

PLÁNOVANIE V ŠKOLSKOM PROJEKTE TVORENÉHO JEDNOTLIVCOM

*Či plánujem? Som predsa študent, takže nie, aj keď
viem, že by som mal.*

Michal Lohnický

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

Abstrakt. *Význam plánovania v softvérových projektoch je pomerne jednoznačný. Plánovať má určite zmysel, či už ide o malé alebo veľké tímy tvoriace softvérový produkt. Dokonca aj význam plánovania v školských projektoch na úrovni tímových projektov je veľmi dôležitý a očividný. Ale ako je to so školskými softvérovými projektmi, ktoré vytvára iba jeden študent? Myslím si, že plánovanie má význam aj v takomto špecifickom projekte s relatívne malým rozsahom. Ponúkam dôvody, prečo by mal študent vo svojich projektoch plánovať, pričom niektoré sa môžu zdať triviálne, ale už len ich uvedomenie zohráva významnú úlohu v procese tvorby projektu. Ukazujem, že posunutie od intuitívneho plánovania školských projektov k reálnemu plánovaniu nemusí byť veľký skok, ale na druhej strane môže mať veľký prínos pre študenta.*

Kľúčové slová: *plánovanie, školské projekty, vodopádový model, extrémne programovanie*

Čo je plánovanie

Plánovanie je jeden z prvotných procesov, ktorý vykonáva projektový manažment. Je to proces, ktorý nielen že má za úlohu nastoliť poriadok v nasledujúcich etapách vývoja softvérového projektu, ale taktiež je prostriedkom pre manažéra, prostredníctvom ktorého pochopí celkovú komplexnosť projektu. Ústrednými cieľmi plánovania je redukovať

neurčitost' projektu, zlepšiť výkonnosť pracovníkov, zabezpečiť lepšie pochopenie biznis cieľov [1]. Plánovanie je taktiež základňa pre sledovanie a riadenie vykonanej práce, čo sú životne dôležité procesy pre manažment. Na to, aby vznikol dobrý plán, jeho tvorca musí detailne pochopiť požiadavky projektu na zdroje, prácu, kvalitu a kvantitu práce, pričom nesmie zabudnúť, že všetky tieto požiadavky striktno podliehajú požiadavkám zákazníka. Samozrejme nesmieme zabúdať, že plán neslúži len projektovému tímu, ale taktiež dáva zákazníkovi ucelený pohľad na vytváraný projekt.

Potreba plánovania v školskom projekte

Potreba plánovania v softvérovom projekte je nepochybne veľmi veľká, to mi dá za pravdu asi každý. Takmer žiadny väčší projekt sa nezaobíde bez plánovania, pokiaľ chceme hospodáriť so zdrojmi efektívne. Avšak na druhej strane je veľký rozdiel medzi plánovaním vo veľmi rozsiahlych projektoch, v rozsiahlych projektoch, v malých projektoch a vo veľmi malých projektoch. Ja sa zameriavam na spodnú hranicu rozsiahlosti, kedy je plánovanie ešte potrebné.

Plánovanie určite nemá zmysel, keď máme triviálny projekt, ktorý je realizovateľný vo veľmi krátkom časovom intervale. Je to len plytvanie zdrojmi. Ale kde je tá hranica? Respektíve, je tá hranica nad úrovňou alebo pod úrovňou rozsiahlosti školského projektu? Školské projekty majú rozdielny rozsah. Aj vzhľadom na to si myslím, že tá hranica, kedy je plánovanie ešte potrebné, sa nachádza práve na úrovni týchto projektov. Myslím si však, že aj napriek malej rozsiahlosti projektov je plánovanie v týchto projektoch nesmierne dôležité.

Skutočnosť je taká, že väčšinou študenti vo svojich projektoch neplánujú a ak plánujú, tak iba preto, lebo sú k tomu prinútení školou (mám na mysli formálne plánovanie so všetkými náležitosťami). Študent častokrát nemá ani motiváciu plánovať, pretože v školských projektoch nemá vidinu finančného zisku. Ziskom v tomto prípade môže byť známka, ale väčšina študentov sa uspokojí so známkou E, najnižšou možnou na prejedenie školy. Ďalším ziskom sú nadobudnuté vedomosti. Ale čo ak študent vedomosti z daného predmetu nepokladá za dôležité? Tak prečo by študenti mali vynakladať nadbytočné úsilie ešte na plánovanie? Treba si dôkladne uvedomiť, čo môže študent plánovaním získať. E je tiež známka a na jej dosiahnutie treba vykonať netriviálne úsilie. E potrebuje študent minimálne na to, aby mohol pokračovať, aj keď ho nadobudnuté vedomosti nezaujímajú.

Softvéroví inžinieri veľmi rýchlo pochopili potrebu plánovania vo svojich projektoch. Myslím, že pohnútky pre plánovanie v školských projektoch by mali byť podobné, pričom najväčším ziskom pre študenta môže byť ušetrený čas pri realizácii školského projektu (samozrejme okrem iných výhod).

Dôvody pre plánovanie [1] a ich význam v školskom projekte sú nasledovné:

- *Redukovať neurčitost' (výsledku projektu)* – Pre študenta je dôležité, aby si najprv ujasnil, aký projekt ide robiť, akú bude mať funkcionálnosť a ako sa projekt bude správať ako celok.
- *Zlepšiť výkonnosť* – Podrobným rozpracovaním projektu a následným časovým rozvrhnutím si vie študent lepšie rozložiť svoj čas a vie ho následne efektívnejšie využiť.

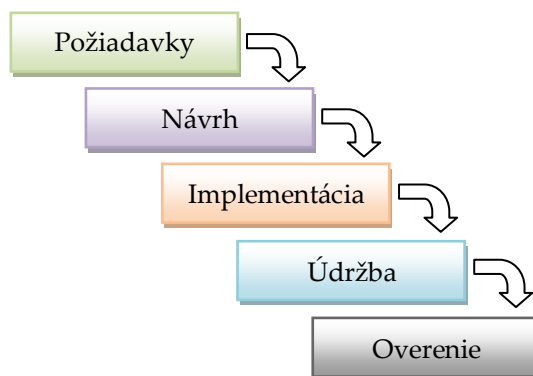
- *Zabezpečiť lepšie porozumenie cieľom projektu* – Vo väčšine prípadov pokiaľ študent dosiahne stanovené ciele školského projektu, naučí sa učivo, ktoré je od neho požadované. Preto je obzvlášť dôležité, aby už na začiatku správne pochopil ciele školského projektu a následne ich dosiahol.
- *Vytvoriť základňu pre sledovanie a riadenie práce* – Tento bod je pre študenta menej dôležitý. Študent ako jednočlenný tím vytvárajúci projekt v krátkom časovom intervale, nepotrebuje mať takú rozsiahlu základňu pre sledovanie, pretože projekt si sám navrhne a sám ho implementuje, takže je dostatočne oboznámený so svojou prácou. Na druhej strane je dobré, keď sa študent drží zamýšľaného plánu a neodbočuje od podstaty projektu a od cieľov projektu.

Ako plánovať v školskom projekte?

Na to, aby študent vedel dobre naplánovať svoj projekt, mal by dostatočne poznať svoju metodiku vývoja projektu a podľa nej prispôbiť svoju tvorbu plánov. Metodológií je pomerne veľa, ale nie všetky sa hodia na tvorbu školských projektov. Ponúkam prehľad dvoch pomerne rozličných metód a ich vplyv na tvorbu plánov v školskom projekte.

Rokmi overená vodopádová metodológia vývoja

Jednou z najznámejších a najpoužívanejších metodík vývoja je vodopádový model. Je to sekvenčná metodika vývoja, kedy sa najprv dokončí jedna fáza vývoja a následne sa pokračuje na ďalšiu (pozri Obr. 1) [3].



Obr. 1. Schéma vodopádového modelu vývoja.

Pri vodopádovom modeli vývoja je väčšina zodpovednosti delegovaná na jedného človeka, ktorý sa špecializuje na tvorbu plánov a odhadov alebo projektový plán vytvára tím špecializovaných ľudí. Takáto špecializácia je vynútená samotným modelom, pretože v začiatkových fázach vývoja sa vytvorí skoro kompletný plán, podľa ktorého sa vyvíja projekt, a teda plánu sa venuje pomerne veľké úsilie. Zmeny požiadaviek väčšinou znamenajú pomerne veľké vynaloženie nákladov, pretože plán je fixný a podlieha mu celý vývoj od začiatku do konca. Čím dlhší je proces vývoja, o to väčšie riziká prináša takýto vývoj a plánovanie, pretože plán musí čeliť nepredvídateľným komplikáciám a taktiež je

obťažne na začiatku navrhnuť kompletný produkt podľa predstáv zákazníka. Koniec koncov ako sa hovorí: „Najprv to musím vidieť!“.

Výhodou takéhoto plánovania sú jasne definované postupy a ohraničenia, presne definované náklady, čas dokončenia, a teda projektový tím vie s čím môže disponovať a s čím nie.

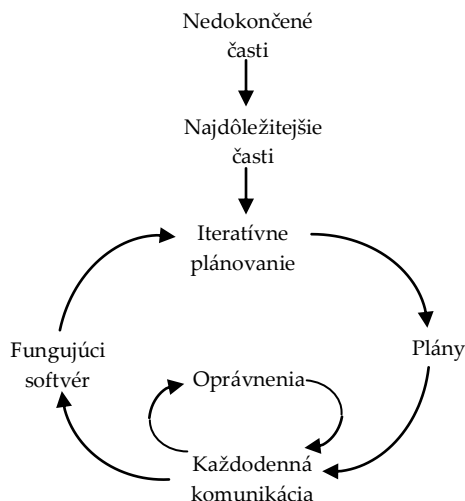
Z pohľadu študenta

Vodopádový model vývoja je vo veľkej miere zastúpený v radoch študentov. Požiadavky sú dané samotnými vyučujúcimi, návrh je intuitívne vytvorený v hlave študenta, následná implementácia je urobená podľa možností študenta, verifikáciu v lepších prípadoch vykoná s častí študent, a potom finálnu verifikáciu vykoná profesor alebo vyučujúci. Pokiaľ študent neopakuje predmet, tak údržba nie je vôbec potrebná. Obávam sa ale, že tento prístup nie je správny. Nie je tu zachytený konzultačný proces a následné zapracovanie pripomienok do projektu.

S plánovaním je to obťažne. Vo väčšine prípadov študent nevie dostatočne predvídať svoje schopnosti, a preto nevie v jednom okamihu na začiatku dôkladne naplánovať celý postup vývoja. Ďalšou nevýhodou je, že keď sa implementujú nové poznatky, študent niekedy nevie odhadnúť, čo je od neho požadované a projekt sa musí prerábať.

Extrémne programovanie

Extrémne programovanie je pomerne mladá metodika agilného typu vývoja softvéru. V prvom rade sa zameriava na otvorenosť vývoja softvéru voči zmenám požiadaviek zákazníka a tým predpokladá vytvorenie kvalitnejšieho produktu. Smerovanie projektu je veľmi často overované ukázkami priebežných verzii zákazníkovi (pozri Obr. 2) [6].



Obr. 2. Schéma vývoja v extrémnom programovaní.

Plánovanie všeobecne v agilných metodikách vývoja prebieha v prvej fáze iba vágne, aby sa stanovili základné črty projektu (aký bude rozsah projektu, aké budú náklady na

realizáciu a koľko to bude stáť zákazníka a iné rozvrhnutie zdrojov). Následne sa časovanie projektu rozdelí do niekoľkých fáz, kde každá fáza trvá približne 1-6 týždňov. Každá fáza je následne podrobnejšie rozplánovaná podobne ako pri vodopádovej metodológii vývoja. V každej fáze je veľmi dôležité naplánovať zapracovanie jednotlivých požiadaviek. V niektorých tímoch sa na zoradenie požiadaviek používa fibonačoho postupnosť, kedy sa na označenie priority použije výhradne hodnota z fibonačoho postupnosti. To zaručuje správne rozloženie priorít.

Z pohľadu študenta

Na Universität Karlsruhe pán Müller a pán Tichy urobili pokus, kedy prinútili študentov využívať výhradne extrémne programovania ako vývojovú metodológiu. Narazili na určité komplikácie, pretože študenti sa nevedeli sústrediť na výhradné plánovanie zapracovania požiadaviek. Stále sa snažili plánovať viac do budúcnosti, čo zdržiavalo vývoj [**Chyba! Nenalezen zdroj odkazů.**].

Môj pohľad na využitie extrémneho programovania je trochu odlišný. Rozsah študentských projektov (na ktoré sa primárne zameriavam) je približne 2-3 týždne vývoja a následnej implementácie, pričom stretnutie s vyučujúcim je väčšinou raz za týždeň. Univerzitné prostredie je vynikajúce na učenie z viacerých uhlov. Môžeme tu aplikovať istý zmenšený model extrémneho programovania, kedy je vyučujúci v pozícii zákazníka a kladie požiadavky a na druhej strane je študent, ktorý konzultuje tieto požiadavky s vyučujúcim. V tomto prípade si môžeme predstaviť cvičenie respektíve konzultačnú hodinu ako ukážku verzie softvéru zákazníkovi, pričom vzniká priam učebnicová spätná väzba. Pozícia vyučujúceho a študenta je trochu odlišná ako v reálnom prostredí, ale má k nej veľmi blízko. Študent si tu taktiež overuje smerovanie projektu, aby v čo najväčšej miere uspokojil požiadavky zadávateľa. Nehovoriac o tom, že sa pri tom študent učí nové veci.

Celý vývoj takéhoto projektu je veľmi podobný s extrémnym programovaním a agilnými metódami vývoja, pretože študentovi v počiatku stačí menej rozsiahle plánovanie, aby vedel narábať so svojimi zdrojmi (hlavne čas, aby sa stihla vykonať finálna verzia do pevne stanoveného termínu). Po menej rozsiahlom globálnom plánovaní nasleduje podrobnejšie plánovanie medzi inkrementami respektíve konzultačnými hodinami.

Od „hurá systému“ k reálnemu plánovaniu

Študenti vo väčšine prípadov používajú takzvanú „hurá systém“ metodológiu, kedy si spravia v hlavách intuitívny návrh systému, a potom idú okamžite implementovať. Takýto prístup má viacero obmedzení, ktoré som už vyššie spomenul (neschopnosť odhadnúť svoje schopnosti, možnosti a zdroje). Presvedčiť študenta, aby začal plánovať nie je jednoduché, pretože plánovanie si vyžaduje väčšie úsilie v začiatkových fázach projektu. Väčšina študentov odkladá realizáciu projektu, čo najviac k termínu odovzdania. Preklenutím týchto „mentálnych“ bariér, ale študent môže získať. Využitím metodiky, ktorá je podobná k „hurá systému“, študent nemusí meniť vo veľkej miere svoje návyky. Naopak, dokáže dokonca ušetriť niektoré svoje zdroje (hlavne čas) a pritom sa viac naučí (respektíve možno využitie zdrojov zostane na rovnakej úrovni, ale naučí sa ich

efektívnejšie využívať). Z vybratej metodiky následne vyplýva spôsob plánovania, čo je už vlastne návod ako postupovať pri tvorbe a realizácii projektov.

Zatiaľ som sa venoval plánovaniu ako unifikovanému procesu, ktorý má študentovi uľahčiť tvorbu školských softvérových projektov. Avšak plánovanie nie je unifikovaný proces, ale má viacero častí a pohľadov, ktorými sa tvorca plánu musí zaoberať. Druhy plánov [1] a ich význam v školskom projekte sú nasledovné:

- *plánovanie rozsahu* (vychádza z biznis cieľov a zahŕňa rôzne alternatívy projektu) – Pre študenta je veľmi dôležité, pretože zahŕňa samotnú podstatu projektu (čo má urobiť, ako to má vyzeráť atď).
- *plánovanie činností* (aké činnosti sa budú v projekte vykonávať a kedy) – Pre študenta opäť veľmi dôležité, pretože plán konkrétne identifikuje, čo musí vykonať, aby vytvoril finálnu verziu projektu.
- *plánovanie zdrojov* (aké zdroje napr. ľudia, zariadenie, materiály by sa mali použiť na vykonanie činností) – Nie je veľmi podstatné, pretože v softvérovom projekte je prakticky jediným zdrojom študent sám (tu nerátam čas ako zdroj, pretože jeho použitie vyplýva z iných plánovaní).
- *plánovanie nákladov a rozpočtu* (náklady, ktoré treba na uskutočnenie a prevádzku činností) – Pri školských projektoch sa nedá hovoriť o financiách priamo súvisiacich s projektom. Do úvahy sa dajú zobrať financie, ktoré študent tratiť, keď pracuje na školskom projekte a teda nepracuje v práci, kde by mohol zarábať.
- *plánovanie manažmentu rizík* (definovanie krokov pri výskyte udalostí spôsobiacich škodu) – Najväčším rizikom pri školských projektoch je nepochopenie zadania a teda zlého interpretovania požiadaviek. Samozrejme na toto by mal byť študent v dostatočnej miere pripravený.
- *plánovanie akosti* (zabezpečuje dodržanie kvality produktu) – Úroveň plánovania akosti v školskom projekte je priamo úmerná známke, ktorú študent očakáva.
- *plánovanie komunikácie* (určuje kedy, kde a aká informácia je potrebná a ako sa k nej dá dostať) – Keďže študent plánuje a realizuje svoj projekt osobitne, tak zaoberať sa komunikáciou medzi členmi je irelevantné. Skôr by si mal študent naplánovať komunikáciu s vyučujúcim a prístup k informáciám (napr. obmedzený prístup do digitálnych knižníc iba z univerzitných počítačov).
- *plánovanie v obstarávaní* (určuje aké externé zdroje sú potrebné v projekte) – Vo väčšine prípadov nepodstatné pre študenta.

Ako vidieť, nie všetky aspekty plánovania sú pre študenta kľúčové, ale na druhej strane niektorými by sa mal vážne zaoberať. Možno to z tohto pohľadu vyzerá, že dobré plánovanie musí obsahovať veľa rôznych plánov, a teda veľa komplikácií, ale nie je to tak. Rôzne druhy plánovania umožňujú dekompozíciu problematiky na jednoduchšie problémy, a teda zjednodušuje proces plánovania.

Využívanie takéhoto prístupu pri tvorbe školských projektov pripravuje študenta na reálnu prácu v tíme. Študent sa učí odhadovať svoje možnosti, narába s ohraničeným časom, pracuje za odmenu (známka), učí sa komunikovať so zákazníkom (vyučujúci) atď. Výhodné na tom je, že toto všetko sa nedeje pod takým stresom ako keď ide o peniaze

a keď má veľkú zodpovednosť za svoje rozhodnutia. Ďalšou výhodou je, že realizácia projektov prebieha v ideálnom prostredí, ktorým je škola.

Pri procese plánovania je vysoká potreba využívať rôzne nástroje, ktoré uľahčia plánovanie, ale obávam sa, že chcieť od študenta používať nástroje na plánovanie, by bolo trochu naivné. V konečnom dôsledku to ani nie je potrebné pri plánovaní projektu s takýmto rozsahom a počtom zainteresovaných osôb. Študentovi stačí, aby projekt realizoval sekvenčne (z pohľadu vykonania činností), nepotrebuje plán na paralelnú tvorbu softvéru a teda nemusí využívať napríklad metódu kritickej cesty. Vie si jednoduchšie naplánovať jednotlivé mílniky, ktoré musí dosiahnuť a nerozdeľuje svoju prácu medzi viacero ľudí a teda nepotrebuje vytvárať diagram s rozdelením činností v čase (Ganttov diagram, PERT). I keď nad týmto by sa dalo tiež polemizovať, či by mal na rozvrh svojich činností používať podporný softvér. Myslím si, že v princípe nemusí, ale pri komplexnom plánovaní (nie len školský projekt, ale aj práca a iné mimoškolské aktivity) by to mohlo taktiež vytvoriť určité výhody a lepší prehľad.

Prečo neplánovať rovno vo veľkom

Zatiaľ som sa zameriaval na jednotlivé školské projekty. Ale prečo neísť za hranice takého plánovania. Inkrementálna metóda vývoja sa dá chápať aj v širších súvislostiach, kedy jednotlivé ukážky produktu sú realizáciou jednotlivých zadaní na rôznych predmetoch. Proces takéhoto plánovania by trval počas celého štúdia študenta. Úplne ideálne je, keď študent plánuje tak, že počas štúdia nemusí vykonávať žiadnu duplicitnú prácu. Študent by mal myslieť na to, že si môže prispôbovať jednotlivé školské projekty tak, aby znovu využil to, čo už naprogramoval v predchádzajúcich zadaniach. V samých začiatkoch štúdia by mal študent preštudovať aké projekty bude musieť v škole realizovať a podľa toho naplánovať a prispôbiť každý jednotlivý školský projekt. Takéto myslenie by ušetrilo veľa času študentovi a zvýšilo jeho efektivitu počas celého jeho štúdia.

Globálne plánovanie v tomto smere má určitú komplikáciu. Študent samozrejme pred začiatkom štúdia nemá predstavu ako sa správne plánuje. Riešenie tohto nie je netriviálne a nájsť ho nie je predmetom mojej eseje, ale verím, že existuje.

Záver

Určitý čas trvalo, kým plánovanie získalo v tvorbe projektu taký význam aký má. Neskôr si to uvedomil aj manažment v malých tímoch a teraz je na čase, aby si to uvedomili aj jednotlivci a hlavne študenti v súvislosti, nielen s ich školskými projektmi, ale aj v širšej súvislosti ich každodenných povinností. Aj keby plánovanie v živote študenta zostalo iba na úrovni plánovania jeho školského softvérového projektu, tak študent takýmto prístupom môže veľa získať. Aj keby plánovanie v určitom zmysle znamenalo stratu, respektíve plytvanie zdrojov, stále je to výborná investícia do budúcnosti, pretože sa naučí odhadovať svoje možnosti a zároveň si uvedomí svoju hodnotu na trhu. Okrem iného, vďaka plánovaniu jeho hodnota na trhu reálne stúpne, pretože si bude vedieť lepšie zorganizovať svoj čas a lepšie ho využiť.

Použitá literatúra

1. Bieliková, M.: *Softvérové inžinierstvo*. Bratislava, 2000.
2. Jurison, J.: Software Project Management: The Manager's View. In: *Communications of AIS*. Vol. 2, No. 17 (1999) 1-57.
3. Kolektív autorov: *Waterfall Model*. Dostupné na internete: http://en.wikipedia.org/wiki/Waterfall_model, [cit: 2009-Október]
4. Müller M., Tichy W.: *Case study: extreme programming in a university environment*. In ICSE '01: Proceedings of the 23rd International Conference on Software Engineering, pages 537–544, Washington, DC, USA, 2001.
5. Schwaber, K.: *SCRUM Development Process*. Dostupné na internete: <http://jeffsutherland.com/oops/schwapub.pdf>, [cit: 2009-Október]
6. Wells, D.: *Extreme programming*. Dostupné na internete: <http://www.extremeprogramming.org>, , [cit: 2009-Október]

Annotation

Planning in school projects made by an individual

The meaning of planning in software projects is rather straightforward. The planning has definitely its purpose in small teams, as well as in large ones dealing with software products. Even the meaning of planning in school projects on the level of team projects is very crucial and obvious. But how about school software projects made by one student only? I think that the planning has its purpose also in such a specific project with a relatively small extent. I offer some reasons why a student should plan in his/ her projects. Some of them might seem trivial but at least the realization of them plays a significant role in the process of project making. I point that in the movement from an intuitive planning of school projects to the real planning need not be any great effort. On the other hand, it may bring great benefit to students.