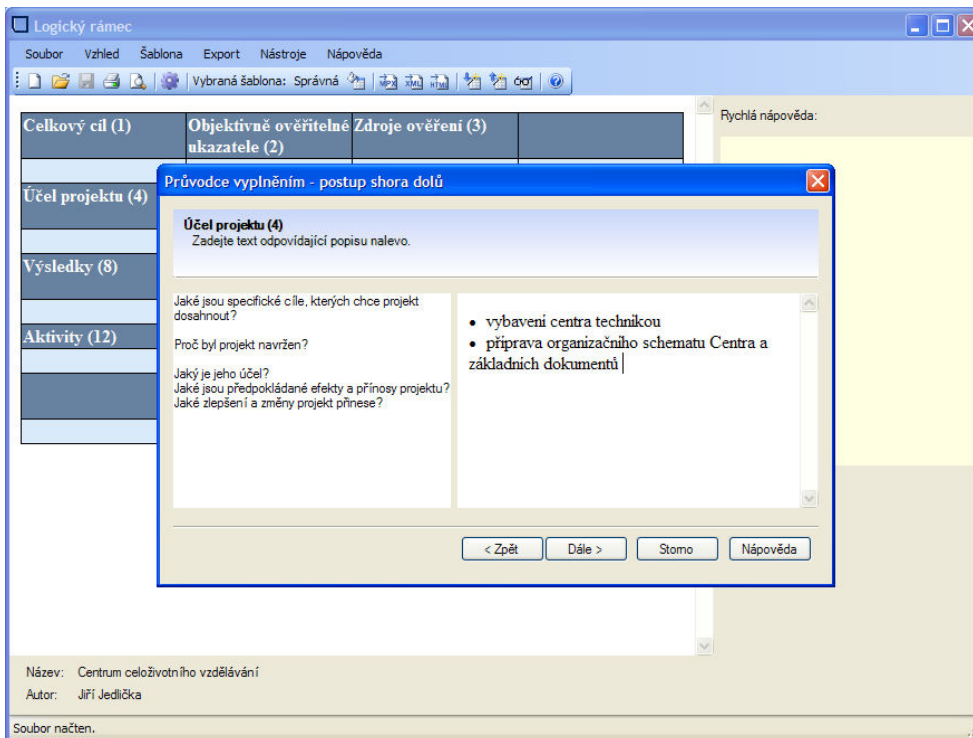
- [1] ROUSENAU, Milton D. *Řízení projektů*. 2. vyd. Brno: Computer Press, 2003. 344 s. ISBN 80-7226-218-1.
- [2] PHILLIPS, Joseph. *IT Project Management: On Track from Start to Finish*. 2nd edition. Emeryville, California, USA : McGraw-Hill/Osborne, 2004. 535 s. ISBN 0-07-223202-1.
- [3] KRESLÍKOVÁ, Jitka. *Řízení projektů informačních systémů: Studijní opora*. Brno : FIT VUT, 2006. 124 s.
- [4] KŘENA, Bohuslav, KOČÍ, Radek. *Úvod do softwarového inženýrství: Studijní opora*. Brno : FIT VUT, 2005. 87 s.
- [5] ZENDULKA, Jaroslav, BARTÍK Vladimír, KVĚTOŇOVÁ Šárka. *Analýza a návrh informačních systémů: Studijní opora*. Brno : FIT VUT, 2006. 178 s.
- [6] Capability Maturity Model - Wikipedia, the free encyclopedia [online]. 2001 [cit. 2007-12-19]. Dostupný z WWW: <http://en.wikipedia.org/wiki/Capability_Maturity_Model>.
- [7] Capability Maturity Model Integration - Wikipedia, the free encyclopedia [online]. 2001 [cit. 2007-12-19]. Dostupný z WWW: <<http://en.wikipedia.org/wiki/CMMI>>.
- [8] Ministerstvo pro místní rozvoj. *Logický rámeček : Metodická příručka*. 2004. 19 s. Dostupný z WWW: <<http://www.strukturalni-fondy.cz/upload/1083944668logickyramecesropv1.2.doc>>.
- [9] FLORIAN, Jan. *Základní dokumentace projektu v rámci dotačního titulu 7*. 2002. 25 s. Dostupný z WWW: <<http://www.isu.cz/download/LOGRAM.rtf>>.
- [10] KRESLÍKOVÁ, Jitka. *Listina základního vymezení projektu*. 2004. 11 s. Podpůrný materiál předmětu IRP na FIT VUT v Brně.
- [11] KALIŠ, Jan, HYNDRÁK, Karel, TESAŘ, Vlastimil. *Microsoft Project: Kompletní průvodce*. 1. vyd. Brno : Computer Press, 2003. 597 s. ISBN 80-251-0074-X.
- [12] ANDERSON, Tim. The HtmlEditor - a C# control that wraps MSHTML. *Tim Anderson's ITWriting* [online]. 2004 [cit. 2008-05-07]. Dostupný z WWW: <<http://www.itwriting.com/htmleditor/>>.
- [13] Microsoft. *Microsoft Pomoc a podpora: PRJ: Description of the MPX Project File Exchange Format* [online]. 2007 [cit. 2008-05-07]. Dostupný z WWW: <<http://support.microsoft.com/kb/270139>>.
- [14] Team.cz. *O firmě* [online]. 2006 [cit. 2008-05-07]. Dostupný z WWW: <<http://www.team.cz/cz/about.htm>>.

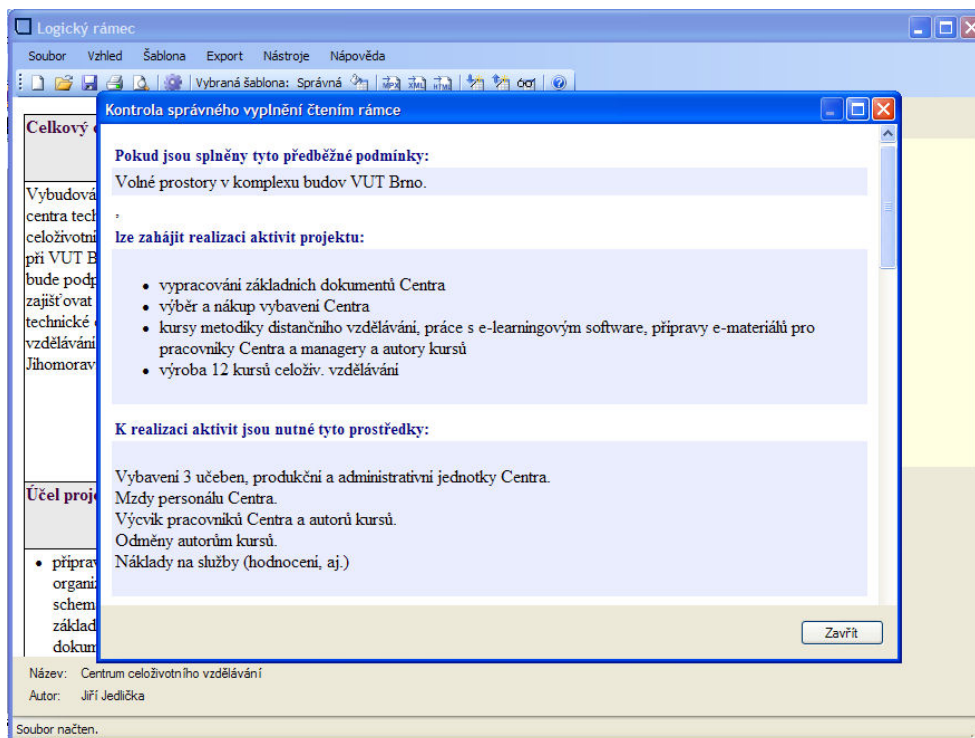
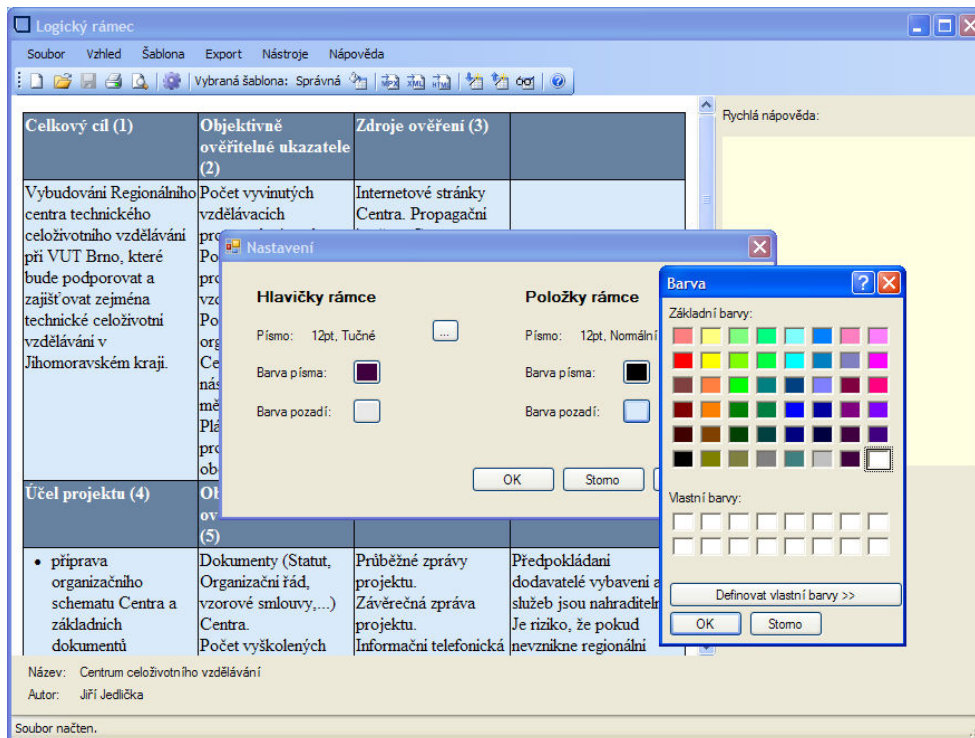
- [15] Ministerstvo pro místní rozvoj. *Výzva k podávání žádostí o dotace na přípravu projektů obcí a venkovských mikroregionů pro Společný regionální operační program*. 2004. 6 s. Dostupný z WWW: <www.mmr.cz/upload/1090490600vyzvasrop290604.doc>.
- [16] Ministerstvo pro místní rozvoj. *Nápověda k IS DIS MMR, Program obnovy venkova* [online]. 2006 [cit. 2008-05-07]. Dostupný z WWW: <http://www3.mmr.cz/pov/help/Start_fr.htm?ref=1210147811576>.

Přílohy

A Zpracování projektu metodou logického rámce







Logický rámec

Soubor Vzhled Šablona Export Nástroje nápověda

Vybraná šablona: Správná

Uložit jako

Uložit do: zadost

Poslední dokumenty
Plocha
Dokumenty
Tento počítač
Místa v síti

Název souboru: logicky_ramec Uložit
Uložit jako typ: Logical Frame Files Storno

Celkový cíl (1)	Objektive	Učel projektu (4)	Objektive	Průběžná zpráva projektu.	Závěrečná zpráva projektu.	Informační telefonická	dotavatelé vybavení a služeb jsou nahraditelní. Je nízké, že pokud nevnikne regionální
Vybudování Regionálního centra technického celoživotního vzdělávání při VUT Brno, které bude podporovat a zajišťovat zejména technické celoživotní vzdělávání v Jihomoravském kraji.	Počet vyvinutých vzdělávacích programů - kurzů Počet vytvářených programů a vzdělávacích programů Počet účastníků kurzů organizovaných Centrem v následujících 6 m. letech Plán činnosti Centra pro následující období	<ul style="list-style-type: none"> příprava organizačního schématu Centra a základních dokumentů 	<ul style="list-style-type: none"> Organizační řád, vzorové smlouvy,... Centra. Počet vyškolených 				

Název: Centrum celoživotního vzdělávání
Autor: Jiří Jedlička
Soubor načten.

Náhled

Zobrazení jedné stránky Prizpůsobit buňce

Logický rámec Stránka 1 z 2

Celkový cíl (1)	Objektive a měřitelné ukazatele (2)	Zdroje ověření (3)	Zdroje ověření (6)	Předpoklady (7)
Vybudování Regionálního centra technického celoživotního vzdělávání při VUT Brno, které bude podporovat a zajišťovat zejména technické celoživotní vzdělávání v Jihomoravském kraji.	Počet vyvinutých vzdělávacích programů - kurzů Počet vytvářených programů a vzdělávacích programů Počet účastníků kurzů organizovaných Centrem v následujících 6 m. letech Plán činnosti Centra pro následující období	Internetové stránky Centra. Propagační brožura Centra s nabídkou kurzů a služeb. Závěrečná zpráva projektu.	Průběžná zpráva projektu. Závěrečná zpráva projektu. Informační telefonická a elektronická linka Centra. Internetové stránky Centra.	Předpokládání dodavatelé vybavení a služeb jsou nahraditelní. Je nízké, že pokud nevnikne regionální středisko celoživotního technického vzdělávání při VUT Brno, které univerzita není schopna vytvořit ze státních dotací, tyto služby nahradí roztržitěra a neuplná, draží nabídka soukromého sektoru.
Účel projektu (4)	Objektive a měřitelné ukazatele (5)	Zdroje ověření (10)	Předpoklady (11)	
<ul style="list-style-type: none"> příprava organizačního schématu Centra a základních dokumentů vybavení centra technickou výchova odborníků centra výchova odborníků na celoživotní vzdělávání a autorů distančních didaktických pomůlek vytvoření 12 kurzů celoživ. vzdělávání 	<ul style="list-style-type: none"> Dokumenty (Situa, Organizační řád, vzorové smlouvy, ...) Centra. Počet vyškolených pracovníků centra. Počet vyškolených pracovníků VUT Brno. Dokumentace vytvořených kurzů. Vybavení centra 	Průběžná zpráva projektu. Závěrečná zpráva projektu. Závěrečná oponentura.	Financování projektu musí probíhat od začátku a plynu.	
Výsledky (8)	Objektive a měřitelné ukazatele (9)	Zdroje ověření (14)	Předpoklady (15)	
Centrum schopné produkovat a realizovat kurzy celoživotního vzdělávání v souladu s potřebami vzdělávacího trhu regionu. 12 kurzů vytvořených na základě zájmu potenciálních uživatelů. V regionu není istuje odborná tolik potřebná vzdělávací instituce.	Srovnání plánovaných a vytvořených vzdělávacích programů. Srovnání plánovaných a vytvořených programů a vzdělávacích programů u distančního vzdělávání. Srovnání plánovaného počtu účastníků vzdělávacích kurzů B s počtem vyškolených pracovníků.	Průběžná zpráva projektu. Kalendář postupu prací. Zápis z poradní řídicího týmu.	Pouze plynslé financování projektu od počátku, výtva není Centra mus by provedeno v prvích třech měsících práce na projektu.	
Aktiviny (12)	Prostředky (13)			
<ul style="list-style-type: none"> vpracování základních dokumentů Centra výběr a nákup vybavení Centra kurzy metody distančního 	Vybavení 3 učeben, produkční a administrativní jednotky Centra. Náklady personálu Centra. Náklady pracovníků Centra a autorů			

13.5.2008

Stránka 1 z 2

Microsoft Project - exportMPX-1

File Edit View Insert Format Tools Project Collaborate Window Help

Type a question for help

Tasks Resources Track Report

Task Name	Duration	Start	Finish	Pre	12 May '08							19 May '08							26 May '08						
					F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T
1 vypracování základních dokumentů Centra	1 day	Tue 13.5.08	Tue 13.5.08																						
2 výběr a nákup vybavení Centra	1 day	Tue 13.5.08	Tue 13.5.08																						
3 kursy metodiky distančního vzdělávání, práce s e-learnin	1 day	Tue 13.5.08	Tue 13.5.08																						
4 výroba 12 kursů celořiv. vzdělávání	1 day	Tue 13.5.08	Tue 13.5.08																						

Ready

EXT CAPS NUM SCRL OVR

B Přiložené CD

CD se zdrojovými texty aplikace i technické zprávy, programovou dokumentací a ukázkovými daty je přiloženo v deskách diplomové práce.