

Ako riadiť členov tímu v ich domovoch?

DUŠAN ZELENÍK

*Slovenská technická univerzita
Fakulta informatiky a informačných technológií
Ilkovičova 3, 842 16 Bratislava
dusan[.]zelenik[zavináč]gmail[.]com*

Abstrakt. Táto esej, v ohraničenej oblasti, poskytuje spracovanie myšlienok, nadobudnutých znalostí a skúseností autora s podpornými prostriedkami pre manažovanie ľudských zdrojov, zodpovednosti a sledovania úloh v softvérovom projekte. Sústredím sa hlavne na problematiku spojenú s projektom, ktorý sa nevykonáva sústredene v blízkom kontakte členov tímu. Snaha je o preskúmanie problémov, ktoré môžu vzniknúť pri práci doma, a to hlavne so sledovaním úloh, či zodpovednosti. Detailne opisujem nástroje pre podporu takéhoto projektu s uvedením skúseností, a to opisom nedostatkov a kvalít. Esej poskytne aj víziu návrhov pre podporné prostriedky, ktoré by mohli pomôcť k zlepšeniu manažmentu projektu v danej oblasti.

Motivácia

Témou eseje sú podporné prostriedky pre riadenie softvérového projektu. Hranice sú určené na oblasť ľudských zdrojov, zodpovednosti a sledovanie úloh. Esej ešte viac konkretizujem na softvérové projekty, ktoré neprebiehajú sústredene na jednom mieste vzhľadom na ľudské zdroje. A teda na riadenie ľudských zdrojov a pridelovanie zodpovednosti.

Dnes už je bežné, že ľudia môžu na projekte pracovať aj na geograficky odlišných miestach a pritom musia byť koordinovaní a riadení jednotne, tak aby správne prispievali k jednému cieľu. Nejedná sa iba o malé projekty, kde napríklad neexistuje spoločné miesto, alebo sa domov jednotlivca zdá byť vhodnejší a ekonomickejší pre podanie výkonu. Dokonca veľké projekty využívajú pre zrýchlenie vývoja softvéru fakt, že na rovnakej úlohe môžu pracovať ľudia na smeny, či žijúci v komplementárnych časových podmienkach. To znamená, že jeden pracuje, keď druhý spí, pritom môžu žiť na opačných stranách zemegule.

Dôsledky nevládnutia rozloženia ľudskej sily, zanedbania sledovania úloh, strata prehľadu o zodpovednosti môžu projekt ohroziť. Manažment takto postaveného projektu je o niečo zložitejší ako iné projekty. V nasledujúcich kapitolách sa pokúsím vysvetliť, ako je možné riešiť manažment v spomenutých situáciách. Uvediem postupne oblasti manažmentu, na ktoré som sa zameral, a pri každej poukážem na problémy, ktoré môžu vzniknúť pri projekte. Ďalej opíšem podporné prostriedky, ktoré

pri riadení projektu môžu pomôcť nie len manažérom, ale aj ostatným participantom. Prispievam informáciami utvorenými z vlastných skúseností a myšlienok, ktoré vznikli pri študovaní uvedenej literatúry.

Riadenie softvérového projektu

Watts Humphrey [2] prirovnal riadenie projektu ku riadeniu armády. Rozdiel je v tom, že na projekte kde prostredie je vojna môže manažér kontrolovať činy svojich podriadených tak, že iba dozerá. Na softvérovom projekte je však nutné pozrieť hlbšie na priebežné výsledky práce konkrétneho zamestnanca. Ako sa dá toto zariadiť v prípade, že podriadený pracujú doma?

Myslím si, že pri softvérovom projekte by bolo najlepšie, aby každý manažér bol aj informatik a opačne každý informatik manažér. V tomto prípade by nebolo nutné dohliadať na jednotlivých ľudí tak podrobne. Dalo by sa očakávať správne oznamovanie stavu, plánovanie seba samého a sledovanie vývoja zo strany vývojára, a taktiež lepšie chápanie manažéra, ktorý informácie od vývojára príma. Brooks uviedol, že programátor si polovicu času stráveného na projekte myslí, že už má 90% splneného [1]. Väčšie firmy v oblasti IT už dnes do popisu práce na pozícii IT manažér vyžadujú 20% času stráveného programovaním, alebo inou rýdzo informatickou prácou. Práve takto sa dá zaručiť, že manažér prijme a pochopí nevizuálnosť softvérových projektov, a pokúsi sa čo najobjektívnejšie vytvárať správy o stave.

Ľudské zdroje

Ľudia použítí na projekte sa v prípade štandardného projektu dostávajú na miesto, kde vykonávajú spoločnú prácu. Zjednodušíme manažment ľudských zdrojov na pridelovanie ľudskej sily na jednotlivé úlohy. Ak ľudia pracujú doma personálny manažment má uľahčenú prácu, čo sa umiestnenia týka. Rovnako sa jednoduchšie vyberá nový pracovník, pretože tu nie je ohraničenie geografickou polohou.

Zo strany ľudských zdrojov pre takýto projekt plynú výhody hlavne vo forme zjednodušenia zabezpečenia, ale nie v prípade kontroly času stráveného nad úlohou, produktivita a efektívnosť. Predikcia trvania je dôležitá pre prípadné rozšírenie ľudských zdrojov alebo ich zníženie. Aby sa dala vytvárať predikcia trvania projektu, je potrebné, aby práca na úlohe nebola transparentná a dala sa sledovať.

Sledovanie úloh

Okrem sledovania času stráveného na úlohe je teda zaujímavé sledovať aj aká je efektivita tohto času, a či sú úlohy riešené správne. Ak je člen tímu priamo s jeho manažérom, potom sledovanie týchto úloh môže byť formou konzultácií, či priamo ukážky. Manažér si tak vie dať do súvisu výsledky svojich podriadených. Ak má s úlohou niekto problém, môže byť včas presunutá, a projekt napreduje.

Avšak ak už je člen na inom geografickom mieste, potom existuje riziko, že neohlási svoje problémy, čo je pre ľudskú bytosť bežné a snaží sa problém riešiť sám. Toto správanie je netimové a potenciálne ohrozuje a spomaľuje projekt. Opäť platí, že

ak by tento človek mal aj manažérske skúsenosti, vedel by si sledovať úlohy sám. Jeho výstup by obsahoval aj správu o postupe, či časoch na úlohách. Hlavne tu však ide o prekonanie ľudskej individuality a o sociálne myslenie, ktoré u niekoho môže chýbať.

Zodpovednosť

V projekte musí niekto niesť zodpovednosť za úlohy. Je to tiež forma motivácie. Ak človek za svoje činy nenesie zodpovednosť tendencia je uľahčovanie práce a minimalizácia kvality výstupov. Tento fakt sa ešte intenzívnejšie prejavuje v projektoch, kde členovia tímu nie sú v blízkosti, pracujú doma.

Každý projekt by sa mal riešiť čiastkovo, čo znamená vzniknutie množstva úloh. Dôležité je, aby za výsledky bol vždy niekto zodpovedný, avšak nie je vhodné, aby za veľa úloh niesla zodpovednosť jedna osoba a taktiež je vhodné, aby prebiehala aspoň čiastočná kontrola. Na rozdelení zodpovednosti za úlohy je vždy vhodné dohodnúť sa vopred, rovnako ako sa vopred určia role. Zodpovednosť (kto čo rozhoduje) blízko súvisí s rolami na projekte (kto čo robí) [3]. Pridelenie zodpovednosti na vyššej úrovni znamená pridelenie zodpovednosti na častiach projektu, a nižšia úroveň znamená priradenie zodpovednosti na jednotlivých aktivitách. Celé rozdelenie teda vyžaduje nástroj, alebo všeobecne metodiku.

Podporné prostriedky ako nástroje

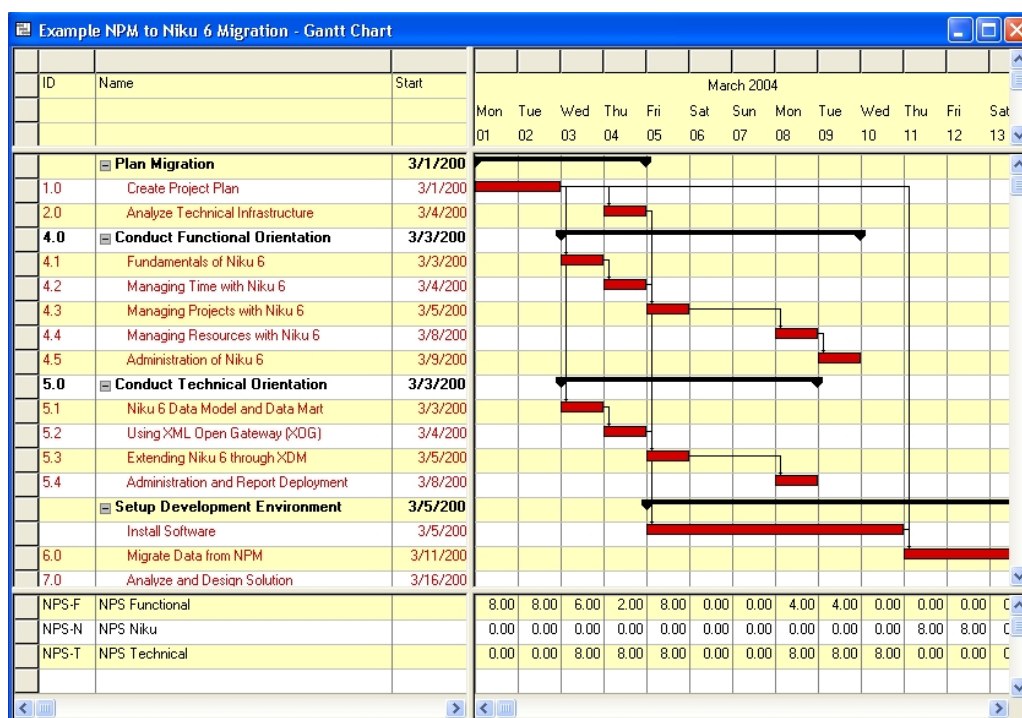
V tejto časti sa pokúsím opísať prostriedky, ktoré sa bežne používajú, ale aj tie neštandardné, s ktorými sa väčšina nestretla a mohli by pomôcť pri riadení projektu. Presnejšie sledovaním úloh, alebo zlepšením komunikácie, ktorá je prítomná pri sledovaní úloh, alebo jednoducho pri manažovaní ľudí.

Pre manažment projektu už dnes existujú nástroje, alebo súbory nástrojov, ktoré poskytujú množstvo užitočných pomôcok. Po preštudovaní a tiež skúške nástrojov pokladám za veľmi populárny Ganttov diagram (pozri Obr. 1). Ten prehľadne zobrazuje úlohy v čase a k nim mapuje mená a popisy týchto úloh. Je to základ pre plánovanie ale aj pre iné časti manažmentu. V ľavej časti sú úlohy a v pravej je časová priamka. Z vlastnej skúsenosti pokladám diagram skôr za pomôcku pre manažérov, lebo v skutočnosti sa ním programátor nedokáže riadiť, nie je schopný a hlavne nie je ochotný. Keď potom vývojár pracuje doma a snaží sa postupovať, diagram jeho činnosť neovplyvňuje. S diagramom sa možno stretnúť napríklad v balíkoch MS Project, Open Workbench, dotProject. Použitie tohto grafu sa však dá nahradiť obyčajnou tabuľkou s štandardnou štruktúrou, čo je vektor obsahujúci hlavne:

- úlohu
- meno
- čas začatia
- čas skončenia

– stav

S touto formou sa skôr stotožní aj obyčajný programátor. Pre mňa to znamená, že nie všetky vlastnosti nástrojov vyhovujú nie všetkým používateľom.



Obr. 1. Ganttov diagram

Podľa môjho názoru je príjemné, keď samotný nástroj ponúka v balíku manažment verzií a tiež súborový systém, do ktorého vidia všetci projektoví účastníci. Ak sa dodržiava spoločná štruktúra napríklad strom, potom každý vie, kde má dokument, či kód hľadať. Verziovanie, nielen čo sa kódu týka, šetrí množstvo času hlavne pri nutnosti vrátenia sa o krok späť. Tiež je veľmi vhodné pre tím, ktorý pracuje oddelene, keď si môže každý z ktoréhokoľvek miesta s pripojením na internet stiahnuť tú najnovšiu verziu a pokračovať v úlohe. Sú však aj projekty, kde nie je možné vytvorenie úložiska verzií pre účely distribúcie a vzdialenej spolupráce. Ide hlavne o projekty, kde sa úzko pracuje napríklad s hardvérom, alebo sa pracuje s dátami, ktoré sú citlivé. Verziovanie už v tomto prípade tvorí iba úlohu prípadnej záchrany a je dostupné iba na lokálnej sieti. Takéto projekty však nemôžu byť realizované jednotlivcami doma a preto sú mimo dosahu tejto eseje.

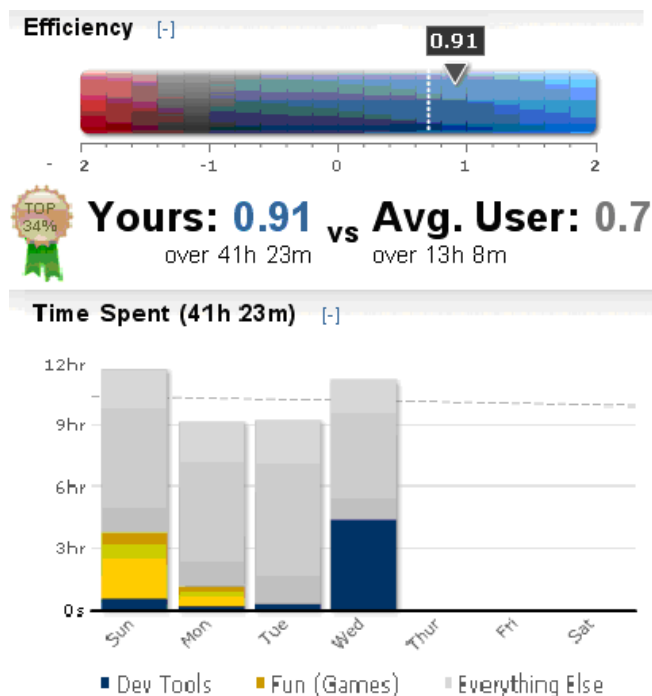
Mne ako členovi tímu, ktorý rád pracuje doma vyhovuje, ak som v priamom spojení s tímom a preto rád používam komunikátor na živé riešenie problémov. Naozaj som sa stretol aj s nástrojmi, ktoré obsahujú vlastné komunikátory a tiež ma zaujali

spôsoby ako spolupráca kódu, alebo dokumentácie. Možnosti ako sledovanie zmien v dokumentoch sú zaujímavé, ale v nasledujúcich podkapitolách poukážem aj na nástroje, ktoré by mohli umožniť viac.

Strážca času

Čas strávený na úlohách projektu priamo súvisí s jeho plánovaním. Čas strávený človekom pri úlohe je nutné sledovať. Sleduje sa spoločne s úlohami a s ich vývojom. Pri projektoch, kde nie je priama kontrola, keď sú pracujúci doma je na mieste sledovať, koľko času naozaj strávil človek na úlohe, alebo dokonca, koľko zo svojho času naozaj venoval práci.

Nestretol som sa so žiadnym nástrojom, ktorý by dokázal rozumne a dôveryhodne strážiť ako sa na úlohách pracuje. Chcem však upozorniť na program RescueTime, ktorý pracuje ako aplikácia konzumujúca webovú službu. Ide o posielanie informácií o aplikáciách, ktoré sa používajú u používateľa. Mám skúsenosť s touto aplikáciou, aspoň v jej štandardnej osobnej verzii, ktorá monitoruje iba jedného človeka. Vďaka tejto aplikácii si človek môže pozrieť, ktoré aplikácie a koľko používal (pozri Obr. 2). Sám môže označiť, ktoré aplikácie sú pre neho produktívne a aplikácia mu potom vie počítať aj efektívnosť času. RescueTime existuje aj v odrode pre tím. Tam je možné porovnať sa nie s konkrétnou osobou, ale s priemerom v skupine. Túto aplikáciu si viem predstaviť ako sledovač nielen mien aplikácií, ale v konkrétnych vývojových štádiách ako sledovač súborov v projekte, častí programu na ktorom sa pracuje. Dalo by sa tak úplne presne zistiť, koľko človek len tak prezerá kód, koľko študuje dokumentáciu, a koľko naozaj tvorí. Ďalej by sa dala zistiť participácia na konkrétnych úlohách, čo by ocenil každý zamestnávateľ. Vie sa sám nájsť v rebríčku čo postačuje k tomu, aby sa globálne efektívnosť zvýšila. Myslím si, že pridanie podobného nástroja do populárnych balíkov by bol výnimočný príspevok pri sledovaní úloh, ale aj ľudských zdrojov. Keď sa už dnes kontrolujú príchody a odchody na minútu presne, zaujímavejšie by bolo rozpoznať aj aktivitu, efektívnosť a produktivitu, ktorú človek na svojej úlohe vyvinie.



Obr. 2. Výstup RescueTime

Správca verzii

Balíky nástrojov dnes väčšinou poskytujú správu verzii. Jednoduché verziovanie sa dá chápať aj obyčajný spravovaný súborový systém kam môžu projektový participanti prispievať. Verziovanie však nie je iba nástroj, ktorý plní svoj primárny cieľ. Sú aj také nástroje, ktoré okrem verziovania vedú aj sledovať aktivitu používateľa. Obsahujú reportovanie, štatistiky vo forme grafov a podobne. Takéto rozšírené verziovanie patrí podľa môjho názoru tiež do prostriedkov, ktoré môžu pomôcť identifikovať stav úloh, kontrolovať zodpovednosť a teda hľadať vinníka. Práve ak nie sú členovia tímu pokope, tento nástroj môže zabezpečiť, aby zodpovednosť bola správne vyhľadaná a takisto pridelená. Autori častí kódu sa udržiavajú v úložisku a kedykoľvek môže byť vykonaná kontrola. Tam sa zistí, keď úloha nesúhlasí s najvhodnejšou osobou. Ďalej sa hľadajú chyby a za tie potom nesú autori pridelenú zodpovednosť.

Pre Visual Studio som sa stretol s nástrojom AnkhSVN. Tento nástroj poskytuje aj grafické rozdelenie aktivít, ktoré sa vykonali na spoločnom projekte. Teda poskytuje aj informáciu produktivite človeka v porovnaní s jeho kolegami (pozri Obr. 3). Dajú sa tu tiež vytvárať reporty, v ktorých môžu používatelia označiť svojho spolupracovníka za toho, ktorý zanesol do kódu chybu. Takéto spravovanie je síce prehľadné, ale ja si viem predstaviť, ako by sa priamo napojila matica zodpovednosti na časti projektu.

Písmená určujú druh zodpovednosti. Z angličtiny S - vyžaduje zrušenie, R vyžaduje revíziu, A –oboznamovaný, P - zúčastnený, I – vyžaduje vstup.

PERSON \ PHASE	A	B	C	D	E	F	...
Requirements	S	R	A	P	P		
Functional	S		A	P		P	
Design	S		R	A	I		P
Development		R	S	A		P	P
Testing			S	P	I	A	P

Obr. 4. Matica priradenia zodpovednosti.

Sústredme sa na rolu vývojára. Matica by sa musela viac špecifikovať na jednotlivé konkrétne úlohy, aby som v svojom pracovnom prostredí našiel aj informáciu o tom, na čom by som mal pracovať a akým spôsobom. Či iba revidovať, alebo priamo tvoriť. Napojením napríklad na správu projektu a verziovanie by sa dalo prehľadne zistiť, čo môžem s časťami kódu robiť.

Ak by som bol tester, matica by sa napojila na testovací nástroj a dostal by som iba potrebné prístupy ku aplikácií, či databáze. Tým by sa obmedzilo riziko pochybenia vo veci, ktorej nerozumiem, a v niektorých prípadoch aj zvýšila bezpečnosť ako napríklad v databáze banky. Reportovanie chýb, ktoré je vo väčšine spomenutých nástrojov samozrejmosťou, by potom bez manuálneho vyberania zodpovedného vývojára umožnilo jeho automatické pripojenie a pridelenie úlohy presne. V testovacích štúdiách, ako je napríklad Mercury, nebolo možno automaticky napojiť zodpovednú osobu. Myslím si, že táto inovácia by zrýchlila proces pridelenia úloh a ich samotné riešenie. Bežne sa stáva, že kód je od konkrétnej zodpovednej osoby menej kvalitný a nemalo by sa stať, že táto osoba potom bude opravovať svoje chyby sama na úkor celého projektu.

Komunikácia a on-line kolaborácia

Pridelenie nových úloh človeku, alebo ich priebežné sledovanie a následné prerozdeľovanie podľa kapacít prináša potrebu správnej komunikácie. Problém je niekedy správne opísať úlohu, či jej stav priamo jej zadávateľom. A v prípade nutnej neosobnej kolaborácie sa problém znásobuje.

Ak chce manažér, alebo zadávateľ novej úlohy, aby bola správne vykonaná a pridelená človeku, potom by mal použiť nástroj, ktorý umožňuje aj diskusiu, alebo rýchle posielanie správ. Väčšina nástrojov má portál, v ktorom sa dá úloha zapísať a k nej pridať komentár. Úloha sa prideli manuálne človeku, a potom na nej zodpovedná osoba pracuje tak, ako ju pochopila. Komunikátory v balíkoch nástrojov

existujú bežne. Zaujímavé je porozmýšľať o ich kvalite a dostatočnosti. Pre vysvetlenie a zadanie úloh na diaľku sa dá použiť aj virtuálna tabuľa, kde všetci zúčastnení okrem písania môžu aj kresliť náčrty a vizualizácie ich problémov. Zaujímavá je aj aplikácia Gobby, ktorá dovoľí písať spoločný dokument viac ľuďom súčasne tak, že navzájom môžu pozorovať kurzor spolutvoreu.

Záver

Poskytol som uvedenie do problematiky a ďalej príspevok v danej tematike. Vo forme hodnotenia nástrojov a návrhu vylepšení, či noviniek som priniesol ďalší z pohľadov na softvérový projekt. A to nie len na projekt, ktorý sa vykonáva tímom kooperujúcim na diaľku, ale aj na ostatné typy projektov, kde sa dajú využiť spomínané vylepšenia. Vyzdvihol som problémy, ktoré sa vyskytujú vo všetkých projektoch, ale iným spôsobom sa musia riešiť práve v projektoch mnou vyšetovaných.

V myšlienkach som často preniesol váhu nosnej témy na ďalšie oblasti manažmentu, vždy účelne a s prínosom. Prispel som poznámkami tiež ku manažmentu kontroly, alebo manažmentu plánovania, nakoľko všetky tieto témy spolu úzko súvisia.

Podľa môjho názoru sa mnohé so spomínaných myšlienok dajú realizovať. Taktiež predpokladám, že táto esej nielen pomôže vytvoriť základ pre ďalšiu štúdiu danej problematiky, ale aj poskytne inšpiráciu pre tvorbu nových nástrojov pre manažment projektov. Tie sú v súčasnej dobe populárne, ale hlavne potrebné.

Použitá literatúra

1. Brooks, Frederick P. *The Mythical Man-Month*: Addison Wesley 1995.
2. Humphrey, W.S, *Why Big Software Projects Fail: The 12 Key Questions*. *The Journal of Defense Software Engineering*, 2004.
3. *A Guide To The Project Management Body Of Knowledge (PMBOK Guides)*, Project Management Institute, 2004.

Annotation

How to manage team members in their home?

This essay is focused on managing human resource in specific conditions. My goal is to bring different point of view on software project. In the scope of the topic I discuss common problems, such as tracking tasks or assigning responsibility. My ideas and know-how is mapped to useful tools, which can be used to support project management. I tried to solve well known problems, which may occur in distributed projects where co-workers work in different geographical places, especially in their home. This essay also provides vision and design of tools which should effectively help in mentioned scope.