

SW nástroje pro podporu projektového řízení v malých a středních firmách.

Ing. Martin Zedek

NetCentrum, s.r.o, Drtinova 10, Praha 5, ČR, martin.zedek@netcentrum.cz

Abstrakt

Efektivní řízení a zvládnání firemních procesů, v tomto konkrétním případě procesu projektového řízení, je pro dosažení vysoké konkurenceschopnosti stejně důležité u malých i velkých firem. Rozdíl mezi malými a velkými firmami je v množství prostředků, které si mohou dovolit tyto firmy do podpůrných SW systému investovat.

Ve svém příspěvku se pokouším specifikovat z mého pohledu hlavní požadavky na systém podpory projektového řízení, konkrétně v prostředí firmy zabývající se tvorbou softwarových produktů. V článku je stručně rozebrán proces zpracování projektu, kdy v každé fázi jsou navrženy možnosti podpory pomocí SW nástrojů, to vše s přihlédnutím k finančním možnostem malých a středních firem. Při specifikaci požadavků vycházím z potřeb a praktických zkušeností z pozice projektového manažera v takové firmě.

1 PROJEKTY VŠUDE KOLEM NÁS

Abstrahujeme-li od technických detailů a technického pohledu na věc, který se nám nejspíše automaticky asociuje před očima, zmíníme-li slovo *projekt*, dá se říct, že vše co ve svém životě děláme je projekt a všechny tyto projekty mají v principu stejné zákonitosti. Ať už je tím projektem stavba našeho domu, zorganizování dovolené nebo například vývoj úspěšného softwarového produktu.

Všechny projekty mají hlavní atributy společné: musí mít dobře a přesně stanoven svůj cíl, bez toho se nikdy nedostaneme k velmi uspokojivému stavu dokončeného projektu, musíme dobře pracovat se zdroji, které pro realizaci máme k dispozici a musíme brát ohled na technická a časová omezení.

Metodiky pro vedení projektů jako například RUP, metody softwarového inženýrství, aj. nabízejí velmi propracované návody jak projekty řešit. Rozkládají projekty na fáze a iterace, definují milníky a checklisty. Pro všechny kroky existují softwarové nástroje, které tyto kroky mají za úkol maximálně zefektivnit, zrychlit a zajistit jejich úspěšnou realizaci.

Za těchto okolností je docela snadné podlehnout iluzi toho, že tyto softwarové nástroje za nás projekt vyřeší. Je dobré myslet na to, že „školácká“ aplikace těchto metod a softwaru k úspěšnému řešení s velkou pravděpodobností nepovede. Software ani metodiky ještě dlouho nenahradí schopného manažera – poslední kámen do mozaiky úspěšného projektu, který jsem zatím nezmínil.

1.1 Opravdová úloha projektového manažera

Předchozími odstavci jsem se pokusil obhájit, že název „nástroje pro podporu projektového řízení“ je oprávněný. Softwarové nástroje pomáhají zjednodušit operativu spojenou s projektem, jsou nástrojem pomáhajícím všem členům tvůrčího týmu i vedoucímu manažerovi, nejsou ale

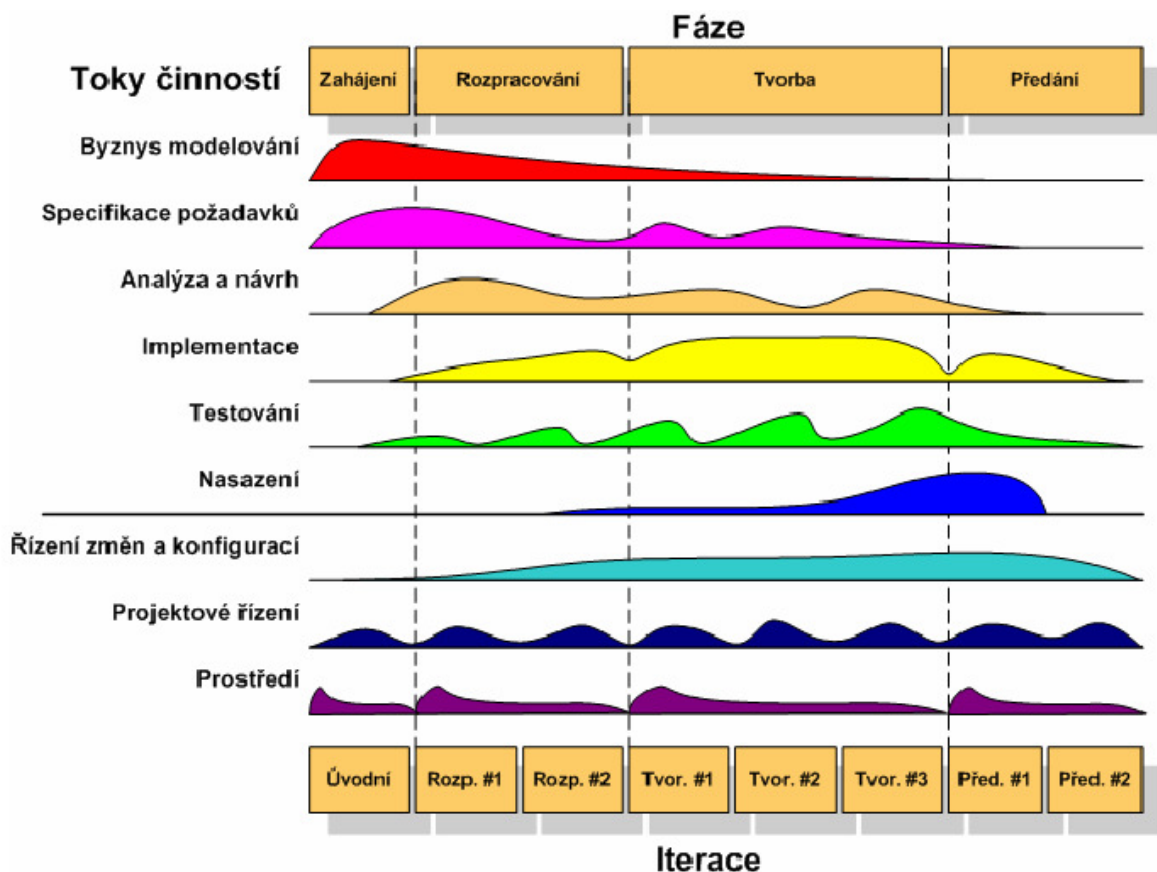
něčím co manažera dokáže nahradit. Bylo by mylné se domnívat, že hlavní úlohou manažera je kontrola termínů, kontrola plnění úkolů členy týmu, toto je opravdu možné při troše snahy automatizovat a nahradit vhodným softwarem, hlavní přidanou hodnotou projektového manažera je jeho schopnost vhodně všechny zdroje vzájemně sladit a umět přinášet správná a strategická rozhodnutí v kritických fázích projektu.

1.2 Papírová nástěnka vs. Softwarové nástroje

Nástěnka s nalepovacími štítky a oběh dokumentů v papírové podobě mohou plnit roli pomocníka projektového vedení stejně dobře jako drahá softwarová řešení. Kouzlo je v osobě manažera, který vše umí zorganizovat a dovést vše úspěšně k cíli. Výhoda softwarových nástrojů se pak projevuje v možnostech snazší editace dokumentů, jejich sdílení, automatické kontrole termínů, zásadní výhoda nastává v okamžiku kdy tvůrčí tým je rozmístěn do více vzdálených lokalit. Tady je efektivní komunikace a sdílení dokumentů nezbytná.

2 ŽIVOTNÍ CYKLUS (SOFTWAREVÉHO) PROJEKTU

Životní cyklus projektu lze dobře demonstrovat na obrázku 1 vycházejícímu z metodiky RUP.



Obrázek 1. Schéma projektu dle metodiky RUP

Projekt je zde rozdělen do několika hlavních fází. Jednotlivé fáze si stručně popíšeme a pokusíme definovat typ softwarového nástroje, které v dané fázi může práci ulehčit. V další kapitole pak z jednotlivých typů složíme komplexní softwarové řešení s možností alternativ.

2.1 Sdílení informací

Na softwarovém projektu se podílí mnoho subjektů – je to minimálně osoba zadavatele, projektového manažera, analytika, designera, programátora, testera a mnohých dalších. Aby mohl být projekt vůbec realizován, je nezbytné aby se všichni členové týmu měli možnost efektivně dorozumívat a vyměňovat si informace. Je to o to důležitější pokud není tým situován v jedné lokalitě – tato situace je v současnosti stále běžnější.

Základem je zpracovaný systém pro *sdílení a správu dokumentů* podporující uživatelská přístupová práva. Důležitou vlastností takového systému je také jeho schopnost zpřístupňovat data jak v interní síti/intranetu, tak všem členům týmu pracujícím ze vzdálených lokalit. Právě možnost přístupu přes webové rozhraní by měla být jednou ze základních vlastností takového systému

Druhým stavebním kamenem je systém *elektronické komunikace*, v praxi to znamená nejčastěji mailový server s definovanými uživatelskými a pracovními skupinami, kalendářovými a adresářovými službami. Platformová nezávislost klientských aplikací a možnost přístupu přes webové rozhraní i v tomto případě velmi zjednoduší práci celého týmu. Pro opravdu rychlou komunikaci je nejlepším nástrojem systém instant messagingu. V případě dislokovaných pracovišť mohou dále pomoci nástroje pro pořádání videokonferencí.

Důležitým nástrojem je také *systém správy úkolů* umožňující zadat a hlídat úkoly každého člena týmu v daném čase projektu.

2.2 Systém podpory projektového plánování

Při realizaci projektů pracujeme se třemi proměnnými – časem, zdroji a kvalitou. Tyto proměnné vystupují ve všech fázích projektu, jsou ve vzájemném vztahu – úspora času znamená potřebu více zdrojů, případně snížení kvality provedení (nemusí být nutně kvalita ale například rozsah zpracování). Je úkolem projektového manažera, aby činil taková rozhodnutí, která tyto tři proměnné vhodně sladí, s cílem maximalizovat kvalitu projektu, minimalizovat náklady a potřebný čas.

Jak bylo řečeno, projekt je rozdělen na jednotlivé etapy, každá etapa má své milníky (kontrolní body), každá etapa je dekomponována na jednotlivé činnosti. Právě zde plní svou roli asi nejdůležitější nástroj – *software pro podporu projektového plánování*. Ten umí vést informace o jednotlivých činnostech, ukazovat závislosti činnosti, seskupovat činnosti patřící k sobě. Dokáže nalézt takzvanou kritickou cestu projektu (nejdelší cesta v síťovém grafu činností ukazující nejbližší možné dokončení projektu) a zobrazit Gantův diagram s informacemi o činnostech a zdrojích potřebných pro realizaci těchto činností. V ideálním případě by měl být tento nástroj napojen na systém správy úkolů aby bylo možné přímo úkolovat konkrétní členy týmu a zároveň získávat zpětnou vazbu a ihned sledovat stav plnění úkolů na stav projektu.

2.3 Fáze projektu

Na začátku zadavatel definuje své představy, definuje klíčové vlastnosti a procesy systému. Kromě standardních nástrojů pro správu dokumentů v této fázi pomohou hlavně *vizuální nástroje* umožňující jednotlivé klíčové procesy a vazby mezi nimi znázornit.

Následuje fáze podrobného zpracování zadání, stanovení hlavních úkolů a etap, zpracování hrubého časového plánu, zpracování rozpočtu, definování tzv. use case modelu. V této chvíli již nástroj pro projektové plánování může pomoci právě při stanovování časového plánu nebo odhadu nákladů

Ve fázi analýzy a návrhu mohou pomoci databázové návrháře apod. Z pohledu nástrojů pro podporu projektového řízení není tento bod příliš relevantní.

Následuje implementace – hlavním úkolem je v této fázi bdít nad realizací potřebných úkolů ve stanoveném čase a pořadí. To opět usnadní nástroj pro projektové plánování.

Důležitá je fáze finálního testování projektu. Z testování přichází množství zpětných vazeb, tuto etapu mají na starosti tzv. *bug tracking* systémy. Opět ideální by bylo napojení bug tracking systému na systém projektového plánování aby bylo možné sledovat vliv finálního testování na plnění projektu ve stanoveném čase.

Ve všech fázích projektu vzniká množství projektové dokumentace, která by právě prostřednictvím sdílení a správy dokumentů měla být přístupná všem členům týmu.

3 HLAVNÍ SOFTWAREVÉ KOMPONENTY

V minulé kapitole jsme si vyjmenovali typy aplikací které mohou práci v procesu realizace usnadnit. Cílem této kapitoly je jednotlivé klíčové aplikace podrobněji popsat a uvést konkrétní příklady jejich řešení.

- *sdílení a správa dokumentů* – je základem pro zpřístupnění všech projektových dokumentů. Základním požadavkem je možnost nastavení přístupových práv k jednotlivým dokumentům. Nastavení přístupových práv a sdílení je dobře zvládnuto například v Novell Netwaru nebo v systému Microsoft Server, podobného efektu lze ale docílit i pomocí freewarových prostředků Linux Serveru. Ve všech případech se budeme potýkat s problémem zpřístupnění souborů přes webové rozhraní. Neméně důležitým požadavkem pro skupinovou práci je systém zamykání dokumentů a hlavně systém sledování změn umožňující držet pohromadě dokumenty na kterých pracuje současně více lidí. Osvědčeným systémem je například CVS nebo SVS. V obou případech budeme opět řešit jak napojit práva k souborům se systémem verzí. Pro samotnou editaci dokumentů lze použít jak známé komerční kancelářské balíky tak například freewarový Open Office, který je schopen editovat i soubory komerčních balíků.
- *elektronická komunikace* – je relativně nejlépe vyřešenou částí. Standardních mailových řešení je několik. Asi nejznámější je Microsoft Outlook a Exchange Server. Ve světě freewarových řešení existuje mnoho kvalitních emailových klientů (Thunderbird, Mutt, The Bat), kvalitních groupwarových nástrojů které by zpřístupňovaly kontakty a kalendář ale ve freewarové podobě mnoho nenajdeme. Vlastovkou je freewarový, čistě webový groupware systém Imp.

V poslední době velmi oblíbeným systémem je tzv. Instant Messaging – systém rychlé výměny zpráv, nejznámějšími řešeními jsou ICQ a Jabber Server. Použitelnými klienty jsou například ICQ, Miranda, PSI. Všechny tyto nástroje jsou dostupné zdarma.

Pro zmíněné videokonference lze s úspěchem použít Microsoft Meeting, nebo pro internetovou telefonii například oblíbený freewareový Skype.

- *systém správy úkolů* – opravdu kvalitních taskových systémů je velmi málo, například „úkolovník“ v Microsoft Outlook mezi ně nelze počítat. Z freewareových nástrojů je za jistých okolností použitelná například freewareová BugZilla. Základním požadavkem na takový systém je možnost určování priorit, možnost přiřazovat úkoly k jednotlivým částem projektu, sledování stavů úkolů, možnost přidávání komentářů, možnost exportu aktuálních denních úkolů dané zodpovědné osobě v jednoduché formě a napojení na komunikační systémy (mail, instant messaging a to obousměrně)
- *systém pro podporu projektového plánování* – asi nejznámější aplikací tohoto druhu je Microsoft Project. Z freewareového světa potom Open WorkBench. Obě tyto aplikace ale z mého pohledu trpí uzavřeností, jsou relativně složité a napojení na zmíněné systémy (hlavně systémy třetích stran je relativně obtížné).
- *vizuální nástroje* – usnadňují přípravu schémat, data flow diagramů, procesů, datových modelů – mezi nejznámější patří Microsoft Visio. Podobných nástrojů, které umí vizuální podobu převést například do databázové struktury nebo dokonce navrhnout zdrojový kód existuje velké množství.
- *bug tracking systémy* – jsou stěžejní ve fázi finalizace a testování projektu. Díky nim je možné mnohem jednodušeji sledovat a zajistit opravu všech známých nedostatků. Nejznámějším je například freewareová BugZilla.

4 ZÁVĚR

Jak tento článek naznačil, softwarových nástrojů, které mohou zjednodušit plánování a správu projektu je velké množství. Jako jejich největší slabinu v současnosti ale vnímám jejich vzájemnou nekompatibilitu. V podstatě neexistuje komplexní balík, který by rozumně a za rozumnou cenu zastřešoval všechny zmíněné oblasti. Velkým problémem je pak také závislost konkrétních řešení na platformě operačního systému. To výrazně stěžuje situaci v nehomogenním vývojovém prostředí. Z mého pohledu tudíž existuje stále velký prostor a důvod takový komplexně pojatý balík vyvinout.

5 LITERATURA

[1] Vondrák, I. Úvod do softwarového inženýrství. Ostrava: 2002, s. 12

[2] West, D. Planning Project with RUP, Rational software development company, 2002, 14 s.