

# Úloha kvality v metodológiách vývoja

MARTIN KOVÁČIK

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

**Abstrakt.** Zabezpečenie kvality je určené dvoma aspektami. Prvý aspekt je poskytnutie vhodných techník, ktoré podporujú vytváranie vysoko kvalitného produktu a techník na zabezpečenie kvality už existujúcich častí. Oba prístupy musia byť skĺbené pri snahe o dosiahnutie efektívneho a kvalitného vývoja softvéru. Cieľom tohto dokumentu je všeobecné priblíženie metodológie a jej prepojenie so zabezpečením kvality.

## Úvod

Definícia kvality je komplexná. Neexistuje a absolútne určenie kvality ako veličiny. Kvalita nie je určená len vonkajšími charakteristikami produktu, ale aj internými znakmi, ktoré ostávajú pred zrakom klienta ukryté. Zameriavanie sa na externé znaky je dané snahou o maximalizáciu efektivity z časovo-finančného pohľadu. Z dlhodobého hľadiska môže zanedbávanie interných znakov negatívne odraziť na napredovaní projektu, či už finančne, alebo inými spôsobmi.

Zameranie sa na interné znaky kvality a ich snaha o ich zlepšenie avšak nemá význam, ak sa ich zlepšenie neprejaví aj navonok. Vonkajšie prejavy vychádzajúce z interných znakov kvality zahŕňajú znížené náklady na vývoj.

## Znaky kvality

V úvode som nadškrtol rozdelenie znakov na externé a interné. Potreba rozdelenia je daná rôznymi pohľadmi, ktoré vnímajú produkt a jeho kvalitatívne znaky.

Pri vývoji softvéru je vnímanie kvality diferencované na vnímanie kvality zákazníkom a interné vnímanie kvality tvorcami softvérového produktu. Zákazník sa zameriava na vonkajšie znaky. Vonkajšie znaky sú jasne definovateľné a sledovateľné. Ich splnenie je predpokladom úspešného dokončenia produktu.

Interné znaky sú vnímané vývojárom produktu ako aj ľuďmi, ktorí prichádzajú do kontaktu so systémom počas jeho vývoja. Interné znaky zahŕňajú statické prvky produktu, ako aj procesy, ktoré sa používajú pri tvorbe produktu. Je možné definovať

nasledujúce kritéria, ktoré sa hodnotia ako interné znaky kvality produktu a sú aj súčasťou vývojovej fázy.

- Statické znaky produktu. V prípade softvérového projektu ide o kvalitu návrhu, zdrojových kódov a dokumentácie k riešeniu.
- Prvky podporujúce odhaľovanie chýb a verifikáciu externých znakov kvality. Príkladom týchto prvkov sú testy produktu.
- Procesy potrebné pre vytvorenie produktu. Tieto procesy zahŕňajú výmenu informácií medzi participantami, ako aj interné procesy priamo zabezpečujúce tvorbu produktu.

Súhrn znakov je charakteristický pre produkt a metodológiu využívanú pri tvorbe produktu.

### **Metodológie vývoja softvéru a manažment kvality**

Metodológie vývoja softvéru priamo definujú interné znaky softvérového produktu. Tradičný model vývoja softvéru je tvorený základnými krokmi

- Špecifikácia a analýza
- Návrh
- Implementácia
- Testovanie

Tieto procesy zabezpečujú korektnú analýzu požiadaviek a správnu implementáciu. Avšak tieto procesy priamo nepodporujú kvalitatívne kritéria ako napr. použiteľnosť a bezpečnosť.

Niektoré procesné modely zahŕňajú aj aktivity, ktoré sa uskutočňujú po testoch a hodnoteniach kvality produktu. Tieto aktivity podporujú stabilitu nasadeného systému. Viaceré procesné modely umožňujú koexistenciu s riadiacimi aktivitami, ktoré sa zameriavajú na hodnotenie kvality s pohľadom zákazníka a finančno-časovými faktormi.

Napriek tomu väčšina metodológií vývoja softvéru nepodporuje priamo zabezpečenie kvality vývoja softvéru a zabezpečenie kvality samotného produktu. Procesy podporujúce integráciu zabezpečenia kvality pre procesnú aj produktovú časť projektu sú potrebné pre dosiahnutie globálneho kvalitatívneho ohodnotenia celého produktu.

Pri existujúcich metodológiách sú typické spôsoby na zabezpečenie kvality navrhnuté prevažne ako nádstavba s určitými ohraničeniami.

- Nádstavbové systémy zabezpečenia kvality by mali byť realizované samostatným oddelením na zabezpečenie kvality. Toto je vyžadované kvôli rozpočtovým a personálnym faktorom.

- Využívajú sa systémy na vyhľadávanie chýb a nedostatkov, ktoré už existujú v produkte. Tento spôsob je nákladný, keďže musí riešiť vzniknuté nedostatky bez toho, aby napomáhal prevencii.

Zavedenie zabezpečenia kvality vývoja smeruje k rovnako kvalitnému produktu, avšak umožňuje sa vyhnúť potrebe nádstavbového riešenia spolu so svojimi nedostatkami.

Cieľom zabezpečenia kvality produktu je zamedziť existencii chýb a nedostatkov v definovaných termínoch projektu. Využívanie nádstavbových techník na zabezpečenie kvality produktu je možné minimalizovať zavedením zabezpečenia kvality vývoja.

Zabezpečenie kvality vývoja softvéru je možné dosiahnuť nasledujúcimi prvkami:

- Testovanie
- Statická analýza
- Vývojové procesy

Integrácia týchto prvkov je najlepší spôsob zabezpečenia kvality, avšak nie je možné to dosiahnuť úplne.

Existuje viacero postupov na dosahovanie zvyšovania kvality:

- Lepšie spracovávanie ohodnotenia kvality
- Lepšie meranie kvality
- Kvalitnejšie procesy
- Kvalitnejšie nástroje na dosahovanie kvality

Na procesnej úrovni sú definované štandarty, ktoré smerujú k zvyšovaniu kvality. Príkladom takýchto štandardov sú nasledujúce:

- Total Quality Management (TQM)
- ISO 9000
- Capability Maturity Model (CMM)

Štandarty na procesnej úrovni sú v roli procesného dozoru. Uvedené štandarty overujú len to, či sú splnené predpoklady pre úspešnú realizáciu procesu.

### **Postupy na zabezpečenie kvality vývojových procesov a zabezpečenie kvality produktu**

Metodológie vývoja softvéru je potrebné integrovať s konkrétnymi postupmi na zabezpečenie kvality.

Prvky zabezpečenia kvality, ktoré je potrebné zakomponovať do súčasných metodológií sú

- Iteratívny spôsob tvorby softvérového produktu
- Stanovenie kvality projektu ako cieľa

- Priebežné vyhodnocovanie kvality
- Sledovanie požiadaviek zákazníka
- Orientácia sa na architektúru
- Orientácia metodológie na tímovu spoluprácu
- Konfigurácia a manažment zmien

Na dosiahnutie vyššej kvality je potrebné zavádzať *iteratívny a inkrementálny spôsob vývoja produktu*. Iteratívny spôsob vývoja je charakteristický postupným zjemňovaním činnosti a zkvalitňovaním produktu. Jednotlivé cykly v iterácií sa skladajú z kompletnej sady činností definovanej používanou metodológiou. Vo všeobecnosti ide o fázy analýzy, návrhu, implementácie, testovania a finálneho nasadenia. Zabezpečenie kvality môže byť uplatňované počas celého cyklu. Výhodou iteratívneho spôsobu vývoja je možnosť postupne zkvalitňovať procesy, ktoré prebiehajú v rámci iterácie.

Iteratívny prístup je vhodný pretože existujú spätná väzba, ktorá smeruje k opätovnému zvýšeniu kvality. Iteratívny spôsob vývoja musí byť zakomponovaný aj do procesu riadenia rizík v projekte. Vyžaduje sa aj oboznámenie zákazníka s týmto procesom.

*Kvalita ako cieľ projektu* musí byť zdokumentovaná zapojením zákazníka a tímu. Toto zaručuje, že miera kvality je dosiahnuteľná a merateľná.

*Priebežné vyhodnocovanie kvality* je dôležité aby bolo možné finálne zabezpečiť kvalitu produktu. Ide predovšetkým o získavanie aktuálneho statusu, ako aj pozorovanie aktuálnych aktivít a možných zmien, ktoré sú potrebné na skoré odhalenie problémov. Priebežné vyhodnocovanie zahŕňa testovanie systému. Okrem interných testov je potrebné verifikovať, či produkt spĺňa kvalitatívne požiadavky klienta.. Proces vývoja preto musí zahŕňať pracovný plán zahŕňajúci externé testy so zákazníkmi pre zaručenie vysokej kvality.

*Požiadavky zákazníka* je potrebné dôkladne zdokumentovať. Je to dôležité pre spätné overovanie kvality zo strany zákazníka. Problémom je netechnická povaha zákazníka. Preto je potrebné sledovať požiadavky zákazníka a spätne overovať ich pochopenie aby nedochádzalo k prípadným nedorozumeniam.

*Orientácia na architektúru* je dôležitá z pohľadu znovupoužiteľnosti. Toto ovplyvňuje časový a cenový aspekt realizácie projektu. Zvlášť dôležitá je architektúra pri integrácií produktu do existujúcich systémov a riešení.

*Tímová spolupráca* je zvlášť dôležitá pri zvyšovaní kvality. Tím je zodpovedný za kvalitu a úspech projektu. V prípade, že je zodpovednosť za kvalitu a úspech delegovaná na jedného človeka, nie je možné zabezpečiť úspech projektu. Zameranie sa na tímovú prácu zároveň zvyšuje motiváciu členov tímu, čo nepriamo smeruje k zvyšovaniu interných kvalitatívnych znakov.

*Riadenie zmien a konfigurácie* priamo súvisí s kvalitatívnym ohodnotením projektu. Zmeny v projekte zavádzajú zmeny aj v ohodnocovaní kvality. Preto je potrebné evidovať zmeny a vedieť ohodnotiť ich vplyv na kvalitu produktu a použitých procesov.

### **Priebežné vyhodnocovanie kvality**

Vyhodnocovanie je možné priebežným sledovaním stavu projektu. Vhodným spôsobom sledovania projektu je testovanie.

Testovanie je možné realizovať vo viacerých fázach projektu. Samotné testovanie prebieha súčasne spolu s vývojom. Je možné definovať testovací cyklus [2], ktorý pozostáva z nasledujúcich krokov:

- Špecifikácia
- Návrh. Tester by mali pracovať s vývojármi a zisťovať, ktoré časti návrhu sú testovateľné a podľa akých parametrov tieto testy fungujú.
- Plánovanie testov.
- Tvorba testov. Plánovanie testov zahŕňa aj vytváranie prostredia na testovanie, ktoré je možné využívať aj programátormi pri tvorbe produktu a sledovaní jeho vlastností.
- Vykonávanie testov. Výstupom vykonávania testov je zoznam nedostatkov a porušení kvality. Výstup je formou spätnej väzby predávaný vývojárom.
- Vyhodnocovanie testov. Cieľom je vytvorenie metrík, ktoré pomáhajú určiť, či je produkt v danej fáze pripravený na to aby splnil zákazníkové požiadavky a očakávania.

### **Testovacích plánov**

Testovací plán je systematický prístup na testovanie softvérového produktu. Cieľom je vytvorenie pracovného plánu na presné vykonávanie a vyhodnocovanie testu.

Základom testovacích plánov je definícia

- Rozsahu testovania
- Časového plánovania
- Ciele testovania
- Riziká. Určujú problematické časti, ktoré môžu nastať pri testovaní – ide napríklad o využívanie produktov tretích strán, nedostatočná dokumentácia a iné.
- Kritéria testov

Dôležitá časť testov sú kritéria. Kritéria definujú podmienky akceptácie testovaného produktu zákazníkom a tím priamo definujú kvalitatívne znaky.

### **Záver**

Zabezpečovanie kvality produktu je neoddeliteľnou súčasťou moderného vývoja softvéru. Je viacero možných spôsobov ako zvýšiť kvalitu vývojového procesu. Jedna

z alternatív je zavedenie tzv. „dozerajúceho štandardu“, ktorý sa tesne integruje s procesmi spojenými s vývojom.

Samotný vývojový proces je charakteristicky využívanými metodikami. Rozšírenia metodík nedostatočne integrujú vývojový proces so zabezpečením kvality. V mojej eseji som navrhol miesta v daných metodikách, ktoré je možné využiť na tesnejšiu integráciu zabezpečenia kvality a procesu vývoja softvérového produktu.

### **Použitá literatúra**

1. Zuser W., Heil S., Grechenig T.: Software Quality Development and Assurance in RUP, MSF and XP – A Comparative Study. *ACM SIGSOFT Software Engineering Notes, Proceedings of the third workshop on Software Qualit*, Vol. 30, Issue 4, (May 2005).
2. Software Testing, URL:[http://en.wikipedia.org/wiki/Software\\_testing](http://en.wikipedia.org/wiki/Software_testing), stiahnuté 25.10.2006

### **Annotation**

#### *Quality Assurance in Development Process*

The support of software quality in a software development process may be regarded under two aspects: first, by providing techniques, which support the development of high quality software and second, by providing techniques, which assure the required quality attributes in existing artifacts. Both approaches have to be combined to achieve effective and successful software engineering. Goal of this essay is to provide introduction to software development process and to interconnect it with software quality assurance and development.