

# ČO ROBÍ ZLÝ PLÁN ZLÝM

*„Plán nie je dogma, je to nástroj!“*

*Peter Svorada*

Slovenská technická univerzita  
Fakulta informatiky a informačných technológií  
Ilkovičova 3, 842 16 Bratislava  
psvorada@gmail.com

**Abstrakt** *V dnešnom svete sú plány, ako súčasť bežného života vnímané negatívne. Častokrát nevychádzajú, a tak človek len ťažko hľadá motiváciu plánovať. S podobným fenoménom sa môžeme stretnúť aj pri vývoji softvérových systémov. Vývojári sa častokrát vyhýbajú plánovaniu pre zlé skúsenosti so zlými, nevydarenými plánmi. Avšak kde je hranica medzi vydareným a nevydareným plánom? Táto esej je v prvom rade zameraná na analýzu toho, čo robí zlý plán zlým, či sem nespádajú aj ľudia, ktorí vytvoria plán, rozhodnutí ho slepo nasledovať. Snažím sa určiť hranice medzi plánom ako dogmou, plánom ako nástrojom a jednoduchým odhadom. Zameriavam sa na pozitívna a negatívna jednotlivých pohľadov na funkciu plánovania. Esaj na záver rozoberá, či existujú postupy, ako vytvoriť plán, ktorý plne vyhovuje týmto jednotlivým pohľadom.*

**Kľúčové slová:** *plán, plánovanie, nástroj, dogma, chyby*

## Úvod

Nie dlho pred tým, ako som začal pracovať na tejto eseji, sa ma spolubývajúca spýtala, ako to bude s tenisovými raketami, ktoré som si zabudol doniesť pri sťahovaní do Bratislavy. Povedal som jej, že neviem či sa mi ich podarí priniesť autobusom a tak bude musieť pravdepodobne počkať do týždňa okolo 17. Novembra, kedy plánujem prísť do Bratislavy na aute, aby som sa mohol cestou späť do Lučenca zastaviť v Krupine, kde má kamarátka stužkovú. Spolubývajúca ostala v šoku. Keď som sa jej spýtal, čo sa jej nepáči, povedala, že ona takto dopredu nikdy neplánuje. Tak som sa jej spýtal, že koľko dopredu si teda plánuje. Odpoveď bola prinajmenšom smutná. Ona si vraj totižto neplánuje pre istotu vôbec. Pre istotu preto, lebo načo si bude ona niečo plánovať, keď jej žiadne plány aj tak nevychádzajú.

Najtragickejšie na tomto príbehu je, že toto nie je pohľad jedného nie práve najbystrejšieho dievčaťa a toto nie je ani pohľad skupiny stále neodrasených tínedžerov. Dnes si robí v živote plány skutočne málokto. Prečo aj?

Ono filozofia ľudí je skutočne taká, že sa párkrát v živote rozhodnú niečo veľmi divoko naplánovať, nasľubujú celému okoliu, že sa to všetko stihne, že každého stretnú, potom príde na lámanie chleba a „ups“. Zrazu nič nevychádza, nič sa nestíha a všetci vieme, kde nastal problém. Človek plánoval! Fuj! Hlupák, čo nevie, že plánovanie je na nič.

Ale je to skutočne tak? Nastala chyba naozaj v momente, keď človek začal plánovať? Nenastala náhodou v procese toho plánovania? Nemohol si vymyslieť lepší plán? Plán, ktorý sa dá aj stihnúť? Mohol. Nenastala chyba aj pri plnení toho plánu? Chcel ho človek pri jeho zostavovaní aj skutočne naplniť? No zrejme nie.

Všetko sa to ale dá robiť aj inak. Skutočne existujú aj ľudia, čo si niečo naplánujú a potom to aj stihnú. A skutočne im to vychádza s relatívne pravidelnou platnosťou.

Kde je ale potom rozdiel medzi tými dobrými a tými zlými? Čo robí ten dobrý tak dobre? Iná otázka. Čo robí ten zlý tak zle? Potrebujeme aby bol plán skutočne dobrý na to aby bol dobrý? Čo je jeho skutočný účel? Skutočne má na minútu presne odhadnúť každú našu ďalšiu činnosť?

Postavili sme si mnoho otázok. Najvyšší čas začať hľadať odpovede.

## **(Ne)robia to všetci**

V úvode sme sa pozreli na to, prečo ľudia vo svojich životoch neplánujú. Žiaľ, tieto ich skeptické pohľady na plán, ako nástroj, sú častokrát prenesené aj do ich profesionálneho života. Kým pri rekonštrukcii nášho domu, prišiel majster na to, že je niekde problém, mali sme zbúranú už polku nosnej steny. A samozrejme celá rekonštrukcia trvala v porovnaní s prvým odhadom tri krát toľko. Nikto neplánoval, veď načo.

Oblasť informatiky a softvérových projektov v tejto problematike samozrejme nezaostáva. Je síce pravda, že vysokoškolsky vzdelaný informatik strávil v škole trošku viac času, ako ten majster, čiže jeho o tom plánovaní aj učili. Že taký plán, to vie byť miestami dosť fajn záležitosť. On dokonca vyjde z tej školy a párkrát si plánovať v praxi aj skúsi. Vo veľkej časti prípadov je však dosť mimo a tak zisťuje, že plánovanie mu bude na dve veci. Už nedostane body za to, že stráví čas sedením nad Microsoft Project a aj tak mu to nevychádza, tak načo? Bude si robiť tak, ako mu to príde.

Jedného pekného dňa sa narodil človek menom Steve McConnell. Ak si človek zadá toto meno do vyhľadávača pochopí, že dotyčný pán nebude zrejme nejaké hocijaké béčko. A ak si človek prelistuje v tom vyhľadávači trošku viac zistí, že dotyčný pán nenapísal len množstvo rozsiahlych kníh, ale aj jednu krátku, trojstranovú, esej s názvom *The Nine Deadly Sins of Project Planning* [2], pojednávajúcu o chybách, ktoré ľudia robia pri tvorbe plánov. Ak sme sa v úvode pýtali, čo robia ľudia zle, keď vyrábajú plány, ktoré „nefungujú“, toto bude zrejme dobré miesto, kde začať hľadať odpovede. Poďme sa teda spoločne pozrieť na chyby, ktoré pán McConnell popisuje a spoločne si ich rozoberme.

## Deväť chýb plánovania pri projektoch [2]

### Neplánovať

Spomenul som už moju kamarátku, ktorá neplánuje, lebo plánom neverí. Spomenul som majstra stavby, ktorý neplánuje, lebo mu to „netreba“. A spomenul som nášho čerstvého absolventa vysokej školy informatiky, ktorý sa po pár neúspešných pokusoch rozhodne na plánovanie vykašľať. Všetci títo traja ľudia sa dopustili už prvej chyby pri plánovaní, ktorú spomína vo svojej práci Steve McConnell a tou je neplánovať vôbec.

McConnell zachádza v príklade dokonca tak ďaleko, že hovorí, že každý amatér je schopný pri zvážení všetkých potrieb svojho projektu zostaviť lepší plán ako projektový manažér, ktorý nezoberie do úvahy všetky fakty. S touto časťou sa však neviem úplne stotožniť. Myslím si, že amatér, ktorý o svojom projekte vie veľa a profesionálny projektový manažér, ktorý o projekte vie menej, môžu zostaviť kvalitatívne rovnaké plány. Kvalitatívne rovnako zlé. Len každý s chybami niekde inde.

### Neohľad na všetky časti projektu

Druhou chybou, ktorá je v eseji uvedená, je nebratie ohľadu na všetky časti projektu, akými sú konverzia dát z predchádzajúcich verzií programu, testovanie kompatibility a podobne. Práve na testovaní zvyknú manažéri plánovania ubrať najviac času, predpokladajúc, že programátor má pracovať stopercentne. Toto je podľa mňa najväčšou chybou väčšiny zlých plánov. Každý sme nejaký programátor, niekto lepší, niekto horší. Neexistuje však svet, v ktorom by programátor dokázal myslieť na všetky možné a nemožné diery. Osobne s väčšími softvérovými projektmi len začínam (ak sa dá tímový projekt za väčší softvérový projekt označiť), ale už teraz viem, že nech akokoľvek skúšam správnosť kódu v iteráciách jeho písania, vždy sa nakoniec nájde, ako by povedal jeden z našich vyučujúcich, chrobák, ktorého príčina sa hľadá a odstraňuje častokrát dlhšie, ako sa písal samotný kód. Tento fakt treba samozrejme do plánu zarátavať, no kto to v skutočnosti robí?

### Nezarátanie rizík

Treťou spomenutou chybou, je nezarátanie rizík do plánu. Riziká sú samozrejme hlavnou témou iných esejí, iných ľudí na tomto predmete, no nemôžem sa pri nich nepozastaviť. Riziko, často krát v množnom čísle, je neodlúčiteľnou súčasťou každého projektu. Pri vývoji softvérových projektov môžeme veľmi zhruba rozdeliť riziká na malé, väčšie a fatálne. Čo sa fatálnych rizík týka, tie do plánu zahrnieme len veľmi ťažko. Ak by sa Mária s Tónom (najväčšie hlavy nášho školského tímového projektu) večer na internáte opili a vypadli von z okna dvanásteho poschodia, mohol by som ako manažér plánovania, plánovať akokoľvek, z niečoho takéhoto by som jednoducho nevykorčuľoval. Ak by vypadli len z druhého poschodia a Mária by si zlomil ľavú ruku, Tóno pravú, tu je už priestor na úpravu plánu, netreba ho pokrčiť a zahodiť, len prispôbiť. Nuž a ak by sa len opili a z toho okna nevypadli, znamenalo by to len toľko, že na ďalší deň ich pracovný výkon bude jemne pokrívkať. S takýmto rizikom musí plán počítavať už dopredu. Čiže aj

bez úprav, by mal byť schopný takéto malé sklzy spôsobené nepredvídateľnými okolnosťami pokryť.

Samozrejme problém môže nastať, ak sa k programátorovi dostane plán so zarátanými rizikami a pracuje voľnejšie, veď má čas. Potom príde na záver jeho úlohy nejaký problém, s ktorým nepočítal a je opäť raz mimo plán.

Ako sa dá postaviť k tomuto problému rieši Phillip G. Armour vo svojej práci *To Plan, Two Plans* [1]. Rieši ho spôsobom, ktorý sa mne veľmi zapáčil, preto sem toto riešenie v krátkosti zhrniem.

Základom je vytvorenie dvoch plánov. Prvý plán je určený pre vývojárov, nezahŕňa žiadne riziká, vychádza z potrebného času, ktorý sa nejakým spôsobom odhadol, na dokončenie tej ktorej časti projektu. Druhý plán je vlastne prvý plán avšak so zarátaním rizík. Tento plán je určený pre koncového zákazníka, aby mal predstavu o časovom horizonte projektu. Takýmto spôsobom sa zabezpečí to, že vývojári pracujú naplno a aj zákazník je spokojný, lebo ak aj pri vývoji nastali nejaké menšie problémy, s tými sa v pláne, ktorý bol jemu predstretý, počítalo.

### **Rovnaký plán pre všetky projekty**

Najhoršia vec, ktorá sa môže človeku stať po dokončení projektu, ktorý sa stihol podľa plánu, je to, že sa tento plán označí za perfektný, nebudaj dokonca univerzálny. Nemyslím si, že existuje niečo ako perfektný plán pre hocijaký projekt, tobôž nie plán, ktorý by sa mohol označiť za univerzálny.

Súhlasím s pánom McConellom, že isté časti plánu sa dajú opätovane použiť, nesúhlasím však, že by sa takýto postup mal používať (ani pri projektoch, ktoré sú si podobné). Myslím si, že ak sa má niečo z projektu, pri ktorom nejaký plán fungoval, zobrať, tak to sú poznatky zo samotných procesov (dĺžka ich reálneho trvania a podobne), nie plán, alebo jeho časti samotné. Takto získané poznatky zakomponované do nového plánu (pokojne vytvoreného na rovnakých princípoch, ako ten dokončený, fungujúci) sú podľa mňa omnoho užitočnejšie, ako plán vychádzajúci z plánu.

### **Použitie cudzích plánov, návodov bez prispôsobenia**

Pri písaní tejto eseje som narazil na množstvo prác popisujúcich vhodný spôsob, akým vytvoriť „presný“ plán. Ich bezhlavým aplikovaním však cesta k úspechu nevedie. Nemôžem nič iné, ako súhlasiť s pánom McConellom, ktorý zdôrazňuje potrebu prispôsobovania plánu charakteristikám vlastným len pre náš projekt. Treba zvoliť také prístupy, ktoré sú pre projekt najvhodnejšie, nie ktoré vyzerajú najvhodnejšie, keď čítame, aké úžasne presné sú.

### **Plán vzdalujúci sa od reality**

Ďalším problémom, ktorý pán McConell uvádza, je dovoľenie plánu vzdalovať sa od reality. Budem opäť stručnejší, nakoľko súhlasím s tým, že vyrobiť plán na základe nejakej šablóny na úvod projektu a potom čakať, že sa nakoniec splní, je hlúposť. Plán musí byť dynamická entita, prispôbovaná v čase vzniknutými situáciami. Ako som už pri riešení rizík spomínal, ak sa vyskytne problém väčších rozmerov, človek nesmie ostať sedieť

a čakať, že sa plán splní, aj keby traktory padali. Nesplní. Treba ho prispôbiť tak, aby zodpovedal aktuálnej situácii a treba ho prispôbovať tak často, ako to je potrebné.

### Plánovanie detailov vzdialenej budúcnosti

Minulú sobotu som stretol skupinu školákov na stanici vo Zvolene. Nemohli mať viac ako štrnásť rokov. Boli medzi nimi aj dve slečny, ktoré sa rozprávali o plánoch do života. Jedna rozprávala druhej o tom, ako bude mať prvé dieťa, keď bude mať dvadsaťjeden. Druhé, keď bude mať dvadsaťštyri. A rozvedie sa, keď bude mať dvadsaťsedem. Vtedy som si už šľahol, na počudovanie okolia, pomerne hlasne rukou na tvár.

Áno, pán McConnell má pravdu, keď hovorí, že prílišné podrobnosti plánu v ďalekej budúcnosti projektu vedú len k problémom. Je to jednoduché, softvérový projekt je žijúca vec, pri ktorej sa vytvárajú nové rozhodnutia každú chvíľu, kde sa musia meniť postupy na základe vzniknutých situácií, ktoré sa dajú častokrát len veľmi ťažko predpokladať. Stráviť teda dva týždne nad zostavovaním na minúty rozpracovaného plánu šesť mesiacov dopredu, je hlúposť. Taký plán je buď predurčený na zmeny každú chvíľu, alebo vývojári sú predurčení ku práci spôsobom, ktorý nedáva veľmi zmysel.

Druhá vec, ktorú si dovoľím pridať ja a pre ktorú som uviedol ten príklad s deťmi zo stanice, je to, že ak máme v pláne vytýčené nejaké body, musíme vedieť, ako sa k nim prepracovať. A je jedno, ako veľmi sú vzdialené. Čiže vyhnúť sa podrobnostiam a detailom? Áno. Plánovať vzdialené ciele len od oka? Určite nie.

### Plánovanie dobiehania

Pri čítaní eseje pána McConella ma zaujala táto časť zo všetkých najviac. Píše v nej, že vývoj projektu sa v čase nezrýchľuje, naopak spomaľuje. Odôvodnené to bolo logicky. Ak vznikajú pri začiatku projektu chyby z dôvodu nepoznania technológií, tie sa po lepšom oboznámení začnú prejavovať a tak ich treba odstrániť, čo celý proces spomalí. Čiže sľúbiť si, že teraz, keď vieme ako, pôjde všetko rýchlejšie, je veľmi nepresné. Zaujalo ma to hlavne preto, že som bol z tých, ktorí si hovorili, že dobiehať stratený čas problém nie je a tento môj názor na vec sa zmenil.

### Neučenie sa z vlastných chýb

Relatívne najfatálnejšia chyba pri plánovaní, je neučenie sa z vlastných chýb. Myslím si to preto, lebo je naviazaná na všetky predchádzajúce. Od druhej po predposlednú, lebo sme si ich neuvedomili, odmietli uvedomiť. A na prvú, lebo v konečnom dôsledku ak sa na niekoľkokrát nepoučíme, vedie táto chyba práve k tej prvej, konečnej a tou je neplánovať vôbec.

### Desiate prikázanie: Pre plán nezabiješ

V predchádzajúcej časti sme si prešli deväť problémov zlých plánov tak, ako ich vymenoval Seve McConnell a tak, ako ich vidím ja. Znamená to teda, že ak sa týmto zlým „návykom“ vyhneme, zostavíme dobrý plán? Je to v každom prípade pravdepodobnejšie, ako s nimi. Znamená to však, že ak bez týchto zlých „návykov“ zostavíme plán, ktorý nevyjde, že bol ten plán aj tak zlý?

Tu narážame na pre mňa najzásadnejší problém všetkých plánujúcich ľudí. Problém nespomenutý v *The Nine Deadly Sins of Project Planning*. Problém pohľadu ľudí na plán ako taký.

Plán nemusí byť vôbec zlý, procesy vykonané počas jeho plnenia nemusia byť vôbec zlé, napriek tomu sa ho nepodarí splniť a človek označí plán za zlý. Manažér vyhodí vývojový tím a sám sa zavesí za kravatu na luster. Ale prečo?

Určil som si tri možné základné pohľady človeka na plán.

- Plán ako dogma
- Plán ako odhad
- Plán ako nástroj

V ďalšom texte sa skúsím bližšie rozobrať, čo pod týmito pohľadmi myslím, ich pozitíva a negatíva.

### **Plán ako dogma**

Toto je podľa mňa typický pohľad človeka, pracujúceho v oblasti informačných technológií, na plán. Plán má byť čo najpresnejší, plán má mať veľkú granularitu, plán má byť splnený na čas.

Positíva sú jasné, tím pracuje v relatívne presnom tempe, tlačiaci sa do malých časových úsekov. V prípade, že sa niečo nestíha, prichádza sa na to skoro, je možné prideliť na riešenie problému viac zdrojov, prípadne zvýšiť intenzitu práce.

Positíva sú tak do oka bijúce, že je logické sa spýtať, kde má takýto pohľad na plán ako exaktný, prepracovaný rozvrh negatíva. Už na začiatku som napísal, že plán má byť v tomto pohľade čo najpresnejší. Kameňom úrazu je práve tvorba takéhoto presného plánu. Skutočne málokto, dokáže vytvoriť plán tak, že vie byť presný aj pri skutočne krátkodobých úlohách (16 hodín a menej). Častokrát sa na túto úlohu podujmú manažéri, ktorí dostatočne nevidia do jednotlivých, malých podproblémov a tak vznikajú plány, pri troche šťastia príliš voľné, avšak vo väčšine prípadov nestihnuteľné. Ak sa potom pozeráme na plán ako exaktnú dogmu, ktorá sa musí stíhať tak, ako bola napísaná, vedie to k zlyhaniu plánu.

### **Plán ako odhad**

Takýto plán sa vyznačuje len veľmi hrubými odhadmi času potrebného pre dokončenie jednotlivých úloh. Tieto úlohy sú spravidla veľkého rozsahu a obsahu.

Positívom takéhoto pohľadu na plánovanie je hlavne pokoj na duši. Samozrejme takýto plán nemôže nikdy sedieť, ale človek, ktorý ho vytváral to tak ani nikdy nemyslel. Ide viac o spísanie úloh, ktoré treba vykonať, pričom sa k nim dopíšu nejaké časy, na ktorých nikto nevisí.

Toto je však pohľad toho nášho majstra stavby, ktorý mal prerábať dom dva mesiace a trvalo mu to pol roka a toto nesmie byť pohľad seriózneho softvérového inžiniera. Voľnosť áno, ale musí mať svoje hranice.

### Plán ako nástroj

Mne najbližší pohľad na plán, je keď ho beriem ako nástroj. Nástroj na určenie približného času potrebného na riešenie jednotlivých podproblémov, nástroj na efektívne rozdelenie všetkých zdrojov v danom čase a nástroj, ktorý je návodom ako sa dostať k cieľu.

Tento pohľad je niečím medzi dogmou a odhadom. Úlohy môžu byť aj väčšieho rozsahu, ale je kladený dôraz na to aby bol ich obsah, rozdelený do vhodných celkov. V prípade, že sa plán nedarí plniť, zvaží sa, či je vhodnejšie presunúť nejaké zdroje na tú ktorú úlohu, alebo sa upraví plán.

Pozitívom takéhoto plánu je, že dokáže byť relatívne presný a pri dostatočnom sledovaní jednotlivých úloh vedie k dobrým výsledkom. Ľudia nie sú pod extrémnym tlakom stíhať úlohy na hodiny, ale zároveň majú určené nie vzdialené limity tak, aby sa úlohe venovali.

Negatívom takéhoto plánu je, že dokáže byť relatívne presný. Áno, je to tu napísané opäť. Pri relatívne presnom pláne je pravdepodobné, že sa vyskytne problém, ktorý spôsobí posunutie celého procesu (ukáže sa, že úloha, na ktorú mal človek týždeň, má za sebou tak dve hodiny roboty) Tento problém sa dá však odstrániť vhodnou kontrolou progresu jednotlivých vývojárov počas vykonávania im pridenej úlohy.

Druhá vec je, že pri tomto pohľade na plán, ak by k takémuto posunu aj došlo, nerobí to zlý plán zlým, len núti manažéra plánovania prispôbiť ostatné procesy zmenám, ktoré nastali.

### Ideálny prístup

Existuje teda nejaký ideálny prístup k tomu ako sa treba pozeráť na plán? Samozrejme. Avšak záleží od toho, akú úlohu práve riešime.

Ak sme majster stavby, alebo „nášročná“ slečna cestujúca do zahraničia, plán ako odhad nám stačí. A keď sa plán nevydarí naplniť, nesmieme si trhať vlasy. Veď tí ľudia, ktorým sme zbúrali dom, majú stále kde prespať a tí kamaráti, ktorým som sľúbila, že ich prídem niekedy pozrieť pochopia, že to „niekedy“ môže byť aj o mesiac.

Ak sme študent, alebo menšia firma, plán ako nástroj, je podľa mňa pre nás najsprávnejší prístup. Treba pozeráť na termíny, avšak netreba niekomu odrhnúť hlavu, znížiť plat, alebo ho vyhodiť, ak za 16 hodín nestihol to, čo mal. Rovnako tak netreba ani plán označiť za zlý, ak nám v ňom niečo nevychádza. Pri tomto pohľade je plán nástroj, návod na dosiahnutie istého cieľa. Keď montujeme skriňu a ostanú nám vždy aspoň dve nepoužitú skrutky neznamená, že skriňa nestojí.

Osobne som toho názoru, že plán ako dogma je použiteľný úplne všade, avšak nevedie k dobrým výsledkom. Neznamená to, že plán bol postavený zle, len ak sa naňho pozeráme ako na exaktný rozvrh, je odsúdený na to aby bol na konci označený za zlý. Napriek tomu sa dajú pozitíva tohto pohľadu využiť inde.

## Perfektný plán

Nie raz som v tejto práci napísal, že niečo ako perfektný plán neexistuje. Stojím si za tým. Avšak dovoľm si postaviť otázku nasledovne. Je možné zostaviť plán tak, aby mohol byť na konci považovaný za dobrý?

Myslím si, že áno. Na to, aby sme k niečomu takému dospeli, musíme sa vyvarovať chýb, ktoré tu boli spomenuté a spojiť metodiku pohľadu na plán ako dogmu s filozofiou pohľadu na plán ako nástroj. Vysvetlím.

Dôležité je si uvedomiť, že plán má byť nástroj na dosiahnutie cieľa, nie nejaká dogma, ktorú keď sa nám nedarí naplňať, znamená koniec sveta. Neznamená to, že plán je zlý. Znamená to, že niekde nastal problém s ktorými treba do budúcnosti počítať a prispôbiť sa mu. To je filozofia pohľadu na plán ako nástroj.

Metodika pohľadu na plán ako dogmu je práve to delenie problémov na minimálne podproblémy. Nesúhlasím s názorom Joela Spolskeho v práci Evidance Based Scheduling [3], že ak sa úlohy nerozčlenia na časti trvajúce maximálne 16 hodín, že je niekde problém. Myslím si však, že ideálny plán by mal k tomu spieť a mal sa o to snažiť. Potom je možné vychádzať z metodiky uvedenej práve v jeho práci. Je vysoko pravdepodobné, že ak sa podarí rozčleniť čo najväčšie množstvo úloh, na malé problémy, budú sa tieto problémy v jednotlivých úlohách počas vykonávania projektu opakovať. A teda ak raz niečo stihnem dva krát za tri hodiny a mal som na to jednu, je možné podobné úlohy v pláne upraviť a rátať s tým, že na ne treba hodiny tri a nie jednu.

Ak k tomu ešte pridáme spomínanú teóriu vytvorenia dvoch plánov, takýmto prístupom sa podľa mňa dá dospieť k plánu, ktorý bude podľa možnosti čo najpresnejší a s ktorým budeme v konečnom dôsledku spokojní.

## Záver

V tejto práci som sa nezameral na exaktné postupy tvorby perfektných plánov, nie sú tu uvedené žiadne vzorce pre výpočty času určeného pre jednotlivé úlohy. Práca sa snaží objasniť prečo vznikajú zlé plány, prečo a ako sa môžu stať dobré plány zlými.

Vytvoriť dobrý plán neznamená vytvoriť presné časové úseky, do ktorých sa snažíme za každú cenu natlačiť. Vytvoriť dobrý plán znamená mať nástroj na dosiahnutie cieľa a vedieť koľko a akých zdrojov pri tom bude potrebné využiť.

Neopisoval som tu zákonite princípy viazané len na softvérový projekt. Tak ako zasiahli sklamaná z plánovania v bežnom živote profesionálny svet, môže to byť aj naopak. Je najvyšší čas...

## Použitá literatúra

1. Armour, G.P.: The business of software: *To plan, two plans*. Communications of the ACM, Vol. 48, No. 9 (2005) 15-19.



2. McConnel, S.: *The Nine Deadly Sins of Project Planning*. IEEE Software, Vol. 18, Issue 5, 2001
3. Spolsky, J: *Evidence Based Scheduling*. [www.joelonsoftware.com/items/2007/10/26.html](http://www.joelonsoftware.com/items/2007/10/26.html) [cit. 15.10.2010]
4. Spolsky,J: *Painless Software Schedules*. [www.joelonsoftware.com/articles/fog0000000245.html](http://www.joelonsoftware.com/articles/fog0000000245.html) [cit. 15.10.2010]

## Annotation

### *What makes a bad plan bad*

*In today's world plans , as part of everyday life, are viewed negatively. Many times they don't work out and so it is hard to find a motivation to plan. We can see similar phenomenon in development of software systems. Developers often avoid planning because of experience with bad plans that went wrong. But where is the line between good and bad plan? First of all this essay concerns what makes a bad plan bad, whether it is not really fault of persons who created the plan and blindly followed it. I try to find differences between plan viewed as a dogma, plan viewed as evaluation and a plan viewed as a tool. I concentrate on positives and negatives of these views. In the end essay discusses whether there are mechanics to create a plan, that would be suitable for these views.*