

PLÁN, NA ROZPOZNANIE ZLÉHO PLÁNOVANIA

Je to zlý plán, ktorý nepripúšťa žiadnu zmenu.

Juraj Belanji

Slovenská technická univerzita
Fakulta informatiky a informačných technológií
Ilkovičova 3, 842 16 Bratislava
juraj[.]belani[zavináč]yahoo[.]com

Abstrakt. *Keď sa pred človeka postaví nejaký problém, prvá inštinktívna vec, ktorá ho napadne, je vytvoriť si plán na vyriešenie daného problému. Aj keď ten plán nie je oficiálne napísaný, každý človek si v mysli nejaký plán vytvorí. Na vytvorenie softvérového projektu je taktiež potrebný plán. Plán je základný kameň vytvorenia dobrého softvérového produktu. Tak ako je každý človek jedinečný, tak aj každý nový plán musí byť jedinečný. Zle vytvorený projektový plán je najčastejším dôvodom zlyhania softvérových projektov. V eseji sa identifikujú chyby, ktoré sa môžu stať pri tvorbe plánov. Poukazuje sa na nedostatky zlého plánovania a ponúknutý je spôsob ako rozpoznať zlý plán. Esej opisuje aj najčastejšie príčiny zlyhania plánov. Ponúknutý je spôsob ako predísť zlyhaniu plánov a ukázané je, že plán nie je iba niečo, čo zaberá čas, ale viac prostriedkov na vylepšenia procesu tvorby softvérových systémov.*

Kľúčové slová: *plán, plánovanie, zlé plány, preanalyzovanie, univerzálne plánovanie, zlyhanie plánu, softvérový produkt*

Úvod

Podľa [2] sa plánovanie vo všeobecnosti môže opísať ako proces výberu cieľa a ustanovenia zásad, postupov a programov, potrebných na dosiahnutie toho cieľa. Plánovanie projektu je možné opísať ako zavedenie vopred ustanoveného postupu práce. Na priebeh samotného projektu však vplýva veľké množstvo vonkajších faktorov, na ktoré manažér plánovania nemusí byť pripravený. Jeden z takých faktorov je povedzme nemožnosť odhadnutia koľko ľudí zostane na projekte po jeho skončení. Pri plánovaní

2 Juraj Belanji

softvérových projektov taktiež niekedy nevieme presne odhadnúť zložitosť softvéru. Jedným z vážnych problémov je aj neviditeľnosť softvéru. Najväčším problémom, z vlastnej skúsenosti, je nevytvorenie dobrého a stabilného plánu, teda vytvorenie zlého plánu.

Ako však môžeme odhaliť zlý plán? Existuje spôsob ako sa vyhnúť zlému plánovaniu? Je plánovanie naozaj také dôležité?

Rozpoznanie zlého plánu

Najväznejším krokom, pri vytváraní plánu je rozpoznanie chýb v plánovaní. Každý človek, ktorý sa dá do plánovania, si musí najprv uvedomiť, že neexistuje dokonalý plán. Aj keď si možno môžeme urobiť v hlave predstavu o dokonalom pláne, musíme si uvedomiť, že aj ten náš „dokonalý“ plán, nie je v podstate dokonalý. Naveky sa dá niečo urobiť lepšie.

Taktiež si musíme uvedomiť, že pri vytváraní plánov nemôžeme byť veľký optimisti. Z vlastnej skúsenosti môžem povedať, že vytváranie optimistického plánu v 90% prípadov zlyhá. Čo je vlastne optimistický plán? Je to plán, v ktorom naplánujeme všetko tak, akoby sa nič zlé nemalo počas priebehu projektu stať. Je to plán, ktorý termín ukončenia odhaduje vždy predčasne, ktorý nezahŕňa v sebe dostatočné riziká a podobne.

Na druhej strane, ako celkový opak optimistickému, je pesimistický plán. Pesimistický plán je taký plán, ktorý predvída, že predvídaná práca pôjde o dosť pomalšie a termíny budú o dosť neskoršie, ako je na to prvý odhad. V tomto prípade ale strácame dôveru zákazníka.

Keď nie optimistický a nie pesimistický plán, zostáva nám iba prostriedok, teda realistický plán. Ako však vytvoriť dobrý realistický plán, ktorý nie je ani priveľmi pesimistický ani priveľmi optimistický? Potrebujeme najprv identifikovať zlé plánovacie zvyky.

V [3] je opísaných 9 smrteľných hriechov plánovania projektu, ktorých by sme sa mali vyhýbať:

1. *Neplánovanie*
2. *Nebrať do úvahy všetky projektové aktivity, teda nedostatočné plánovanie*
3. *Neplánovanie rizík*
4. *Použitie rovnakého plánu pre každý projekt*
5. *Použitie cudzieho plánu, bez vlastného kritického premýšľania*
6. *Dovolenie plánu vzdialiť sa od projektovej reality*
7. *Plánovanie dopodrobna príliš skoro, teda preplánovanie*
8. *Plánovanie dobehnutia neskôr*
9. *Nepoučenie sa z minulých chýb*

Pri identifikácii zlého plánu si treba dať pozor na tieto chyby. Pozrime sa teraz podrobnejšie na jednotlivé zlé zvyky pri plánovaní.

Neexistujúci plán verus megaplán

Najčastejším dôvodom zlyhania plánu je vytvorenie plánu v hlave. Je to lepšie ako nevytvoriť si vôbec plán, ale skoro rovnako rizikové a zlé. Prvý problém je, že daný plán nemôžete sledovať, lebo sa vám počas práce na softvérovom projekte samotný plán striedavo mení. Druhý je, že aj keď môžete byť vedomý určitých rizík, môže sa stať, že pri strese počas práce zabudnete na niektorý aspekt, a samým tým vám projekt môže byť vážne poškodený.

Aj keď je vytvorenie malého plánu v podstate lepšie ako nevytvorenie žiadneho, na veľké softvérové projekty sa vytvorenie len hrubého plánu nehodí. Malý plán je pri veľkých projektoch skoro rovný neexistujúcemu plánu.

Naproti tomuto je druhý spôsob plánovania a to preanalyzovaním, čiže vytvorením „megaplánu“. Človek sa pri dlhom analyzovaní snaží aby kroky vývoju projektu určil čo najpodrobnejšie. Toto vie priviesť do takého stavu, že sa pri každej ďalšej analýze zistí, že nejaký krok v pláne treba rozdeliť, alebo zobrať do úvahy nejaké novo odhalené riziko späť s tým krokom. Rekurzívnym analyzovaním, čiže preanalyzovaním, sa v podstate vytvára pesimistický plán.

Plán s tohto dôvodu treba vytvárať použitím oboch metód. Takto vytvorený plán bude najrealistickejší a bude mať malú šancu zlyhať.

Ako príklad vytvorenia plánu zoberiem plán ktorý priebežne tvoríme na tímovom projekte. Najprv sa vytvorí hrubý návrh plánu, teda malý plán, ktorý bude obsahovať všetky základné kroky vývoja. Ďalšou analýzou každého kroku sa určia riziká, ktoré sú späté s daným krokom. Následne sa priebežne plánuje iba krátko do budúca (iba zopár krokov).

Ako som už spomenul, neexistuje dokonalý plán. Hore uvedením spôsobom sa ale dá predísť nečakaným problémom a to tým, že sa vytvoril jeden malý plán na celý projekt a časť toho plánu, teda pre najbližšie obdobie, sa rozpracovala.

Univerzálne plánovanie

V [1] je univerzálny plán definovaný ako ľubovoľná funkcia z množiny možných situácií S do množiny primitívnych akcií A . Podľa [1] je univerzálne plánovanie vo väčšine prípadov zlý nápad. Jeden všeobecný plán, nie je vhodný pre dva, alebo viac rôznych softvérových projektov. Je to naozaj tak?

Na vytvorenie univerzálneho plánu potrebujeme vedieť všetky možné akcie, riziká a spôsoby vytvorenia softvérového produktu. Pre nás nie je praktické vypočítať spôsob konania pre každú situáciu, v ktorej by sme sa mohli nájsť. V našom prípade ale univerzálne plánovanie môžeme vysvetliť ako vytvorenie jedného všeobecného plánu na riešenie viacerých projektov. Je toto až také zlé?

Vráťme sa k deviatim smrteľným hriechom plánovania. 4. hovorí o tom, že nesmieme použiť rovnaký plán pre viaceré softvérové projekty. 5. zas o tom, že nesmieme použiť cudzí plán bez vlastného kritického názoru. Keď tieto dva hriechy spojíme, pridáme k uzáveru, že je možné použiť už vytvorený plán, ale potrebujeme ho pre každú situáciu (čiže projekt) prispôbiť. Ako však rozpoznať, či pri plánovaní môžeme použiť aspoň sčasti nejaký už hotový plán?

4 Juraj Belanji

Prvú otázku, ktorú si potrebujeme postaviť je, či ten projekt má súvislosti z naším projektom, ktorý práve vytvárame. Nie je vhodné prebrať plán, ktorý sa vzťahoval na projekt, ktorý nemá žiadne body styku z projektom na ktorom teraz pracujeme. Ale pri podobných projektoch, podľa mojej mienky, nie len že je vhodné prebrať sčasti plán, ale takmer určite ho treba prebrať.

Prebratím častí plánu projektu by sme tak dostali rámcový plán, nad ktorým by sme sa potom museli zamyslieť, a ďalej prispôbiť k nášmu projektu a možnými rizikami, ktoré sú späté s ním.

Malý projekt verzus veľký projekt

Podľa [4], malé tímy pracujú úspešnejšie ako väčšie. Tak ako malé tímy pracujú lepšie ako veľké, tak je aj vytvorenie plánu vhodnejšie pre menší počet ľudí, ako pre veľký. Keď máme veľa ľudí, ktorý pracujú na jednom projekte, ťažko ich je sledovať. Prichádza k vytvoreniu viacerých úloh, plán sa stáva rozsiahlejším a to často prináša k vytváraniu nepresných plánov.

Analogický, v praxi sa ukázalo, že menšie projekty sú ľahšie zvládnuteľné po stránke manažmentu, ako veľké a rozsiahle projekty. Na takéto rozsiahle projekty je preto vhodné, ako sa už v praxi ukázalo, použiť metódu *rozdeľuj a panuj*. Rozdelenie rozsiahleho projektu na viaceré menšie časti, pomáha pri tvorbe plánov. Je samozrejmé, že je oveľa vhodnejšie plánovať na kratšie časové obdobie pre malé časti projektu, ako plánovať na dlhšie obdobie pre jeden rozsiahly projekt.

Zložité softvérové systémy je ťažké špecifikovať. Požiadavky na takéto komplexné systémy sa počas ich vývoja drasticky menia. Preto je aj ťažko plánovať pre projekty na vytváranie takýchto systémov. Delením projektu na menšie časti získavame viaceré výhody:

- Rozdelením projektu, môžeme zistiť ďalšie riziká, ktoré pôvodná analýza neodhalila. Tak je možné včas reagovať a vyhnúť sa neskorším problémom, ktoré môžu výrazne ovplyvniť projekt.
- Je ľahšie odhadnúť množstvo práce, ktorú je jeden človek schopný urobiť za menšie časové obdobie (povedzme mesiac), ako viacerých ľudí počas dlhšieho časového obdobia (povedzme rok). S tohto vyplýva, že sú plány robené nad menšími projektmi presnejšie ako tie, ktoré sú robené nad dlhodobými projektmi.
- Ľahšie sa určuje stav projektu. Z vlastnej skúsenosti môžem povedať, že viem oveľa lepšie oceniť dokončenie nejakého zadania v rámci týždňa, ako zadania, ktoré mám na celý semester.

Pre plánovanie projektov je ešte jedným veľkým problémom pridelenie úloh zlým ľuďom. Potrebujeme si dať veľký pozor na to, aby sme nepridelili nejakú úlohu (teda spoľahli sa s časťou plánu) ľuďom, ktorý nie sú vhodní pre tú úlohu.

Plánovanie je kontinuálny proces

Nesmieme zabudnúť, že jedným z najčastejších dôvodov zlyhania softvérových procesov je aj nesledovanie plánov. Tento problém môžeme rozdeliť na dve časti. Prvý je nedodržanie termínov plánu a druhý neplánovanie priebežne.

Je dôležité dodržať termíny. Nikdy nevieme, čo to môže mať za následky, ak sa termíny nedodržia. Aj na fakulte nás stále učia, že je dôležité dodržiavať termíny. Jeden z hriechov plánovania (konkrétne 8.) hovorí práve o tom. Nesmieme nedodržiavať termíny, lebo v tom prípade budeme musieť plánovať dobehnúť to neskôr. Je to ale jeden z smrteľných hriechov plánovania. Zistilo sa, že projekty, ktoré sa nedodržiavajú termínov, málokedy úspešne dobehnú stratený čas neskôr [3]. Väčšinou to prináša ešte ďalšie oneskorenia. Dôvodom oneskorenia je, že tímy pracujúce na projektoch nezačínajú robiť rýchlejšie a produktívnejšie, ale práve opačne. Pri riešení zisťujú rôzne chyby, ktoré nemohli odhaliť na začiatku projektu.

Druhým problémom ako už bolo spomenuté je neplánovanie priebežne. Z vlastnej skúsenosti viem, že aj keď si človek urobí plán, veľa vecí sa môže stať, ktoré mu ten plán porušia. Vytvoriť plán na začiatku projektu a neaktualizovať ho priebežne, teda držať sa toho rovnakého plánu, počas celej doby vytvárania projektu, je jedným z ukazovateľov, že projekt zlyhá.

Typickým prípadom je povedzme vytvorenie plánu pre šesťčlenný tím. Po mesiaci práce jeden člen tímu ochorie a nemôže pokračovať na realizácii projektu. Keby sa ďalej pokračovalo po pôvodnom pláne s tým, že by sa úlohy odchádzajúceho člena tímu rozdelili na zvyšných členov, prišlo by k zlyhaniu projektu, lebo by zvyšný piati ľudia nestíhali dokončiť nielen tie úlohy, ktoré im boli pridané, ale ani tie svoje.

V takomto prípade je potrebné pozastaviť sa a znovu vyanalyzovať plán a urobiť potrebné zmeny, ktoré by dosiahli to, že by sa projekt dokončil.

To, že je potrebné kontinuálne sledovať a meniť plán neznamená, že ho treba meniť stále. Podľa [2] zmeny plánov za behu nie je vždy vhodné. Meniť plány, ako už bolo poukázané, treba, ale tie zmeny nesmú byť veľké. Plán treba jemne upravovať. Myslím si, že ani nie je reálne počas práce na nejakom veľkom projekte očakávať, že veľká zmena v pláne bude pozitívne vplývať na projekt.

Takisto netreba staré, nesplnené úlohy rozširovať a meniť. Ak úloha nebola splnená, vhodnejšie je pridať novú, ktorá bude v sebe obsahovať aj časti starej úlohy, nie však tie, ktoré boli nesplniteľné.

Treba však brať do ohľadu, že sa plán nesmie rozsiahle zväčšovať, lebo by to zapríčinilo nedodržanie termínov.

Záver

Identifikovali sme chyby a problémy, ktoré sú najčastejším dôvodom zlyhania softvérových projektov. Ukázali sme ako identifikovať zlé plány a tieto chyby boli poviazané z typmi plánovacích procesov. Počas identifikácie problémov pri plánovaní a zlých plánov, sme ponúkli riešenie určitých problémov.

Ukázali sme, že nie je dobré vytvárať ani celkom optimistický ani celkom pesimistický plán, že nie je vhodné málo plánovať ale, že ani netreba preplánovať. Ďalej

6 Juraj Belanji

sme ukázali prečo netreba vytvárať univerzálne plány, ale taktiež sme uzavreli, že niekedy sa hodí prebrať niečo z už hotového plánu. Ukázalo sa, že to najmä šetrí čas, a čas sú peniaze. Ďalej sme hovorili o plánovaní pre malé a rozsiahle projekty, ako aj pre malé a veľké tímy. Na záver sme ukázali, že je potrebné plánovať priebežne.

Tieto riešenia a myšlienky sú iba subjektívne. Na riešenie problémov plánovania je možné ponúknuť množstvo plánov. Toto je iba jeden z možných alternatív.

Proces vytvorenia plánu pre softvérový projekt je zložitý a preto je potrebné nebrať túto tematiku zľahka. Prax ukázala, že je vždy vhodnejšie vytvoriť plán. Ten vytvorený plán, ako vidno z eseje, nielenže môže pomôcť človeku pri organizovaní času, ale aj výrazne môže zlepšiť kvalitu softvérového produktu.

Použitá literatúra

1. Ginsberg, M.L.: Universal Planning: An (Almost) Universally Bad Idea. *AI Magazine* 10, Vol. 4, (1989) 40-44.
2. Kerzner, H.: *Project Management: A Systems Approach to Planning, Scheduling and Controlling*. 10th edition. Wiley, 2009, ISBN: 987-0-470-27870-3
3. McConnell, S.: The nine deadly sins of project planning. In: *IEEE Software*, Vol. 18, No. 5, 2001, 5-7.
4. Rettig, M., Simons, G.: A project planning and development process for small teams. In: *ACM Communications Journal*, Vol. 36, No. 10, 1993, 45-55.

Annotation

A plan for recognizing a bad plan

When a man stands in front of a problem, the first instinctive thing that occurs to him is to create a plan to solve that problem. Although that plan is not formally written, each person creates a plan in his mind. For creating a software product, a plan is also needed. The plan is the cornerstone of creating a good software product. As each person is an unique individual, so must any new plan be unique. Poorly designed project plan is the most common reason for software project failures. This essay identifies errors that occur while creating a plan. It highlights the shortcomings of bad planning and also offers a way on how to recognize a bad plan. The essay describes the most common causes of plan failures. It offers a way as to avoid the disruption of plans and it shows that a plan is not just something that consumes time, but more a means of improving the process of creating a software product.