

Zabezpečenie kvality v softvérovom projekte, príprava testov, plán testovania a overovania vlastností produktu

PETER JAKUBEC

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

Abstrakt. Tento dokument sa venuje problematike zabezpečenia kvality softvérového produktu a dôležitosti testovania. Kvalita softvéru je v posledných rokoch sužovaná rôznymi problémami. Jeden z nich, ktorý je opísaný v dokumente je bezpečnosť. Stroje sú neomylné avšak ich tvorcovia nie. Systémy čelia neustálym útokom takzvaných heckerov, ktorí s obľubou testujú ich odolnosť. Vývojári sa snažia prispôbovať vývoju a na svoj softvér vydávajú stále nové a nové aktualizácie, mnohé z nich sú však len záplaty na chyby a medzery, ktoré neboli pri vývoji odhalené, prípadne nezostal čas na ich detekciu a opravu. Zlá kvalita softvéru a odstraňovanie chýb si často vyžaduje ďalšie zbytočné investície, napriek tomu európske firmy zanedbávajú kvalitu pri vývoji aplikácií.

Úvod

Pod pojmom kvalita rozumieme mieru splnenia požiadaviek zákazníka na vytvorený produkt. Rastúcu dôležitosť softvéru a jeho kvality v období nástupu informačnej spoločnosti nie je potrebné zvlášť zdôrazňovať. Odzrkadľuje sa v rozširovaní oblastí, do ktorých softvér preniká a v náraste objemu obchodu s informačnými a komunikačnými technológiami (Frühauf, 2000).

Zabezpečenie kvality softvéru (angl. Software Quality Assurance) sa snaží zaistiť, aby bol projekt kompletne vykonávaný v súlade s vopred stanovenou špecifikáciou, štandardmi a požadovanou funkcionalitou bez chýb a možných problémov. Ako vieme, nedodržanie týchto úkonov môže viesť k nečasnému odhaleniu chýb, čo vedie k časovým a materiálnym škodám. V rámci QA sa monitoruje a snaží zlepšiť proces vývoja aplikácie a to už od začiatku tohto projektu. Všeobecne je QA založené na princípe "prevencie".

Biznis je biznis

Podľa prieskumu spoločnosti Compuware až takmer 80 percent zúčastnených európskych firiem zanedbáva kvalitu pri vývoji aplikácií [4]. Priorita sa kladie na výsledný produkt, pričom sa neprihliada na to, ako bol tento výsledok dosiahnutý. Svet sa točí okolo financií a tak to vyzerá aj v softvérových firmách. Čím skôr je predsa niečo hotové, tým väčší zisk sa dosahuje a tým sme „všetci“ spokojnejší. Nikto predsa nebude investovať do niečoho ako je napríklad testovanie, radšej sa spoľahneme na schopnosti a neomylnosť vývojárov, programátorov, a dúfame, že všetko bude fungovať podľa predstáv. Risk je zisk (alebo strata). Avšak problémy nenechajú na seba dlho čakať. Prichádzajú prvé pripomienky, prvé chyby, no všetko je zatiaľ v norme, predsa nikto nie je dokonalý, chyby sa vyskytnú vždy. Horšie to je, keď prichádzajú stále, až sa projekt nakoniec predraží a končí neúspechom. Neúspech projektov stojí firmy veľké prostriedky. Odborní analytici odhadujú, že až pri 50 % IT projektov sa nepodarí dosiahnuť stanovené ciele a že nedostatočne kvalitný základ projektu je častou príčinou budúcich problémov. Firmy si túto skutočnosť uvedomujú, a preto je prekvapujúce, že aj napriek tomu stojí zabezpečenie kvality na okraji ich záujmov.

V malých firmách, málo „priestoru“

Predovšetkým v malých a stredných firmách, ktoré sú mladé, sa objavuje nedostatočne vybudovaná organizačná štruktúra a časové rezervy pre analýzu vnútorného chodu firmy a budovanie systému kvality. Veľa malých firiem je riadených majiteľom jeho vlastným štýlom, pričom celý systém práce je založený na existujúcich vedomostiach a schopnostiach malého počtu pracovníkov. Práca v malej firme kladie na jednotlivcov úlohy z pomerne širokého spektra. Od strategického rozhodovania, manažérskych zručností pri riadení firmy, cez ekonomiku a účtovníctvo až po čisto odborné a technické otázky. Ťažko sa dajú očakávať znalosti jednotlivca zo všetkých spomenutých oblastí. Toto všetko sa záporne odzrkadľuje v kvalite softvérových produktov.

Prevenčia?

V medicíne je všeobecne zaužívanou múdrosťou, že lepšia prevencia ako samotná liečba. Nielenže je finančne menej náročná ale vyhneme sa zlým pocitom, ktoré nám choroba spôsobuje. Skúsme sa pozrieť z tohto hľadiska na softvér. Prečo sa nesústrediť na produkt ktorý vytvárame a vyhnúť sa tým zbytočným chybám, ktoré by mali byť testovaním odhalené a vyliečené? Treba si však uvedomiť, že nikto vám nebude platiť za to, že budete nekonečné hodiny tráviť sedením a rozmyšľaním za počítačom. Termíny zvyknú vyvíjať tlak, pracovať treba rýchlejšie a vtedy sa chybám málokedy

vyhneme. Známi sú programátori, ktorí pracujú metódou pokus-omyl, kedy doladujú kód dokým sa program netvári ako má, pritom však ani nerozumejú ako to vlastne funguje. Ja niekedy, keď neviem pochopiť princíp, snažím sa dospieť od vzorových vstupov k vzorovým výstupom a to tiež nemusí byť veľmi uspokojujúce. Táto množina býva limitovaná a vždy sa môže vyskytnúť prípad pre ktorý je výstup nesprávny. Niekedy musí testovanie počkať, klient potrebuje vidieť výsledky. Preto je dôležité počítať s chybami už od začiatku a vyhradiť na testovanie potrebný čas, finančné a ľudské zdroje.

Hackeri, pomáhajú alebo škodia ?

Čo sa zanedbá pri testovaní, stáva sa zaujímavé pre skupinu ľudí nazývaných hackeri. Hackeri sú obyčajne nadšení programátori, ktorých záľubou je skúmanie a bádanie dokonalosti rôznych technológií. Aj keď možno nie každý si to prizná, ale určite ste už využili ich služby aj vy. Aký by bol predsa internet, keby sme si nemohli stiahnuť freeware cez ftp(file transfer protocol) alebo http(hypertext transfer protocol) a nemohli používať operačné systémy určené na chod týchto aplikácií? Sú to práve hekeri, ktorý využívajú rôzne bezpečnostné medzeri a generujú kľúče a cracky na používanie softvéru, za ktorý by sme inak zaplatili aj tisícky korún. Z času načas sa podarí hekerom nabúrať niektorú z databáz veľkých firiem a tu už dochádza nielen k finančným stratám, ale aj úniku osobných údajov, ktoré môžu byť veľmi jednoducho zneužitú. Existujú však aj takí, ktorým nejde ani tak o to aby napáchali čo najväčšie škody, ale len chcú dokázať svoju moc, dokonca sú takí ktorí to robia s dobrým úmyslom, poukázať na nedostatky. V poslednej dobe sa stáva hitom, nechať otestovať bezpečnosť systémov práve hekermi a výsledky bývajú prinajmenšom zaujímavé.

Život-zachraňujúce(ohrozujúce) prístroje

Počítače a softvér dnes už prenikli takmer do všetkých oblastí nášho života. Odbremeňujú nás od zbytočných monotónnych úkonov, prekvapujú presnosťou a rýchlosťou zložitých výpočtov, no to najdôležitejšie je, že stroje sú neomylné. Táto neomylnosť však stále stojí na šikovnosti vývojárov a programátorov. V odvetviach ako zdravotníctvo, kde počítače dokážu často nahrádzať životné funkcie človeka, je neomylnosť nevyhnutnosťou. Logicky, práve v takýchto prípadoch zohráva testovanie najdôležitejšiu úlohu vo vývoji. Často prejde niekoľko rokov, kým sa vyvíjaná technológia dostane z laboratórií na svetlo sveta. Jej život sa začína dlhým navrhovaním, ešte dlhším vývojom, potom príde na rad pár myšiek, potom niečo väčšie, skupinka dobrovoľných študentov, ktorý si nechajú za pár korún vítať do kolena a ak aj po týchto pokusoch ostanú všetci zdraví, môže sa oslavovať. Avšak financovanie takýchto projektov je finančne veľmi náročné a určite to nie je nič do čoho by sa púšťal drobný podnikateľ, pokiaľ by samozrejme nemal skvelý nápad a aspoň akú takú istotu jeho úspešnej realizácie.

Zvyk je železná košľa

Osobne sa tiež radšej držím zaužívaných princípov a pri svojej práci sa nesnažím priveľmi špekulovať. Venujem sa tvorbe webu a na väčšinu požiadaviek existujú zaužívané riešenia. Takto viem presne, ako mám niečo spraviť, a väčšinou mi to trvá vždy rovnaký čas. Môj kolega neustále špekuluje nad tým, prečo je to implementované takto a nie takto, a aj keď z času na čas príde na nejakú zaujímavú myšlienku, podľa šéfa väčšinou len zabíja zbytočne čas. Nie všade sa proste nájdu financie na tvorivejšiu a plodnejšiu činnosť. Na vývoj nám ostávajú škaredé jesenné večere v teple domova.

Kvalita

Predchádzajúce časti boli skôr zamerané na bezpečnosť softvéru, treba však zdôrazniť, že bezpečnosť je len jeden z atribútov určujúcich kvalitu. Vymedziť však úplne presne tento pojem môže byť veľmi ťažké. Každý rozumie pod týmto pojmom niečo iné. Pre klienta je produkt kvalitný, keď sa nekadzí, netreba vynakladať veľké prostriedky na jeho údržbu a slúži jeho potrebám. Taktiež, keď sedím doma a niečo si objednávam, nech je to aj softvér, sledujem podobné vlastnosti, napriek tomu že som programátor. Keď si však sadnem v kancelárii za stoličku a pridelia mi moju úlohu na projekte, moje kritéria sa mierne menia. Je pre mňa potešujúce, keď sa dokážem v kóde rýchlo zorientovať. To mi umožní venovať sa môjmu problému a nezabíjať čas lúštením podivných názvov funkcií, štúdiom ich podivných konštrukcií, či zabávaním sa na vtipných komentároch kolegov, ktorí sa zjavne riadne vytrápili a sem-tam zabudli niečo zmazať. Kód často prezradí čo-to nielen o kvalite softvéru ale aj o kvalite jeho autora. Avšak ani prehľadný kód nie je zárukou, že softvér je aj kvalitný, je to však výborný predpoklad na to aby sa ním stal. Taktiež detekcia a oprava chýb zaberie omnoho menej času a v neposlednom rade aj nervov. Z programátorského hľadiska je teda kvalita o prehľadnosti kódu, udržiavateľnosti, efektívnosti a čistote kódu. V súčasnej dobe existujú aj mnohé nástroje na optimalizáciu kódu, čo môže značne pomôcť a však komentáre, prípadne vhodne zvolené názvy funkcií stále ostávajú na autorovi.

Testovanie v kocke

Podme si teda niečo bližšie povedať o testovaní. „Testovanie softvéru je kritickým prvkom zabezpečenia kvality softvéru a reprezentuje konečné zhodnotenie špecifikácie, návrhu a implementácie“ [3]. Inak povedané, aby softvér mohol byť považovaný za kvalitný, musí prejsť všetkými potrebnými testami. Na druhej strane, testovanie nemôže nikdy kompletne zaručiť korektné správanie softvéru. Namiesto toho pristupuje k softvéru kriticky a hľadá v ňom chyby (defekty) alebo porovná stav

a správanie softvérového produktu oproti špecifikácii. [1] Tieto testy by sa dali rozdeliť do dvoch kategórií – statické testovanie a dynamické testovanie.

Ako si už mnohí z názvu určite domysleli statické testovanie je také, pri ktorom sa testuje niečo, čo nebeží(skúmaný objekt sa kontroluje a reviduje). Teda programový kód sa nevykonáva fyziky, len sa prezerá. Hovorí sa mu tiež inšpekcia alebo analýza kódu. Väčšinou ostáva toto testovanie na samotnom tvorcovi kódu a malo by byť vykonávané priebežne, pretože s odstupom času aj sám autor môže zabudnúť, čo myslel daným kódom(funkciou) a mohol by tak znovu analyzovať analyzované, alebo sa na to radšej vykašľať a veriť, že ak bude niekto po ňom niečo opravovať bude tomu rozumieť. Chyby objavené týmto testovaním je jednoduchšie a najmä finančne menej náročné opravovať, ako chyby odhalené akýmkoľvek iným spôsobom. Opäť platí staré známe, že chyba objavená v samotnom zárodku napáchajú menej škody ako tie ktoré sú odhalené neskôr. Cieľom statického testovanie je nájsť logické chyby, preklepy, chyby pri práci so vstupom a výstupom daného kódu.

Dynamické testovanie je možné prevádzať až keď je hotová dostatočne veľká časť implementačných prác, pretože sa jedná o testovanie aplikácie za behu. Jeho hlavným cieľom je kontrola funkčnosti programu podľa špecifikácie a odhalenie prípadných chýb. Druhým najčastejším cieľom dynamického testovania je testovanie výkonnosti alebo tiež záťažové testovanie. Toto testovanie simuluje prácu veľkého počtu užívateľov na testovanej aplikácii alebo generuje veľký počet operácií v informačnom systéme. Súčasťou záťažového testovania je meranie výkonu a odozvy aplikácie v závislosti od počtu užívateľov súčasne pracujúcich s danou aplikáciou. Na základe analýz z jednotlivých meraní sa identifikujú problémy vzniknuté záťažou, následne sa môžu vykonať opravy, aktualizácie alebo optimalizácie aplikácie. [2]

Záver

Spoločnosti si musia uvedomiť, že nie všetko z čoho sa hneď nedá vyťažiť, je zbytočné. Vývoj nie je čierna diera, ktorá pohlcuje prostriedky, ale oblasť, ktorá sa usiluje o skvalitnenie postupov a zaistenie hmatateľných výstupov a zisku. [4] Ten sa však dá dosiahnuť len zmenou prístupu ku kvalite. Jej úroveň úzko súvisí z testovaním, na čo je treba dbať už od prvej fázy vývoja. Kvalita je základom úspechu akejkoľvek firmy, a kým si spoločnosti túto skutočnosť neuvedomia, budú na seba neustále pliesť bič.

Cieľom zabezpečovania kvality a testovania softvéru je vytvorenie kvalitného, bezchybného produktu, dodaného načas. Pritom musí tento produkt spĺňať všetky požiadavky(očakávania) a zároveň byť spravovateľný a udržiavateľný.

Použitá literatúra

1. Hanko Marián, Testovanie softvéru, 2008
<http://butteland.blogspot.com/2008/08/testovanie-softvru.html>

2. Hanko Marián, Zát'azové tesovanie, 2008,
<http://butteland.blogspot.com/2008/09/aov-testovanie.html>
3. Pressman, R.S., Software Engineering(3rd Ed.): A Practioner's Approach, McGraw-Hill, Inc. 1992.
4. Žoffa Marcel, Zabezpečenie kvality, 2007
http://www.pcrevue.sk/buxus_dev/generate_tage.php?page_id=40830

Annotation

Quality assurance in software project, test preparing, product test and evaluation plan

This document deals with the problem how to secure the quality of the software product and the importance of its testing. The quality of the software suffers from different problems in the last years. One of them, which is described in the document, is its security. The machines do not make the mistakes but their creators do. The systems have to face the systematic attacks of so called hackers who enjoy testing their resistance. The developers try to adapt to the development and they create still new and new updates and patches for their software but many of them are just real patches for the mistakes and lacks which were not discovered during their development or there was not any time for their detection and correction. The bad quality of the software and correcting the mistakes often need next useless investments. In spite of it European firms neglect the quality during the development of the applications.