

VYUŽITIE PODPORNÝCH PROSTRIEDKOV V RÁMCI OPEN SOURCE PROJEKTOV

*Jednoduchá cesta nie je vždy tá dobrá. Ja hovorím
zložitá cesta je zlá.*

Marián Florek

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

Abstrakt. *Koordinácia a riadenie vývojového tímu si od projektového manažéra vyžaduje informovanosť o aktuálnom stave a rozdelení úloh v projekte. Na prehľad a správu týchto aktivít existuje široká škála podporných prostriedkov, ktoré v rámci softvérového projektu zohrávajú dôležitú úlohu. Pri veľkých projektoch sú tieto nástroje potrebnou, a priam nevyhnutnou podmienkou úspechu. Opodstatnené uplatnenie našli v komerčnej sfére, kde je motivácia k dodržiavaniu termínov odovzdávania pridelených úloh zrejmá. Esej uvažuje prípad využitia podporného nástroja pre sledovanie a riadenie úloh v rámci Open-source projektov. Pojednáva projekty, ktoré sú vyvíjané komunitou a vývojárom sa môže stať v podstate „hocikto“. Motivácia takéhoto vývojára je diskutabilná a manažér musí pri manažovaní takéhoto tímu riešiť problémy s ľudskými zdrojmi, ako je záťaž alebo čas trvania úlohy, oveľa častejšie, pričom riešenia nie sú vždy priamočiare a problémy sa môžu často opakovať.*

Kľúčové slová: *podporné prostriedky, open-source, voľný softvér, manažment projektu, vývoj softvéru*

Úvod

Riadenie projektov, či už softvérových alebo aj z nejakej inej oblasti, na ktorom pracuje tím s väčším počtom členov, si v dnešnej dobe manažéri už ani nevedia predstaviť bez podporných prostriedkov.

Nástroj, ktorý ma slúžiť ako podporný prostriedok pre softvérový projekt, v sebe zahŕňa všetku funkcionality spojenú s činnosťami v rámci životného cyklu vývoja softvéru, ako napríklad pridelenie úloh, komunikáciu, generovanie súhrnných správ, vytváranie grafov (Ganttov graf, graf ľudských zdrojov), atď. Nesie údaje o projekte, členoch tímu, úlohách, či časových termínoch. Jeho hlavnou úlohou je odbremeniť pracovníkov od papierovania, zefektívniť komunikáciu vo veľkých ale aj malých tímoch a prehľadným a zrozumiteľným spôsobom prezentovať informácie manažérovi, poprípade upozorniť ho na kritické situácie, aby mu umožnil adekvátne reagovať.

Na tom, že použitie správneho podporného prostriedku v rámci vývoja softvéru nevyklučuje jeho úspech sa všetci odborníci zhodli a aj laik, keď sa dobre zamyslí, môže s týmto názorom iba súhlasiť. Ako je to však s projektmi vyvíjanými komunitou ako open-source? Oplatí sa nasadiť podporné nástroje aj na takéto projekty, alebo prevládnu nevýhody ich využívania, vzhľadom k tomu, že neboli presne určené na takéto využitie? Na tieto otázky sa pokúsím nájsť odpoveď v nasledujúcom texte.

Pre spresnenie pojmov open-source projekt, je projekt, v rámci ktorého je vytváraný voľný softvér dostupný pod licenciou GNU GPL (General Public Licence) [4]. Z veľkej časti sa na jeho vytváraní podieľa komunita, prevažne zbehlá v IT oblasti, čo zvyšuje porozumenie v rámci sledovania chýb a znižuje nároky na dokumentácie čo sa týka zrozumiteľnosti pre široké spektrum ľudí.

Ako definovať úspech

V komerčnej sfére za úspech môžeme považovať spokojnosť zákazníka, ak sme pracovali na zákazke, poprípade, či vyvinutý softvér ide na odbyt. Pri projektoch vyvíjaných komunitou, s licenciou GPL, nie je možné určiť vlastníka a nie je možné softvér predať. Preto je náročné určiť, či projekt bol úspechom alebo neúspechom. Na tému ako odmerať úspech open-source projektu sa vedú mnohé diskusie, bez jednoznačne akceptovateľného výsledku [3]. Toto je hlavná príčina prečo sa nedá objektívne posúdiť, či sa pri využití podporných prostriedkov dosiahol úspech častejšie ako v prípadoch, kde sa podporné prostriedky nevyužívali.

Keby sme za kritérium úspechu zobrali počet stiahnutí, tak napríklad na portáli sourceforge.net z ich štatistiky má prvý v zozname okolo 550 000 000 celkových stiahnutí, pričom už 2700-tý má pod 100 000 stiahnutí a 31 500-tý má iba 2000 stiahnutí čo je oproti prvému neuveriteľne málo. Na portáli je registrovaných cez 330 000 projektov. Ak si za kritérium kvality zvolíme nad 2000 stiahnutí, tak iba 9,5 % projektov bolo úspešných. Z toho vyplýva že prevažná väčšina projektov bola neúspešná, čo nemusí byť pravda.

Keby sme ako kritérium zobrali rozsah projektu, to jest počet súborov, prípadne počet riadkov kódu, bolo by to asi zavádzajúce. Rôzne problémy si vyžadujú rozsahovo rôzne riešenia a tieto rozdiely sú určite ovplyvnené aj implementačným prostredím.

Tieto spôsoby hodnotenia sú čiastkové a môže sa stať, že pri jednom projekte dávajú korektné hodnotenie a pri druhom projekte môžu dávať opačný výsledok, aj keď bol projekt úspechom. Korektné posúdiť, či bol projekt úspešný, môžu asi iba ľudia čo sa na ňom podieľali a to tak, či z daného vstupu dostali požadovaný výstup.

Z tohto dôvodu nechcem posudzovať či bol projekt úspešný vďaka podporným prostriedkom. Skôr by som sa chcel zamerať na problémy, ktoré by mohli vzniknúť využitím týchto nástrojov pri vývoji open-source projektu.

Podporný prostriedok pre open-source

Prostriedok by mal byť jednoznačne na spôsob web aplikácie, aby bol dostupný komunite. Vo svete je obrovské množstvo bezplatných aplikácií. Tieto weby ponúkajú širokú škálu nástrojov na manažovanie projektu. Manažér má dve možnosti, buď si bude prevádzkovať vlastný server alebo využije možnosti viacerých webov, kde založí svoj nový projekt a tam ho bude spravovať. Druhý spôsob má výhodu v tom, že takéto weby už majú určitú komunitu a aj vo vyhľadávачoch majú určitú relevantnosť, čo dopomáha k zviditeľneniu.

Počet členov v tíme alebo veľkosť tímu v prípade open-source projektu nie je smerodajné pre výber prostriedku, vzhľadom k tomu, že nemá presne určený počet. Členovia môžu pribúdať, strieďať sa a odbúdať. Určite obsahuje aj stálu základňu, kde sú ľudia s prvotným nápadom projektu a ľudia ktorí iniciovali projekt. Avšak malé tímy (2-3 členovia), asi aj v komerčnej sfére, dokážu byť efektívne riadené aj bez podporných prostriedkov. Dôvodom, prečo manažéri open-source projektov siahajú po týchto prostriedkoch je najmä úložisko samotného produktu/softvéru, nezaujímajú ich pomoc pri riadení. A v prípade, že sa projektu prestanú venovať ostane dostupný na webe, a ak sa nájdú záujemcovia môžu v ňom pokračovať alebo ho vpracovať do svojho projektu. Tu vidím zásadný rozdiel v motivácii zvolenia podporných prostriedkov pre malé tímy v open-source a komerčnej sfére, kde ich využívajú hlavne k odbremeniu pracovníkov od papierovania a možno vizualizáciu stavu projektu.

Dôležitým faktorom, v čom v podstate spočíva ďalší rozdiel medzi komerčnými projektmi a open-source projektmi, je časové obmedzenie. V komerčnej sfére je dodržanie termínov hlavným faktorom úspechu. Pričom pri open-source vývoji môžeme povedať, že sa nikam netreba ponáhľať, lebo časové obmedzenia nie sú až tak dôležité.

Ďalším faktorom sú ľudské zdroje. V komerčnej sfére si človeka možno najat', na rozdiel od open-source, kde ak nie je o projekt záujem tak sa vývojári veľmi ťažko zháňajú. Spomeniem ešte dva faktory a to je komunikácia a testovanie, ktoré spolu úzko súvisia.

Časové obmedzenie

Časové obmedzenia v projekte úzko súvisia s plánovaním a pridelovaním úloh. Samozrejme ak niekomu na vývoji projektu záleží, tak určite bude mať nejaký časový plán. Aby sa dosiahol úspech, tak treba k vývoju pristupovať seriózne, čo znamená časový plán. Preto túto možnosť netreba hneď zavrhnúť a aj určitý typ plánovania by open-source projekt mal obsahovať.

Avšak neviem si celkom dobre predstaviť využitie Ganttovho grafu. Vychádzajme z toho, že vývojár má nejakú prácu, domácnosť, záľuby (kde môže patriť aj účasť na open-source projekte). Teraz popri tom všetkom si nájde pár hodín týždenne na open-source

4 Marián Florek

projekt, pričom niekedy je to viac a niekedy menej. Odkiaľ má manažér, ktorý robí plánovanie úloh vedieť koľko a kedy môže s týmto vývojárom počítať. Určite musí byť viac benevolentnejší, čo sa týka časových nárokov pre jednotlivé úlohy. S tým je spojená aj réžia pridelenia ďalších úloh, kde sa možno častejšie stávajú prípady, že niektoré úlohy sa spĺňajú skôr ako bolo plánované. V podstate to nie je nič zlé, až na prípady pridelenia úloh typu Koniec- štart, ktoré sú závislé na dokončení iných úloh.

Burndown grafy by sa dobre dali využiť na časový odhad celého projektu. Teda ak manažér nemá skúsenosti s open-source projektmi a ich plánovaním. Dokáže po určitej dobe pomocou tohto grafu odhadnúť trvanie projektu. Keď sa mu vykryštalizujú jednotlivé „pracovné doby“ členov tímu.

Ludské zdroje

Vývojári, čiže ľudské zdroje sú esenciálne dôležité. Tu zohráva dôležitú úlohu motivácia, ktorá núti pracovníkov pracovať na pridelených úlohách. V komerčnej sfére je dostatočnou motiváciou finančné ohodnotenie a kariérny rast, čo zase znamená lepšie finančné ohodnotenie. Pri práci na open-source projekte je motivácia všetko len nie finančné ohodnotenie. Môže sa jednať o nadšencov, ktorí to berú ako určitý druh zábavy odborníkov v oblasti, ktorí sa vo svojom zamestnaní nevenujú tejto problematike ale chcú sa v tejto oblasti stále zdokonaľovať, vývoj softvéru v rámci sebazvedávania, alebo iba majú chuť spraviť určitý softvér dostupný aj zadarmo.

Nižšie uvedené nezaručuje aby vývojár z open-source projektu nevystúpil, alebo aby sa správal aktívne, možno až na nadšencov. Jedná sa o takzvaného dobrovoľného pracovníka. Graf ľudských zdrojov je v tomto prípade veľmi dôležitý, aby sa manažér vyhol preťaženiu niektorého z „dobrovoľníkov“, čím by oňho mohol prísť. Podľa môjho názoru by nemusel ani využívať kapacitu ľudských zdrojov na 100%. Ale to záleží od manažéra a od prístupu jednotlivých vývojárov.

V podstate v tíme open-source projektu nie sú striktne stanovené pozície v tíme, nemusí byť presne určené, kto sa komu zodpovedá. Všetci vývojári pracujú na jednej úrovni a dá sa povedať, že pozície sa striedajú. Častá situácia je taká, že nejaký odborník sa podieľa iba na určitej časti/module a viac už k vývoju neprispieva, ale rieši podobné problémy na iných projektoch [4]. Táto skutočnosť iba upevňuje potrebu matice zodpovednosti, alebo niečoho na jej spôsob, aby sa predišlo chaosu a bolo každému jasné čo od neho ostatní členovia očakávajú, ak sa chce do projektu zapojiť.

Open-source projekty, v prevažnej väčšine nevytvárajú typovo nový softvér ale pracujú na vytvorení voľne dostupnej konkurencie pre komerčné riešenia daných problémov. Táto skutočnosť v neposlednej miere poskytuje inšpiráciu a umožňuje detekciu chýb a možné zlepšenia oproti komerčnej verzii produktu. Väčšina vývojárov vie ako to v danej oblasti chodí, v lepšom prípade sú to odborníci, ktorí sa s problematikou stretli aj v pracovnom živote. Čo môže zaručiť úspech, ale hlavne to odbúrava potrebu dôkladnej analýzy, čím ostáva viac času pre ostatné etapy v rámci životného cyklu vývoja softvéru.

Komunikácia a testovanie

Komunikácia medzi členmi tímu je dôležitá súčasť vývoja. Väčšina podporných prostriedkov poskytuje určitý spôsob komunikácie, aj keď to nepatrí medzi základnú funkcionálnosť týchto prostriedkov. Uvediem príklady najčastejšie používaných prostriedkov komunikácie:

- emailová notifikácia
- emailové listy
- diskusné fóra
- dorozumievanie sa v reálnom čase (instant messaging/IM)

Diskusné fóra a emailové listy na podporu pre používateľov, kde je dôležité sprístupniť tieto komunikácie komunite (človek sa poučí z chýb druhých). Emailová notifikácia na pridelenie úloh a IM pri komunikácii, samotnom programovaní a analýze medzi vývojármi.

Z hľadiska open-source projektu je pre podporný prostriedok priam nutnosť obsahovať určitý spôsob komunikácie, najlepšie viac druhov. Je veľmi pravdepodobné, ba priam isté, že jednotliví účastníci nepochádzajú z rovnakého regiónu alebo štátu. Táto skutočnosť vylučuje tímové stretnutia, kde by bol každý fyzicky prítomný. V dnešnej dobe už komunikácia medzi jedným a viacerými nie je problém. Napríklad VoIP a videokonferencia dokáže nahradiť tímové stretnutia, čo už je ale asi dosť veľký extrém pri open-source projektoch. Toto manažérovi uľahčuje pridelenie a sledovanie úloh.

Určite k tomu, že podporné prostriedky obsahujú spôsoby komunikácie prispelo to, že v komerčnej sfére sa riešia projekty viacerými tímami na rôznych miestach.

Open-source vývojári zabezpečujú podporu pre používateľov, ktorí sa na nich obracajú s podozreniami na chyby, možno neopodstatnenými, ale to môžu posúdiť iba samotní vývojári. V mnohých prípadoch sa tieto žiadosti opakujú a vo väčšine prípadov sa o chyby ani nejedná. Takýto otvorený prístup má aj svoje výhody, pretože odhalí väčšinu chýb a to v krátkom čase, ak je predpoklad, že o projekt je záujem. Na druhej strane sa dá povedať že používatelia, čiže komunita, určitým spôsobom otravuje vývojárov a uberá im z času, ktorý by mohli využiť na programovanie alebo iné činnosti spojené s posúvaním projektu k úspešnému koncu. Tu je dôležité nájsť rovnováhu.

Stopovanie chýb (tracking), je v rámci open-source projektov veľmi dôležité a nahrádza testovanie. Do testujúcej komunity patria aj samotní vývojári.

Problémy aplikácie podporných prostriedkov na open-source projekty

Podporné prostriedky v komerčnej sfére boli vyvíjané na základe určitého obrazu štruktúry, či už tímu alebo organizácie. Táto štruktúra si zakladá na autoritatívnom princípe, kde je hierarchicky určené kto sa komu zodpovedá alebo kto komu prideliuje úlohy. Na základe tohto princípu vznikli dve skupiny. Tí vyšší čo zadávajú projekt a tí nižší, ktorí ho riadia a realizujú.

Jedným z hlavných princípov open-source projektov je, že nikto nemá administratívnu autoritu a vývojár sa riadi sám. V podstate sám si vyberá čo, kedy, ako a prečo bude robiť. Z toho vyplýva chaos a ešte väčší chaos. Tu by sme mohli zaviesť tiež

6 Marián Florek

určitý nepísaný hierarchický systém, že vyššiu váhu pre projekt majú ľudia, ktorí prispeli viacerými príspevkami, či už sa jedná o zdrojový kód alebo návrhy riešenia problémov [4]. Z môjho pohľadu sa tento princíp nedá a ani by sa nemal aplikovať na projekty väčšieho rozsahu aj keď sa jedná o vývoj open-source softvéru.

V týchto dvoch opozitných princípoch vidím najväčší problém využitia podporných prostriedkov pri open-source projektoch.

Keď sa pozrieme na motiváciu využívania podporných prostriedkov v komerčných projektoch, ide skôr o to aby čo najefektívnejšie, najlepšie a najrýchlejšie vytvorili softvér, ktorý by spĺňal požiadavky používateľov tohto softvéru.

Motivácia pri open-source projektoch je vytvoriť informačné kanály a zdokumentovať vývoj softvéru pre iných vývojárov, ktorí by sa s projektom stretli. Skôr sa zameriava na to, prečo a ako boli veci riešené ako na to, kto, kedy a koľko, čo má v komerčnej sfére opodstatnenie.

Napriek týmto rozdielom je opodstatnenie využívať prostriedky korektné pri oboch typoch vývoja. Pričom oba využívajú určitú funkcionálnu funkciu viac alebo menej.

Záver

Po bližšom skúmaní aplikácie podporných prostriedkov na open-source projekty, som narazil na situácie, kde ich využitie nie je jednoznačné. Týka sa to najmä časových obmedzení a plánovania, kde musí byť manažér veľmi obozretný a prispôsobivý. Nemyslím si však, že ich použitie má negatívny vplyv. Sledovanie úloh a záťaže vývojárov, je aj v tomto prostredí veľmi dôležité, aby sa včas odhalili nedostatky. Pri opise problémov som sa snažil využiť svoje osobné skúsenosti z bakalárskej práce a zameriavať sa na tie najzaujímavejšie.

Jednou z markantných prerekvizít využitia podporných prostriedkov je chcenie celého tímu, aby práca na projekte bola na určitej úrovni. Podľa môjho názoru väčšina vývojárov sa s podpornými prostriedkami stretáva v profesionálnom živote a ich uplatnenie preferujú a akceptujú aj v rámci open-source projektov. Pričom sprehľadňujú riadenie a uľahčujú prácu, nielen manažérom ale všetkým členom tímu.

Druhou podmienkou ich využitia je dostatočne zložitý projekt s určitou stabilnou tímovou základňou. Nesmiem zabudnúť ani na testovanie komunitou, čo závisí od záujmu o projekt, i keď toto nevyklučuje klasické testovanie.

Mojím záverom je, že v oboch sférach, komerčnej i open-source, sa používajú podporné prostriedky iným spôsobom (kladú dôraz na inú funkcionálnu funkciu). Myslím že je to dobré v tom, že sa rozvíja všetka funkcionálna funkcia podporných prostriedkov. Nie iba tá na ktorú kladie dôraz jedna sféra.

Použitá literatúra

- [1] AHMAD, N., LAPLANTE, P. A. Software Project Management Tools: Making a Practical Decision Using AHP. In *Software Engineering Workshop: Proceedings of the 30th Annual IEEE/NASA*. Columbia (Maryland), April 2006, p. 76 – 84.

- [2] HERBSLEB, J. D. Global Software Engineering: The Future of Socio-technical Coordination. In *Future of Software Engineering*. Minneapolis (Minnesota), May 2007, p. 188 – 198.
- [3] KOCH, S. Exploring the effects of SourceForge.net coordination and communication tools on the efficiency of open source projects using data envelopment analysis. In *Empirical Software Engineering*, 2009, vol. 14, no. 4, p. 397-417.
- [4] SCACCHI, W. Free/open source software development. In *Proceedings of the the 6th Joint Meeting of the European Software Engineering Conference and the ACM SIGSOFT Symposium on the Foundations of Software Engineering*. Dubrovnik (Croatia), September 2007, p. 459-468.

Annotation

Use of management tool in open-source projects

Coordination and management of the development team demands from the project manager, information about the current status and distribution of tasks in the project. There is a wide range of support management tools to overview and administrate these activities. These tools play an important role in the software project. For large projects, these tools are necessity necessary, almost a inevitability of success. Their application is Reasonable in the commercial sphere, where the motivation for on-time fulfillment of assigned tasks is clear. Essay considers the case of using management tools in open-source projects to monitor and manage tasks. Discusses projects that are developed by community and a developers can become basically "anybody". Such motivation of developer is questionable and manager by managing such a team has to solve problems with human resources, such as load or duration of tasks, more often. The solutions are not always straightforward and problems often repeat.