

Manažment konfliktov v tíme

PETER KASAN

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

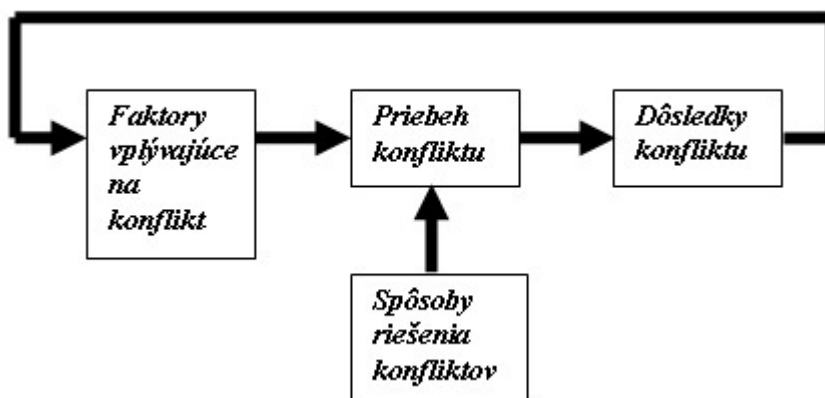
Abstrakt. Problematika tímov a vzájomné vzťahy ich členov sú v dnešnej „tímovej“ dobe stále viac aktuálne a skúmané. Vzťahy v skupine a ich dynamika síce patria do oblasti skupinovej psychológie, ale každý dobrý manažér alebo tím líder by sa mal s nimi oboznámiť. Jedným s dôležitých prvkov týchto vzťahov, na ktorý treba upriamiť viac pozornosti, je konflikt. Je dôležité pochopiť čo to je, ako vzniká, aké má fázy a ako ho treba riešiť, prípadne mu úplne predchádzať. Tieto otázky sú riešené v prvej časti tejto eseje. Druhá časť sa sústreďuje na spôsoby minimalizovania faktorov, ktoré spôsobujú konflikt a to špecificky pre dve roly v tíme – programátorov a testerov. Sú tu popísané niektoré situácie, ktoré vytvárajú napätie medzi spomínanými skupinami a tiež možný postup ako im predchádzať.

Úvod

V dnešnej dobe, kedy sú oblasti znalostí príliš rozsiahle a kedy začínajú prevládať projekty využívajúce mnohé tieto oblasti naraz, nie je možné aby boli takéto projekty zvládané jednotlivcami. Preto vznikajú tímy ľudí, kde sa každý špecializuje na určitú oblasť znalostí alebo má pridelenú úlohu v rámci tímu. Keď sa spomína pojem tím treba ho bližšie popísať. Tímy sú dynamicky interagujúce entity, ktoré sa skladajú z navzájom sa ovplyvňujúcich častíc. Akokoľvek sa zvonku zdá tím ako kompaktná entita, stále je zložená z častíc, jednotlivcov, ktorí majú svoje potreby, povahu a ciele, pričom tieto sa veľmi často nezhodujú s potrebami iných častíc, jednotlivcov. A ako to v prírode býva, rozdiely často vyvolávajú interakciu. Táto interakcia môže byť pozitívna, ale veľmi často je negatívna a prerastá do konfliktov.

Konflikt

Konflikt možno definovať ako nezhodu v cieľoch alebo záujmoch jednotlivcov či skupín. Taktiež by sa dal definovať ako proces, kedy jedna strana vníma, že jej záujmy alebo ciele sú v protiklade, alebo sú negatívne ovplyvňované záujmami a cieľmi druhej strany [1]. Možno ho znázorniť ako cyklus, v ktorom majú jednotlivé položky stanovenú svoju úlohu (pozri **Obr. 2:** Konflikt ako cyklus [1]).



Obr. 2: Konflikt ako cyklus [1]

Každý kto sa už dostal s niekým do akéhokoľvek konfliktu alebo vystupoval len ako nezúčastnená strana vie, že tento nevzniká len tak sám od seba a bez príčiny. Vždy na začiatku existujú nejaké faktory, ktoré vo väčšej alebo menšej miere podkurujú pod kotlom. A ako to býva, problémy nenechajú na seba dlho čakať a konflikt je na svete. Tento nemôže trvať večne aj keď často trvá veľmi dlho (prípady storočnej vojny). Nakoniec sa skôr či neskôr síce vyrieši, ale záleží len od spôsobu jeho riešenia aké veľké škody napácha (prípadne aký prínos z neho vznikne). Toto je dôsledok konfliktu, ktorý zase spätne ovplyvňuje faktory vzniku. Žiarivým príkladom je nikdy nekončiaci kruh „oko za oko, zub za zub“. Každopádne, spôsoby riešenia a minimalizovanie počiatkových faktorov môžu znížiť počet obehov po tomto bežecskom okruhu.

Faktory vplyvajúce na konflikt môžu byť rozdelené do viacerých kategórií. Medzi najdôležitejšie z nich patria nasledujúce:

1. **Faktory individuality:** osobnosť - povaha, vzdelanie, skúsenosti, potreby, záujmy, ciele individuálneho člena tímu.
2. **Faktory tímu:** veľkosť tímu, vedenie tímu, história predchádzajúcich konfliktov, pozície v tíme.
3. **Faktory projektu:** charakteristika vyvíjaného systému a jeho dôležitosť, dostupnosť zdrojov, časový tlak a obmedzenia na projekt.

Samotný konflikt môže vznikať na dvoch úrovniach: Na úrovni ľudí (osobností) a na úrovni rolí (úloh v tíme).

Na úrovni ľudí vznikajú konflikty najmä v dôsledku rôznych osobností v tíme. Vo veľkom množstve prípadov sa v skupine nájdu ľudia (jeden alebo viac), ktorí majú z hľadiska ostatných členov konfliktnú povahu aj keď si to veľmi často neuvedomujú. Jedná sa napríklad o ľudí, ktorí majú pocit, že všetko vedia najlepšie a vždy musia mať pravdu. S takými ľuďmi sa nielen ťažko diskutuje, ale aj pracuje. Veď prečo by ste mali vôbec vymýšľať nejaké riešenie, keď to ich je to najlepšie? Takisto veľmi „oblíbené“ povahy sú ľudia, ktorí už z princípu musia všetko a všetkých kritizovať a musia mať vo všetkom posledné slovo. Týchto pár príkladov tvorí len malú časť možných osobností (povahových čŕt), ktoré majú vysoký podiel na vzniku konfliktov

medzi členmi tímu. Zriedkavé prípady nezhôd môžu vznikáť aj na úrovni sympatií medzi členmi tímu, kedy si vyslovene niektorí členovia navzájom nesadnú a sú si navzájom nesympatický. Tento prípad je ale zriedkavý a týka sa väčšinou len dvoch individualít.

Na úrovni rolí alebo úloh v tíme vznikajú konflikty najmä z hľadiska odlišného pohľadu na projekt, z hľadiska dostupných zdrojov, časového tlaku a tiež z hľadiska sledovaných cieľov pri projekte. Možno tu identifikovať konflikty medzi testermi a programátormi, medzi manažermi a analytikmi, medzi návrhármi a programátormi, atď. Ďalšia časť (Hľadanie faktorov vplývajúcich na konflikty medzi testermi a programátormi a ich riešenie) sa bude sústreďovať podrobne práve na identifikáciu možných faktorov vplývajúcich na jeho vznik a ich minimalizovanie.

Medzi spôsoby ako sa vysporiadať s búrlivou situáciou medzi členmi tímu patria postupy a metódy, ktorých aplikovaním môžeme získať rozriešenie konfliktu. Patria sem:

1. **Vnútenie názoru** – je to spôsob riešenia konfliktu, kedy sa individuálny člen alebo skupina tímu snažia vnútiť (presadiť) svoj názor alebo riešenie druhej strane. Je to riešenie výhra–prehra. Takýto spôsob riešenia môže, ale nemusí viesť k negatívnym dôsledkom konfliktu, kedy porazená strana môže prepadnúť znechuteniu vedúcemu k zníženiu efektivity práce.
2. **Prispôsobenie** – je to opačný spôsob ako predchádzajúci. V tomto prípade dochádza k obetovaniu svojich potrieb a požiadaviek v záujme uspokojenia druhej strany. Takisto sa jedná o riešenie výhra–prehra.
3. **Kompromis** – jedná sa o riešenie výhra–výhra, kedy sa obidve konfliktné strany snažia dospieť k určitému kompromisu v spornej oblasti. Tento spôsob je asi najvhodnejší z hľadiska dôsledkov konfliktu, nakoľko majú obidve strany pocit istej výhry. Je to najvhodnejší spôsob v situáciách, kedy nie je možné uspokojiť všetky požiadavky zúčastnených strán.
4. **Riešenie problému** – je to spôsob, ktorý sa snaží nájsť možnosť, ako plne uspokojiť záujmy všetkých zúčastnených strán. Jedná sa o riešenie výhra–výhra. Používa sa v prípadoch, keď existuje možnosť, že také riešenie existuje a treba ho len nájsť. Tento spôsob je tiež veľmi vhodný z hľadiska dôsledkov konfliktu.
5. **Vyhýbanie sa konfliktu** – spôsob, kedy sú individuality alebo skupina ľahostajné k záujmom druhej strany a odmietajú sa podieľať na konflikte, čím sa vzdávajú zodpovednosti za riešenie[1]. Tento spôsob je najmenej vhodný zo všetkých spomenutých, nakoľko sa problém nerieši a dochádza k zníženiu efektivity práce. Použitie tohto spôsobu vedie veľmi často k negatívnym dôsledkom.

Dôsledky konfliktov možno rozdeliť na pozitívne a negatívne. Pozitívne dôsledky vznikajú vtedy, keď bola použitá vhodná metóda riešenia sporu. Vtedy sa dá povedať, že tento spor bol prínosom pre projekt, na ktorom pracuje tím. Negatívne dôsledky vznikajú pri nevhodnom riešení konfliktu a majú nepriaznivý vplyv na projekt a postup prác na ňom. Medzi pozitívne dôsledky patrí:

1. **Prínos k projektu:** vylepšenie procesu, spokojnosť členov tímu, zníženie nákladov na riešenie, naplánovanie ďalšieho postupu.
2. **Prínos k vyvíjanému systému** – zvýšenie kvality a použiteľnosti.
3. **Prínos k individualite (osobnosti)** – zvýšenie efektívnosti, sebauspokojenie.

Medzi negatívne dôsledky patria najmä:

1. **Negatívny prínos k projektu:** nekvalitný proces riešenia, nespokojnosť a rozčarovanosť členov tímu, zvýšenie nákladov na riešenie, nedostatočné naplánovanie ďalšieho postupu.
2. Negatívny prínos k vyvíjanému systému – zníženie kvality a použiteľnosti.
3. **Negatívny prínos k individualite (osobnosti)** – zníženie efektívnosti, rozčarovanie, apatia.

Dôsledky kladné alebo záporné spätnou väzbou ovplyvňujú faktory, ktoré sú zdrojom možných problémov. V prípade, ak sa nezhoda medzi zúčastnenými správne vyrieši, môže ovplyvniť vzťahy v tíme k lepšiemu a zredukovať pravdepodobnosť vzniku ďalšieho konfliktu. V prípade, ak sa vyrieši nesprávnym spôsobom, môže sa napätie v skupine prehĺbovať a spory sa stanú častejšími. Preto je dôležité používať správny prístup ako zvládať vzniknutú krízovú situáciu.

Možnosťou ako zredukovať konflikty v budúcnosti nie je len aplikácia správneho riešenia sporu, ale aj minimalizovanie faktorov, ktoré ho môžu podporovať. Ak napríklad tím líder vidí, že má v tíme problematickú osobu, ktorá nevie vychádzať s ostatnými členmi skupiny bez toho aby v nich vždy nevyvolala búrku negatívnych pocitov, môže to riešiť či už vymenením tejto osoby alebo prístupom k iným opatreniam riešiacim tento problém (napríklad sa môže porozprávať s danou osobou a upozorniť ju na existujúci problém).

Aspektom predchádzania vzniku konfliktov by som sa rád podrobnejšie venoval v ďalšej časti, pretože podľa môjho názoru existuje väčšie množstvo negatívnych sporov ako pozitívnych. A práve dopad tých negatívnych, aj keď v mnohých prípadoch vhodne riešených, má nepriaznivý vplyv na celý tím. Odhliadnuc od možných škôd, ktoré takéto spory môžu napáchať, vždy stojí menej úsilia a času odstrániť na začiatku neželané faktory, ako neustále riešiť konflikty, ktoré vyvolávajú.

Predchádzanie konfliktov: Prínos alebo strata?

Táto otázka je na mieste, nakoľko často je konflikt práve tou hybnou silou, ktorá posúva riešenie problému vpred. Zoberme si situáciu, že v tíme vznikol problém. Na jeho riešenie sa zide celý tím a všetci členovia sa snažia nájsť cestičky a úskalia jeho riešenia. Skupina sa skladá z rôznych ľudí s rozdielnym spôsobom myslenia a prístupom k veciam, pričom každý z nich bude hľadať vlastné cestičky. Jeden sa bude snažiť prekážku preliezť, druhý podliezť a tretí ju jednoducho obíde. Práve

v tomto bode vzniká konflikt záujmov, ktorý treba riešiť. Vhodným spôsobom (napríklad usmerňovanou civilizovanou diskusiou, pri ktorej ľudia po sebe nehádzu okolitými predmetmi) sa dospeje k záveru, ktorý môže, a často aj je prínosom k riešeniu problému. Nakoľko pri spore má každá zo strán svoje argumenty často práve z týchto vysoko potenciálne „kamene na ceste“. Pokiaľ sa tieto identifikujú, zníži sa šanca, že sa tím na ceste k problému potkne.

Na druhej strane často existujú spory, ktoré nemajú vôbec žiaden prínos k akémukoľvek problému a len vytvárajú zlú atmosféru medzi členmi tímu. Často to bývajú úplne bezvýznamné problémy, ktoré vyplývajú z povahy a zvykov jednotlivých ľudí (ako spor o to, či má byť kávovar položený pri dverách alebo v strede kancelárie na stole) až po vážnejšie problémy, ktoré už môžu mať vážne následky. Takéto negatívne konflikty treba identifikovať a snažiť sa ich odstrániť už v zárodku.

Úlohou dobrého tímu lídra je preto odlišiť pozitívne konflikty od negatívnych. Tie ktoré sú pozitívne, treba nechať prepuknúť a vhodnými metódami riadiť ich priebeh, pričom záporné treba zastaviť, kým sa vôbec stačia rozvinúť, a to hlavne minimalizovaním faktorov, ktoré ich podnecujú. Preto odpoveď na otázku ‘Prínos alebo strata?’ nie je jednoznačná a záleží len na okolnostiach.

Ďalšou kapitolou by som rád naviazal na túto problematiku a špeciálne na identifikovanie a odstraňovanie faktorov spôsobujúcich vznik negatívnych konfliktov medzi programátormi a testerami.

Hľadanie faktorov vplývajúcich na konflikty medzi testerami a programátormi a ich riešenie

Ako už bolo spomenuté, softvérový tím sa skladá z členov, ktorí majú pridelenú rolu (úlohu) v tíme, na základe ktorej majú rozdielny pohľad na projekt a sledujú rôzne ciele pri projekte. Tieto faktory nevyhnutne vedú k prepuknutiu konfliktu medzi členmi tímu majúcimi rozdielne úlohy. Z dlhodobého pozorovania boli vytipované niektoré viac alebo menej konfliktné páry rolí. Medzi tie viacej náchylné na konflikt patria aj dvojica programátor–tester. V rámci konfliktov rolí existujú ďalej rôzne úrovne (a k nim prislúchajúce faktory vzniku), na ktorých môže k týmto konfliktom dochádzať. Na základe otázok boli identifikované tieto tri základné úrovne [2]:

1. **Konflikty na úrovni procesu – faktory projektu:** nedostatok času, rôzne technické požiadavky.
2. **Konflikty na úrovni ľudí – faktory individuality:** rozdielne charakterové črty, naštrbenie perfektnosti kódu.
3. Konflikty na úrovni organizácie – faktory tímu: pozícia v tíme.

Ako si možno všimnúť, rozdelenie konfliktu do vrstiev [2] identifikuje pre každú vrstvu faktory vzniku, ktoré možno zaradiť do členenia, ktoré bolo popísané v kapitole Konflikt, a vychádza z [1]. Mój názor je taký, že všeobecný model a identifikácia atribútov [1] je postavený na podobných konkrétnych štúdiách ako [2] a vystihuje reálnu situáciu, ktorá panuje v oblasti konfliktov v tíme. Zároveň ukazuje, že všetky

reálne spory v tímoch medzi ktorýmikoľvek rolami majú rovnakú štruktúru ale rozdielne parametre.

Konflikt na úrovni procesu – faktor: nedostatok času

Čas tvorí jeden z troch kľúčových zdrojov pri akomkoľvek projekte, či už sa jedná o softvérový projekt alebo iný. V dnešnej dobe keď sa softvérové projekty týkajú systémov, ktoré majú byť značne komplexné, bezporuchové, efektívne, udržiavateľné a musia spĺňať mnoho ďalších vlastností, sa čas pri projektoch týkajúcich sa týchto systémov stáva čoraz vzácnejším artiklom. Projekt treba preto podrobne naplánovať a jednotlivým fázam prideliť čas, za ktorý by mali byť zrealizované. Často sa pritom stáva, že fáza testovania je podhodnotená a počíta sa na ňu s minimálnym možným časom. Lepší prípad nastane, keď sa tento (síce krátky) čas na testovanie dodrží. Horšie je to už ale v prípade, keď má niektorá z predchádzajúcich fáz oneskorenie a potom sa termíny dobiehajú na úkor testovania, ktoré je väčšinou na konci procesu.

A tu sa nachádza kameň úrazu, na ktorom začínajú konflikty medzi testerami a programátormi. Nakoľko väčšinou fáza implementácie (tvorby kódu) tesne predchádza testovaniu, sú väčšinou testeria závislí od programátorov. Môžu začať testovať až vtedy, keď im programátori dodajú potrebné výstupy (moduly, komponenty, atď.). Veľmi často sa stáva, že programátori si nechávajú dodanie ich výstupov na poslednú chvíľu a testeria sú potom časovo tlačení, aby stihli zrealizovať testovanie do požadovaného termínu. Faktom tiež je, že programátori ako nie koncový článok reťaze procesu, nie sú priamo vystavení koncovému termínu odovzdania (tak ako testeria) a preto u nich nie je až také silné nutkanie dodržať pridelený čas. Často sa stáva, že sa fáza implementovania systému natiahne na úkor práve testovacej fázy. Tieto časové prešľapy vyvolávajú u testerov pocit naliehavosti, čím rastie napätie a stres a konflikt je na svete.

Minimalizovanie faktorov spôsobujúcich takýto typ konfliktov sa môže realizovať viacerými spôsobmi. Jedným z nich je pridelenie dostatočného času na fázu testovania. Manažér alebo tím líder si musí uvedomiť dôležitosť testovania v celom procese a adekvátne tomu vyčleniť aj potrebné množstvo času. Druhou možnosťou je plánovať priebežné testovanie, ktoré overuje systém cyklicky. Takto sa odhalí veľká časť chýb už počas vývoja systému a fáza testovania neostane na koniec.

Možnosťou ako riešiť časové prešľapy implementačnej fázy na úkor testovacej, je vyčleniť istý časový balík, ktorý nebude pridelený žiadnej fáze projektu ale bude uložený v záložnom rezervoári pre prípad potreby. Je to vhodnejšie ako rozdeliť tento čas medzi fázy projektu, nakoľko je známe, že ktorákoľvek projektová fáza trvá minimálne toľko času koľko ho dostala prideleného. Preto ak existuje záložný čas a niektorá fáza presiahne pridelený časový interval môže jej byť stále pridelený čas z rezervoára. Takto nebude dochádzať k využívaniu testovacej fázy na dobiehanie časových prešľapov v implementačnej fáze a testeria budú mať dostatočné množstvo času na testovanie. Nebudú tlačení časom a nebude rásť stres spôsobujúci konflikty medzi nimi a programátormi.

Konflikt na úrovni procesu – faktor: používateľské vs. technické požiadavky

Druhou príčinou konfliktov na úrovni procesu je odlišný pohľad programátorov a testerov na proces. Programátori sa sústreďia na kreatívne postupy, snažia sa nachádzať nové možnosti riešenia a nové cesty ako riešiť zadaný problém. Sú orientovaní prevažne na technickú stránku systému a tomu podriaďujú aj svoje ciele. Na rozdiel od nich testerí majú za úlohu testovať systém tak ako sa bude používať. Znamená to, že sa musia čiastočne postaviť do roly používateľa a pozeráť sa na systém jeho očami. Z toho vyplýva, že testerí sú orientovaní na funkcionálnu stránku systému a tomu podriaďujú svoje ciele. A práve toto je príčinou vzniku konfliktu. Veľmi často sa totiž stáva, že programátori použijú nový prístup alebo inováciu, ktorá sa nezhoduje s používateľskými požiadavkami, čo nakoniec spôsobí problémy testerom. Z pohľadu programátora je to vylepšenie riešeného problému, ale z pohľadu testera komplikovanie jeho roboty.

Jedným z možných riešení je zdefinovanie spoločného cieľa pre jednotlivcov aj pre skupiny. Cieľ musí byť dostatočne motivujúci pre všetky skupiny, aby sa pre tento cieľ nadchli všetci a podriadili mu svoju činnosť. Nemal by to byť len cieľ, ktorý kladie dôraz na včasné dokončenie vyvíjaného systému. Ciele by mali byť motivujúce ako napríklad vyvíjanie kvalitného systému, ktorý bude bezporuchový a bude ho možné nasadiť v niektorej prestížnej oblasti a podobne. Druhou možnosťou, ako predchádzať konfliktom na tejto úrovni, je poskytnúť testerom a programátorom opačný pohľad na proces. Tento spôsob zahŕňa príležitostné výmeny rolí, medzi programátormi a testerami, čo im umožní vyskúšať si ako vidia proces ich kolegovia.

Konflikt na úrovni ľudí – faktor: rozdielne charakterové črty

Pri popise konfliktu bolo spomenuté, že tieto nevznikajú len na úrovni rolí v tíme (pri procese) ale tiež na úrovni ľudí. Každý človek má iné myšlienkové pochody a iné charakterové črty, pričom tieto môžu byť ovplyvnené (a často aj bývajú) profesiou alebo špecializáciou toho ktorého človeka. Takisto bolo dokázané, že ľudia si často vyberajú profesiu podľa ich charakteru (osobnosti). Napríklad, testerí sú často popisovaní ako pedanti sústredení na detaily, a skupiny, ktoré vytvárajú, sú popisované ako značne súdržné. Naopak programátori sa často popisujú ako ľudia kreatívni, temperamentní, individualistickí (nevytvárajúci pevné skupiny)[2]. Je to celkom pochopiteľné nakoľko si to vyžaduje ich špecializácia. Tieto charakterové črty pravdaže nemusia byť pravidlom (vždy sa nájdu výnimky). Dokonca by sa dalo povedať, že v dnešnej dobe individualita u programátorov ustupuje. Je to dané rôznymi tréningovými kurzami na vysokých školách, kde je aktuálny trend smerovať študentov k spolupráci v tíme. No nech je to ako chce, rozdielne charakterové črty často spôsobujú nedorozumenia, ktoré môžu prerásť až do konfliktu.

Prvou z možností ako minimalizovať negatívne konflikty na danej úrovni je tréningovanie členov tímu (či už ide o programátorov, testerov ale aj ostatných) v metódach riešenia konfliktov. Tieto metódy rozvíjajú schopnosti členov tímu v oblasti komunikácie, riešenia problémov a mnohé ďalšie. Takéto schopnosti u členov tímu pomôžu značne minimalizovať negatívne dôsledky sporu.

Druhou možnosťou ako ich minimalizovať je posilovať vzťahy medzi jednotlivými skupinami tímu. Veľmi často sa napríklad stáva, že programátori a testerí spolu komunikujú len v prípade, ak nastane niekde problém alebo chyba. Takýto prístup nie je najvhodnejší z hľadiska budovania dobrých vzťahov. V záujme zlepšovania vzťahov medzi skupinami tímu (napríklad medzi programátormi a testerami) je vhodné realizovať spoločné akcie, kde sa členovia týchto skupín môžu stretnúť, zoznámiť sa a porozprávať sa. Dôležité je aby išlo o aktivity, ktoré členov tímu spájajú (či už ide o športové aktivity alebo posedenie pri pive).

Konflikt na úrovni ľudí – faktor: *naštrbenie perfektnosti kódu*

Je dané povahou človeka, že je hrdý na svoje výtvory. Z vlastnej skúsenosti viem, že u programátorov môže byť tento pud značne posilnený. Sú hrdí na svoj kód a často naň nedajú dopustiť. Na druhej strane úlohou testerov je hľadať chyby a upozorňovať na ne. Z povahy ich práce vyplýva, že sú hrdí na schopnosť odhaľovať chyby a sú radi keď nejakú odhalia. A tu je lokalizované epicentrum konfliktu. Perfektný kód verzus jeho spochybňovanie. Programátori často vnímajú snahu testerov nájsť chyby v ich kóde ako spochybňovanie ich schopností (aj keď to tak často nie je). Podľa môjho názoru by mali byť skôr radi, ak sa podarí odhaliť chyby ktoré spravili, aby sa z nich poučili a aby sa ich kód pri najbližšej príležitosti konečne priblížil „dokonalosti“. Bohužiaľ, často to tak vnímané nie je, a preto s tým treba niečo robiť. Faktor vzniku konfliktu je ten, že tester musí hľadať chyby v kóde a oznámiť to programátorovi. Otázkou je, akou formou mu to oznámi respektíve ako poukáže na nájdené chyby. Často sa reportovanie chýb podobá diplomatickému jednaniu.

Spôsobom minimalizovania vzniku sporu v tomto prípade je tréningovanie testerov v diplomatickom prístupe (spôsobe), ako oznámiť programátorom kde a aká chyba nastala. Pokiaľ bude oznámenie podané vhodným spôsobom, slušnou formou a do istej miery diplomaticky, pravdepodobnosť vzniku konfliktu by sa mala značne znížiť.

Konflikt na úrovni organizácie – faktor: *pozícia v tíme*

Veľmi často sa v tíme stáva, že niektorí členovia považujú svoju úlohu pri projekte za tú najdôležitejšiu. Je to často dané hlavne povahou danej osoby a často aj vplyvom okolia. Ak je neustále kladený dôraz na niektorú úlohu v tíme a tá je neustále vyzdvihovaná do popredia pred ostatné úlohy, môže osoba na danej pozícii získavať pocit nadradenosti a nenahraditeľnosti. Naopak, ak je niektorá pozícia prehliadaná, môže osoba ktorá ju vykonáva, nadobudnúť dojem menejcennosti. Takéto pôsobenie prostredia nemá vplyv len na osoby vykonávajúce dané úlohy, ale aj na vnímanie osôb v okolí. Ak napríklad členovia tímu vidia, že tím líder prehliada alebo podceňuje niektorú úlohu (rolu), môžu ju aj oni začať podceňovať a prehliadať.

Veľmi často sa v softvérových skupinách stáva, že úloha testerov je podceňovaná a testerí potom musia neustále bojovať o uznanie a rešpekt ostatných členov. Z praxe je tiež známe, že úloha programátorov je často neúmerne zvýrazňovaná, čo u nich môže viesť k pocitu nadradenosti a môžu začať nazerať na testerov zhora. Títo si to často uvedomujú, čo vedie k zvyšovaniu napätia medzi nimi. Toto napätie často prerastá až do konfliktu.

Riešením tejto situácie môže byť rovnocenný prístup zo strany tímu lídra (manažéra) ku všetkým úlohám v tíme. Dôležité je, aby neprehliadal žiadnu úlohu a naopak aby vyzdvihoval členov nie na základe ich úlohy v skupine, ale na základe dobre odvedenej práce a prínosu k projektu.

Záver

Softvérové tímy (ale aj tímy vo všeobecnosti) sa skladajú z nesúrodnej skupiny osobností, ktoré zastávajú určité úlohy. Pretože jednotliví členovia majú rozdielne povahy a úlohy, ktoré v skupine predstavujú, sa veľmi často ovplyvňujú, vznikajú rozpory v záujmoch a cieľoch, čo vedie k vzniku konfliktov. Úlohou tímu lídra je vopred identifikovať a následne minimalizovať faktory vplývajúce na vznik konfliktov, a tým zabrániť aby vôbec vznikli. V prípade, že už konflikt vznikne, aj napriek snahám mu predísť, musí byť tím líder schopný identifikovať a použiť vhodné metódy na jeho vyriešenie. Použitie vhodnej metódy mu umožní minimalizovať jeho negatívny dopad a niekedy mu dokonca použitie správnej metódy v pravý čas poskytne nástroj, ako obrátiť tento konflikt v prospech projektu a tímu.

Použitá literatúra

1. Barki, H., Harvick, J.: Interpersonal conflict and its management in information system development. *MIS Quarterly* 25, 2 (June 2001), 195-228
2. Cynthia F. Cohen, Stanley J. Birkin, Monica J. Garfield, Harold W. Webb: Managing Conflict in Software Testing. In *Communications of the ACM*, Vol. 47, No. 1 (January 2004), 76-81.

Annotation

Conflict management in team

In this „Team era“, problematics of the teams and the mutual relations of their members are currently being the object of a lot of research. The interactions in a group and its dynamics belong to the area of group psychology but a good manager or a team leader should be aware of this topic. One of the most important part of these relations that we should focus on, is conflict. It is very important to understand what is a conflict, how does it occur, what phases does it have and how solve these situations or how to avoid them. These questions are handled in the first part of this essay. The second part is focused on the means of minimizing of the factors, which cause the conflicts, namely two roles in the team – the programmers and the testers. There are also described some situations, which make pressure between the two mentioned groups and some possible tactics of minimalizing them.