

Manažment kvality a testovanie softvéru

LUBOŠ ZELINKA

*Slovenská technická univerzita
Fakulta informatiky a informačných technológií
Ilkovičova 3, 842 16 Bratislava*

zelinka04[zavináč]student[.]fiit[.]stuba[.]sk

Abstrakt. V softvérovom inžinierstve je kvalita softvéru jednou z najviac diskutovaných tém súčasnosti. Softvér sa stáva kriticky dôležitým pre veľké množstvo firiem a inštitúcií čím rastú nároky na jeho kvalitu a spoľahlivosť. Zvyšujúca sa zložitosť softvéru znamená vyššiu možnosť výskytu závažných chýb vznikajúcich už v počiatočných štádiách vývoja, ktoré sa v neskorších štádiách veľmi ťažko opravujú. Softvéroví inžinieri sa z tohto dôvodu orientujú na prevenciu takýchto chýb aplikáciou metód manažmentu kvality softvéru, pokrývajúcich celý proces vývoja softvéru. Táto esej sa podrobnejšie sústreďuje na hlavné procesy manažmentu kvality softvéru, testovanie ako proces vyhodnotenia kvality softvéru a základné podporné nástroje používané v procese vývoja softvéru.

Úvod

Kvalita je veľmi dôležitým atribútom softvérového produktu a často býva rozhodujúcim faktorom pre zákazníka. Pod pojmom kvalita rozumieme mieru splnenia požiadaviek zákazníka. Nejde iba o jednu vlastnosť softvéru ale skôr o súhrn vlastností, pričom zákazník určuje ich dôležitosť.

História záujmu o vytváranie kvalitných produktov siaha do 13-teho storočia, teda do čias stredovekej Európy, keď sa remeselníci začali združovať do cechov, ktoré vytvárali prísne pravidlá pre určenie kvality finálnych produktov a služieb. Odvtedy sa rôznym spôsobom menili požiadavky kvalitu výrobkov, avšak stále sa hľadelo iba na výsledný produkt. Zlom nastal až po 1. svetovej vojne, keď sa začalo hovoriť aj o kvalite procesu, ktorým bol výsledný produkt vytvorený. Táto myšlienka sa ďalej prehĺbila po 2. svetovej vojne, kedy sa upriamila pozornosť na zlepšenie kvality všetkých častí výrobného procesu. Tento prístup bol neskôr nazvaný Total Quality Management a stal sa základom dnešných prístupov k manažmentu kvality.

Prístup Total Quality Management prenikol aj do oblasti softvérového inžinierstva lebo sa ukázalo, že testovanie vytvoreného softvéru a odstraňovanie chýb je neefektívne. Chyby, ktoré vznikli v počiatočných fázach vývoja, boli veľmi nákladné na odstránenie a často vyžadovali prerobenie veľkej časti vytvoreného

softvéru. To viedlo k snahe sledovať kvalitu každej fázy procesu vývoja softvéru, aby boli vzniknuté chyby zachytené čo najskôr, najlepšie v tej fáze, v ktorej vznikli.

Manažment kvality softvéru

Manažment kvality softvéru vychádza z Total Quality Management, čiže pokrýva celý proces vývoja softvéru. Jeho cieľom je zabezpečenie potrebnej úrovne kvality softvérového produktu. Zvyčajne zahŕňa definovanie vhodných štandardov kvality a procedúr, pričom sa snaží zabezpečiť ich dodržiavanie.

Kvalita je veľmi dôležitá hlavne pri veľkých softvérových projektoch, kde sa vyžaduje spolupráca viacerých oddelení. Je potrebné, aby si každý, kto na danom softvérovom projekte pracuje, uvedomoval dôležitosť vytvárania kvalitného softvéru a tak aj pristupoval k svojej práci. Pri týchto projektoch sa kladie dôraz aj na vytváranie dokumentácie týkajúcej sa kvality, ktorá je indikátorom stavu vyvíjaného softvéru z hľadiska kvality.

Manažment kvality softvéru je v podstate súbor procesov, ktoré sa skladajú z množstva aktivít. Každý proces pokrýva určitú časť vývoja softvéru a často dochádza k ich prekryvaniu. Úlohou procesov manažmentu kvality softvéru je určenie, do akej miery softvérové produkty spĺňajú alebo budú spĺňať požiadavky zákazníka, akú hodnotu zákazníkovi poskytnú a akú poskytnú kvalitu softvéru potrebnú pre splnenie požiadaviek na softvér.

Medzi základne procesy manažmentu kvality softvéru podľa štandardu IEEE12207 patria:

- Procesy zabezpečenia kvality
- Procesy verifikácie
- Procesy validácie
- Procesy revízia
- Procesy auditov

Zabezpečenie kvality softvéru

Štandard IEEE12207 definuje zabezpečenie kvality softvéru ako proces, ktorý poskytuje adekvátnu záruku, že softvérové produkty a procesy v životnom cykle produktu vyhovujú ich špecifickým požiadavkám a dodržiavajú ich pevne stanovené plány. Z tejto definície vyplýva, že zabezpečenie kvality softvéru je proces, teda súbor nejakých metód a prístupov, ktorý sa sústreďuje na celý životný cyklus softvérového produktu tak ako samotný manažment kvality softvéru. Z tohto dôvodu sa často tieto pojmy mýlia a zamieňajú.

Proces zabezpečenia kvality softvéru sa snaží udržiavať kvalitu počas celého vývoja a údržby softvéru vykonávaním rôznych druhov aktivít v každej fáze, čo môže

viest k skorému odhaleniu problémov, ktoré sú takmer nevyhnutnosťou každej komplikovanej činnosti.

Dôležitou súčasťou procesu zabezpečenia kvality softvéru je plán. Plán určuje organizáciu a účel programu zabezpečenia kvality. Dobrý plán zabezpečí vytvorenie vhodného návrhu, opatrnú implementáciu a splnenie všetkých požiadaviek ešte pred vydaním softvérového produktu. Výborný plán zahŕňa aj analýzu nedostatkov a priebežné vylepšovanie kvality procesu vývoja [2]. Takýto plán je charakteristický pre väčšinou veľké firmy, ktoré pôsobia v oblasti vývoja softvéru už dlhšiu dobu.

Štruktúra plánu zabezpečenia kvality softvéru zvyčajne obsahuje nasledujúce časti [3]:

- Predstavenie produktu
- Plán produktu
- Popis procesu
- Ciele kvality
- Riziko a manažment rizík

Okrem týchto základných častí obsahuje plán [1]:

- Špecifikáciu rozloženia aktivít a úloh s ich cenou a potrebnými prostriedkami, všeobecné ciele manažmentu a časový plán.
- Identifikáciu dokumentov, štandardov, praktík a konvencie riadenia projektu a ako budú kontrolované a monitorované.
- Identifikáciu opatrení, štatistických techník, procedúr pre oznamovanie chýb a opravné akcie, prostriedkov ako napríklad nástrojov, techník, metodológií, bezpečnostných opatrení pre fyzické médiá, školení, správ o zabezpečení kvality a dokumentácie.
- Určenie aktivít zabezpečenia kvality alebo iných typov aktivít opísaných v softvérových plánoch.
- Akceptačné kritériá ako aj aktivity podávania správ a manažmentu, ktoré sú kritické pre kvalitu softvéru.

Napriek tomu, že sa v dnešnej dobe kladie dôraz na kvalitu pri každom softvérovom projekte, požadovaný stupeň kvality výsledného produktu nemusí byť rovnaký. Všetko závisí od toho, aké úlohy bude softvér plniť a v akej oblasti bude nasadený. Podľa [2] spektrum zabezpečenia kvality pokrýva širokú oblasť:

- Výskumný a experimentálny softvér – požiadavky na kvalitu sú nízke a proces zabezpečenia kvality zvyčajne zahŕňa iba ladenie a regresné testovanie.

- Nástroje pre obchod, výrobu a zábavu – očakáva sa, že takýto softvér bude fungovať, avšak občasná chyba nie je prekvapením.
- Nástroje kriticky dôležité v oblasti obchodu – vyžaduje sa vyšší štandard plánovania a kvality. Plán vývoja softvéru tohto typu vyžaduje podstatnú investíciu do zabezpečenia kvality.
- Systémy, ktoré sú rozptýlené a ťažko opraviteľné – sú ťažko dostupné alebo prístup k nim je veľmi nákladný, čo vedie k potrebe extenzívneho testovania a návrhu vzdialeného prístupu. Takéto systémy musia spĺňať vysoké nároky na kvalitu a je potrebné ich dôkladne otestovať v rôznych prostrediach.
- Životne a účelovo kritické systémy – zlyhanie môže spôsobiť straty na ľudských životoch, alebo rozsiahle kolapsy rôznych systémov. Testovanie a zabezpečenie kvality často spotrebuje polovicu finančných prostriedkov a času potrebného na vývoj systému.

Zabezpečenie kvality je v rámci organizácie zvyčajne realizované skupinou ľudí, ktorá je nezávislá od ostatných skupín podieľajúcich sa na vývoji softvéru. Táto skupina ľudí zodpovedá za definovanie procesu zabezpečenia kvality a jeho monitorovanie. Oddelenie zabezpečenia kvality nemusí byť veľké, pokiaľ majú zamestnanci tohto oddelenia nezávislosť, znalosť procesu a pochopenie produktu. Z toho vyplýva, že zamestnanci zaoberajúci sa zabezpečením kvality musia byť veľmi skúsení v tejto oblasti.

Verifikácia a validácia

Verifikácia a validácia sú procesy, ktoré kontrolujú či daný softvérový produkt zodpovedá špecifikácii a plní určený účel. Oba procesy prebiehajú paralelne s vývojom softvéru, aby boli chyby odhalené a opravené čo najskôr, čím sa dosiahne efektívnejšie využitie finančných prostriedkov. Hlavným cieľom verifikácie a validácie je odstránenie kritických chýb ešte pred nasadením softvéru do prevádzky.

Procesy verifikácie a validácie sú komplementárne, ale každý z nich má trochu odlišný cieľ. Verifikácia sa snaží určiť, či vyvíjame softvérový produkt správnym spôsobom zatiaľ čo validácia skúma, či vyvíjame správny softvérový produkt. Z toho vyplýva, že verifikácia predchádza validáciu. Verifikácia vyhodnocuje požiadavky, špecifikáciu, plány, dokumenty a kód, pričom využíva prevažne statické metódy, ktoré nevyžadujú vykonanie programu. Validácia sa vykonáva na vytvorenom produkte prevažne pomocou dynamických metód čiže testovania.

Tak ako pri procese zabezpečenia kvality aj pri procesoch verifikácie a validácie je potrebné vytvoriť plán. Úlohou plánu je jasné zadelenie zdrojov, rolí a zodpovedností. Výsledný plán by mal obsahovať minimálne tieto časti:

- Úvod a prehľad
- Testovací plán komponentov a procedúry

- Funkčný a systémový testovací plán a procedúry
- Akceptačné testy a príprava na nasadenie do prevádzky
- Výsledky verifikácie a validácie

Verifikáciu a validáciu môže vykonávať aj skupina ľudí, ktorí nie sú ani vývojári ani zákazníci, no môžu pochádzať z jednej alebo druhej organizácie. Takýto proces sa nazýva nezávislá verifikácia a validácia. Jedným z najväčších prínosov je nestrannosť takejto skupiny. Nestrannosť je typicky zaručená reputáciou organizácie, ktorá nezávislú verifikáciu a validáciu vykonáva. Reputácia sa ťažko nadobúda a veľmi ľahko stráca.

Revízie a audity

Revízie a audity sú podporné procesy pre vyhodnocovanie aktivít hlavne v rámci procesov verifikácie a validácie používané procesmi zabezpečenia kvality softvéru. Štandard IEEE1028 definuje revíziu ako proces alebo stretnutie, počas ktorého je softvérový produkt prekontrolovaný projektovým personálom, manažérmi, používateľmi, zákazníkmi, zástupcami zákazníkov alebo inými zainteresovanými osobami pre pripomienkovanie alebo schválenie. Ten istý štandard identifikuje aj viacero druhov revízií [1]:

- Manažérska revízia – používajú sa na monitorovanie priebehu, určovanie stavu plánov, potvrdenie požiadaviek a ich systémové rozloženie, alebo ohodnotenie efektívnosti manažmentu.
- Technická revízia – cieľom je ohodnotenie softvérového produktu na rozhodnutie, či je vhodný na jeho určené použitie. Navyše sa pomocou nej identifikujú rozpory so schválenou špecifikáciou a štandardmi.
- Inšpekcia – účelom je odhalenie a identifikovanie softvérových anomálií. Zvyčajne je zameraná iba na malú časť produktu. Pomocou inšpekcie je možné skontrolovať nefunkcionálne charakteristiky.
- Previerka – je podobná inšpekcii avšak menej formálna. Hlavnými cieľmi sú: nájdenie anomálií, zlepšenie produktu, uváženie viacerých alternatív a vyhodnotenie súladu so štandardmi a špecifikáciami.
- Audit – snahou je poskytnúť nezávislé ohodnotenie zhody softvérového produktu a procesov s aplikovateľnými reguláciami, štandardmi plánmi a procedúrami.

Testovanie

Testovanie je proces používaný na meranie kvality. Úlohou testovania je odhalenie chýb vyvinutého softvéru, odstránením ktorých je možné zvýšiť jeho kvalitu. V minulosti sa pristupovalo k testovaniu len na takej úrovni, že sa testoval iba výsledný softvér, no v súčasnosti prebieha testovanie počas celého vývoja a údržby. Testovanie sa tým stalo priamou súčasťou výroby softvéru.

Kvalitu softvéru nie je možné testovať priamo, ale môžeme testovať vlastnosti, ktoré od kvalitného systému očakávame. Dôležitosť jednotlivých vlastností závisí od oblasti prevádzky softvéru. Spomínané vlastnosti sa zvyčajne rozdeľujú do 3 skupín [4]:

- Vlastnosti externej kvality: správnosť, spoľahlivosť, použiteľnosť, integrita
- Vlastnosti internej kvality: efektívnosť, testovateľnosť, dokumentácia, štruktúra
- Vlastnosti budúcej kvality: flexibilita, znovupoužiteľnosť, udržiavateľnosť

Techniky testovania rozdeľujeme na statické a dynamické. O statických technikách sme hovorili v krátkosti pri druhoch revízií. Dynamické techniky sú klasické testovacie techniky používané už dlhšiu dobu. Základom týchto techník je vykonanie vyvinutého programu s určitou vopred pripravenou množinou testovacích údajov, pri ktorých vieme aké výsledky môžeme očakávať.

Softvér sa obyčajne testuje na viacerých úrovniach počas jeho vývoja. Kľúčovým aspektom úspešného testovania je výber vhodnej úrovne testovania a pokrytia v danej fáze softvérového projektu. Pri voľbe úrovne pokrytia pre určitý stupeň vývoja softvéru treba hlavne myslieť na to, že testovanie neodhalí všetky chyby a teda potrebná úroveň je závislá na požadovanej výslednej kvalite. Konceptuálne môžeme rozlíšiť tri úrovne testovania:

- Testovanie komponentov – overuje funkcionality jednotlivých častí softvéru, ktoré je možné samostatne testovať. Komponenty sú testované za účelom preukázania správnej implementácie vzhľadom na detailný návrh.
- Testovanie integrity – je proces testovania integrity a komunikácie medzi jednotlivými komponentmi softvéru.. Identifikuje problémy vznikajúce pri agregovaní komponentov do väčšej časti programu. Cieľom je overenie požiadaviek týkajúcich sa funkcionality, výkonnosti a spoľahlivosti hlavných častí návrhu.
- Testovanie systému – vykonáva sa na kompletnom integrovanom systéme aby sa overila jeho zhoda so špecifikovanými požiadavkami. Tento spôsob testovania sa používa hlavne na vyhodnotenie nefunkcionálnych systémových požiadaviek.

Testovanie môže byť priamo zamerané na určité funkcionálne alebo nefunkcionálne vlastnosti softvéru, čiže sleduje určitý cieľ. Podľa toho je možné rozpoznať veľké množstvo druhov testovania, z ktorých najvýznamnejšie sú:

- Akceptačné testovanie – sleduje chovanie sa systému a porovnáva ho s požiadavkami zákazníka.
- Alfa testovanie – testovanie úzkou skupinou ľudí z interného prostredia organizácie, ktorý hotový softvér používajú a podávajú správy o jeho funkčnosti.
- Beta testovanie – je podobné alfa testovaniu a rozdiel je iba v tom, že sa na testovaní podieľa zvyčajne širšia skupina ľudí z externého prostredia organizácie.
- Regresné testovanie – selektívne opätovné otestovanie systému alebo komponentu na overenie toho, či zmeny nespôsobili nežiaduce chovanie softvéru.
- Výkonnostné testovanie – kontroluje splnenie špecifikovaných výkonnostných požiadaviek
- Zátťažové testovanie – skúma ako sa výsledný softvérový produkt chová pri maximálnej vyťažnosti.

Ako bolo povedané, účelom testovania je odhalenie chýb. Na odhalenie chýb existuje viacero techník. Tieto techniky sú založené na skúsenostiach inžinierov, špecifikácii, štruktúre kódu, chybách potrebných odhaliť, oblasti použitia a povahy softvéru. Rozdeľujú sa na dve skupiny:

- Techniky bielej skrinky – techniky založené na znalosti softvérového návrhu a zdrojového kódu.
- Techniky čiernej skrinky – techniky založené iba na sledovaní výstupu programu v závislosti od poskytnutého vstupu.

Záver

Softvér sa stáva čoraz zložitejší kvôli rastúcim požiadavkám jeho používateľov. Uspokojenie týchto požiadaviek a zároveň vytvorenie primerane kvalitného softvéru je stále ťažšie a ťažšie. Z tohto dôvodu si myslím, že zavedenie prístupu Total Quality Manažment aj do oblasti softvérového inžinierstva bol krok správnym smerom. Tento prístup so sebou priniesol systematickú metódu na dosiahnutie požadovanej kvality výsledného softvérového produktu pod názvom manažment kvality softvéru. Procesy manažmentu kvality softvéru sa snažia odhaliť problémy čo najskôr a práve toto je podľa mňa najväčším prínosom týchto procesov, pretože čím skôr vznikne v procese

vývoja softvéru chyba, tým je na jeho konci táto chyba závažnejšia a ťažšie odstrániteľná.

V súvislosti so vznikom manažmentu kvality softvéru som spomínal, že testovanie výsledného softvéru nie je postačujúce na zabezpečenie jeho kvality. To však neznamená, že je na nič. Testovanie plní dôležitú úlohu pri hľadaní chýb v softvéri, hodnotení jeho kvality a porovnávaní so špecifikáciou zákazníka. Podľa môjho názoru, testovanie z manažmentu kvality nikdy nevymizne, skôr sa bude jeho význam ešte zvyšovať, lebo je to jediná konkrétna možnosť ako zistiť správanie sa softvéru v určitých špecifických podmienkach.

Použitá literatúra

1. A. Abran et al., eds., *Guide to the Software Engineering Body of Knowledge: SWEBOK*, IEEE CS Press, 2004, <http://www.swebok.org>.
2. Feldman, S. 2005. *Quality assurance: much more than testing*. Queue 3, 1 (Feb. 2005), 26-29, <http://www.acmqueue.com>.
3. I. Sommerville, *Software Engineering*, 7th ed., Addison-Wesley, 2005.
4. Hetzel, B. 1988. *The Complete Guide to Software Testing*, 2nd Ed., QED Information Sciences, Inc.

Annotation

Software quality management and testing

In software engineering software quality is one of the most discussed topics of the present. Software is becoming critically important for many firms and institutions which increases its quality and reliability requirements. Increasing software complexity means higher chance of fatal error occurrence in early development stages which are very hard to correct in later stages. Because of this software engineers orient on error prevention by application of software quality management methods covering the whole development process. This essay focuses in detail on main processes of software quality management, testing as a process of software quality evaluation and basic support tools used in software development process.