

Michal Ficek - Zkušenosti z projektu SS7Tracker

(m.ficek@volny.cz)

Prerekvizity projektu.....	1
Korekvizity projektu	1
Výběr projektu	2
Recyklace	2
Výběr členů týmu.....	2
Vedoucí projektu.....	3
Vedoucí týmu	3
Komunikace v týmu	4
Schůzky	4
Přehled práce na projektu	5
Studium v zahraničí	6
Verzování	6
Dokumentace	7
Dokumentace – software.....	9
Odevzdání projektu.....	10
Licence	10
Obhajoba – prezentace	11
Jak poznat toho kdo se surfuje a co s ním	11
Závěrem.....	12
Poděkování	12

Prerekvizity projektu

Před zahájením výběru projektu, natož zahájení prací, doporučuji pročíst si všechny zkušenosti z projektů předchozích. Zabere to čas, ale víte na co se máte duševně připravit a čeho se vyvarovat.

Počítejte s tím, že dříve byly pravidla pro projekty jiná, podle toho jsou také starší zkušenosti „zkreslené“, leč obecně stále platné!

Korekvizity projektu

Velmi doporučuji následující předměty, které se na MFF UK vyučují:

- **Operační systémy** vás uvedou do práce na větší semestrálce v malém kolektivu. Maximální zkušenost a podle jednoho z cvičících „programátorské peklíčko“. Mohu jen potvrdit, ale pozitivní zkušenosti převažují.
- **Řízení projektů** poskytuje velmi šikovné rady na ukočirování byť pětičlenného kolektivu, ve kterém budete projekt vytvářet. Naučíte se postupy, které vám usnadní nejen vedení projektu, ale hlavně pochopení základních principů časování, kritické cesty, skladbu týmu a další.
- **Doporučené postupy v programování.** Není co dodat.
- **Softwarové inženýrství pro praxi.** Celkové pojetí procesu vývoje software. Naučíte se řadu poznatků z této oblasti a nebudete tak muset „vymýšlet kolo“.
- **Nástroje pro vývoj a monitorování software.** Není co dodat.
- a řada dalších...

Tyto předměty vystudovat nemusíte, ale jejich absolvování před, nebo během průběhu prací na projektu vám podle mého názoru dost pomůže a ušetří pár šedých vlasů. Já jsem před

projektem „prodělal“ pouze legendární OSY, ale zato mi ostatní předměty šly docela dobře později (díky zkušenostem z projektu).

Výběr projektu

Věnujte výběru projektu nemalé úsilí a čas. Než kývnete kamarádům na „pojď s námi něco programovat“, zkuste najít radši hodně zajímavý projekt. Určitě si sežeňte placený a nejlépe hodně dobře placený projekt, například u nějaké firmy. Projektu budete věnovat nemalý čas, většinu času, hůře ještě všechen svůj čas a tak místo brigády nebo zaměstnání bokem je příjemné pracovat na něčem, za co jste placení.

Projekt by měl být *zajímavý*.

- Nebraňte se technologiím, které neznáte. Pokud jsou netriviální, komise jistě uzná náročnost a například to, že jste dva až tři měsíce studovali specifikace mobilních sítí.
- Nebraňte se většímu počtu různých technologií. Během práce na projektu může vykristalizovat určitá specializace členů týmu – unixář, databáze, kodér v C++, kodér v Javě, low-level programátor apod.
- Projekt, který bude nasazen, popřípadě v době obhajoby již nasazen je, mně osobně přijde užitečný. Můžete se pochlubit například počtem uživatelů, přínosem projektu na „živých“ datech apod.
- Netradiční projekt určitě zaujme. Webový portál v PHP je jistě také projekt, ale byla jich již řada. Zato takový model spalovacího motoru... Každému podle jeho gusta. Ovšem u stokrát ohraných projektů má komise s čím srovnávat (pokud si pamatuje předchozí projekty), nebo má minimálně dobrý odhad na to, jak vámi naprogramovaná úloha běžně vypadá a v jakém stavu ji jako hotovou prezentujete vy.

A ještě jedna drobná rada – najděte se *dobře* placený projekt. Ať už se ve vašem týmu stane cokoli, pořád zůstane stmelovací prvek smlouvy se zadavatelem projektu, popřípadě vaše čistě zjištěná motivace.

Je docela výhodné pracovat na projektu například v práci. Pokud pracujete v „matfyzácké“ firmě s několika svými spolužáky, není od věci naimplementovat nějaký netriviální a rozsáhlejší program / informační systém, za který budete placeni, děláte jej v pracovní době a spojete tak potřebné s užitečným. Příjemné to může být možná také, ale s tímto přáním opatrně.

Recyklace

Pokud si vyberete projekt šikovně, prodáte jeho části na různých předmětech ve škole. Například specifikace v UML na předmětech podobného ražení. Pokud budete implementovat nějaký informační systém, máte předmět Informační systémy skoro zadarmo. A tak dále. Zápočetník do Javy může být část GUI programovaného v Javě, stejně tak zápočet na Programování v C# a .NET.

Recyklace by neměla být primární motivací při výběru projektu, ale je šikovné se nad ní zamyslet. Na druhou stranu – i když budete mít maximálně nerecyklovatelný program, často i z něj něco jinde použijete.

Výběr členů týmu

Výběr členů týmu je popsán snad v každé zkušenosti z projektu. Je stěžejní pro úspěch projektu.

Nemusíte na projektu pracovat s lidmi, které dobře znáte. V principu myslím platí, že nemusíte svého budoucího kolegu dobře znát, ale měli byste vědět, jakou práci odvádí. Někdo s kým chodíte po škole na pivo bude dobrý do začátku projektu, protože se znáte jako kamarádi. Ale až po dvou měsících zjistíte, že programuje jako čuně, bude docela pozdě.

Seznamte se s tím, co váš budoucí kolega již programoval, na čem pracoval, nebo jaký má přístup k práci. Ideální je osobní zkušenost – například ze semestrálky Operačních systémů. To ale nejde vždy a tak bude výběr zbytku (nebo celého týmu) vždy trochu loterie. Udělejte si obrázek z toho, jak včas odevzdávají budoucí členové týmu své zápočty, jak plní úkoly, jestli stíhají školu i práci apod. To vše vám ale neřekne moc o tom, jaký je váš kolega „parták“. Můžete mít v týmu geniálního programátora, ale když bude líný, nebo nebude chtít pracovat, jsou vám jeho zkušenosti na nic. A naopak – můžete mít v týmu člověka, který nebude specialista na nic, ale je velmi pracovitý a navíc šikovný; není líný učit se cokoli nového, instalovat a zkoušet nový software a daleko více.

Vedoucí projektu

Oficiální vedoucí projektu nemusí do řízení týmu vůbec zasahovat, ale měl by se zajímat o chod týmu. Dotaz ze strany vedoucího projektu jednou za dva-tři měsíce na to jak se týmu daří není zrovna častý, ale pokud vedoucí týmu (viz níže) odvádí dobře svou práci, nemusí to být na překážku. Vedoucí projektu by však podle mého názoru měl sledovat minimálně souhrn prací na projektu, plnění úkolů a pracovní morálku celého týmu, aby mohl ze své pozice zasáhnout. Pokud se najde v týmu surfař, je vedoucí projektu ten, který jej může z týmu vyloučit.

Vedoucí projektu má svou nezastupitelnou úlohu v konečné fázi projektu. Zejména pročtení dokumentace a její připomínkování pomůže řešitelskému týmu více než co jiného. Pokud v týmu není stylisticky nadaný člen a typograf, bude vedoucí projektu tím, kdo udělí cenné rady, jak má dokumentace vypadat. Vedoucí projektu může také tým značně podržet při obhajobě, zejména svým hodnocením.

Doporučuji vybrat si vedoucího projektu takového, který má s projekty již nějaké zkušenosti, zejména úspěšně obhájené projekty jsou dobrou vizitkou. Není to ale nutná podmínka.

Vedoucí týmu

Vedoucí projektu nemusí být zrovna ten, kdo celý tým vede a zhusta jím ani nebude. V týmu pravděpodobně vykrystalizuje role „vedoucí týmu“, jeden ze členů týmu, který píše záznamy schůzek, komunikuje se zákazníkem, navazuje komunikaci, koordinuje práci členů týmu apod. To samo o sobě zabere dost času, ale nemělo by to být výmluvou pro jinou práci na projektu. Vedení týmu neznamená, že vedoucí bude odvádět méně implementační práce.

Vedoucí týmu by měl ostatní členy motivovat svým přístupem k projektu. Nepřipravený vedoucí týmu zásadně ovlivní například průběh schůzky. Je důležité, aby vedoucí týmu uměl méně výkonné, nedej Bože surfující členy týmu upozornit na jejich klesající pracovní nasazení. Není to vždy snadné, mohu říci, že často „drobné naťuknutí“ nestačí. Vedoucí týmu by měl také umět pochválit členy týmu za jejich práci. Dobře a ochotně se chválí za kvalitní a výborně odvedenou práci. Horší je to se špatně, nebo rychle a ne dobře odvedeným úkolem. Také v tu chvíli by měl vedoucí umět chválit a tým i motivovat, ačkoli to není vůbec jednoduché. Často je přínosnější říci „je to skvělé, ale chtělo by to dodělat to a to“, než „toto to a to je špatně, ale jinak je to pěkné“. Ne všichni členové týmu musí sdílet osobní měřítko vedoucího a ten si toho musí být vědom.

Vedoucí týmu musí být přístupný kritice ze strany ostatních členů týmu. Ti by mu měli poskytovat zpětnou vazbu nejen svými pracovními výkony, ale zejména slovně. Sám vedoucí by měl občas členy týmu pobídnout, aby se vyjádřili, co se jim na jeho vedení nelíbí, co je vhodné změnit.

Komunikace v týmu

Domluvte se ještě před zahájením prací na projektu na komunikačních kanálech, které budete využívat. Sepište si čísla ICQ, Skype a umístěte je do sdíleného úložiště (SVN). E-mail je dobrá volba – zařídte si schránku, ze které se budou e-maily rozesílat ostatním.

V průběhu prací na projektu je nesmírně důležité, aby na e-maily reagovali *všichni* členové týmu. Pokud se na větu „prosím všechny o názor“ ozve jeden člověk, je něco špatně. Nikdo nesedí na e-mailu jako žába na prameni, ale denně si přečíst mail by mělo být reálně zvládnout.

Připravte se na to, že v týmu můžete mít surfaře. Jedním z varovných příznaků je, že nečte e-maily, nekomunikuje, popřípadě odpovídá pouze na nejnovější e-maily a staré nečte apod. V takovém prostředí pracovat je na hlavu. Když strávíte denně hodinu i více psaním e-mailů, na které potřebujete odpověď a přijde vám žádná nebo zmatečná odpověď, můžete jen nadávat. Z vlastní zkušenosti vím, že napomínání, domluva, nebo i řádné vynadání nemusí vůbec pomoci.

Někdo ve zkušenostech k projektu napsal, že zavedli v týmu „pokuty“. Pokud někdo neudělá svou práci, neomluví se ze schůzky nebo jinak nespolupracuje, přinese všem čokoládu. Podle mě je to příliš jednoduché a nesouhlasím s tím. Komunikace v týmu by měla být založena na tom říct ostatním včas, že na svém úkolu nebudu dělat. Těžko se někdo může urazit, když člen týmu oznámí ostatním, že má moc práce do školy a proto se týden nebude projektu věnovat. Pokud se tak neděje každých 14 dní, je to v pořádku. Nechá se domluvit úplně volno (dovolená) o prázdninách. Ale nesmí se stávat, že informace „nic jsem neudělal“ se dozví zbytek týmu až na schůzce, kde jim někdo „strčí“ čokoládu a vše je ok. To není.

Schůzky

Zaveďte *pravidelné* schůzky. Nám se osvědčily týdenní pro tým a cca jednou za 14 dní se zákazníkem. Záleží ale na povaze projektu. Domluvte se, kdy má kdo v semestru čas a najděte okno, ve kterém se budete scházet. Je důležité, aby se schůzek účastnili všichni členové týmu a to nejlépe fyzicky. Nicméně nepřítomnost člena týmu na schůzce se nevyklučuje. My jsme však zavedli

- omluvu předem, s dostatečným předstihem a s odůvodněním
- soupis vykonaných činností od minulé schůzky (co je hotovo, co není hotovo)

Zapisujte si program schůzek. Ty naše obsahovaly:

- místo konání schůzky
- čas
- zúčastněné osoby
- co není hotovo
- co je hotovo
- todo

Část „Co není hotovo“ měla být motivační. Podle mého názoru rozumný a slušný člověk, pokud má přijít před čtyři další lidi a říct jim „neudělal jsem od minula A, B a C“ jde po prvním, druhém takovém prohlášení do sebe a začne makat. Problémem jsou ale členové týmu, kteří to jsou schopni takto praktikovat většinou část doby trvání projektu a nějak jim to prostě nevádí. Pokud někdo přijde na schůzku a potetí za sebou vám s úsměvem oznámí, že něco (vše!) nemá hotovo, začněte přemýšlet jak ho motivovat, nebo co nejrychleji vypakovat. Z vlastní zkušenosti vím, že se to jen tak samo od sebe nezlepší.

Část „Co je hotovo“ slouží jako přehled vykonaných prací. Je důležité si jej vést. Poté co po dvou měsících najdete funkcionalitu, která měla být už dávno naimplementovaná a není, jde snadno dohledat kdo a kdy prohlásil „já to udělal“. Opět se jedná spíš o kontrolní a motivační nástroj, který může ale vyřešit patové situace v týmu. Příklad:

- A: Funkce X není hotová, jak to?

- B: A co tam má být?
- A: Posílal jsem ti to mailem.
- B: Kdy?
- A hledá v mailu a odpovídá: 23.11.2007
- B vidí mail evidentně prvně: Hmm, ten mi nedošel.
- A hledá v soupisu práce, v sekci „Co je hotovo“, najde řádek „B implementoval funkci X“
- B: Hmm, tak já to udělám do příště.

Pokud si myslíte, že vás něco podobného nemůže potkat, tak může. Je to nepříjemné, ale soupis práce může být použit nejen jako motivační nástroj.

Do části „Todo“ se píše co je třeba udělat do příště. Tuto část je možné rozdělit na „Spěchá“, „V nejbližší době“ a „Výhledově“.

Zápis schůzky je třeba ukládat do sdíleného úložiště co nejdříve po schůzce a zároveň jej poslat po vašem komunikačním kanálu – např. mailem.

Co bych příště udělal jinak:

- Volba vhodného nástroje pro podobnou organizaci – nějaký přehledný program, ve kterém se dá jednoduše vyhledávat. My jsme používali plain-text, nechá se s ním vyžít, ale není to moc ono. Na druhou stranu – každý jej přečte v čemkoliv.
- Pečlivě si zapisovat, co se na kreativních schůzkách odsouhlasilo (a proč), ale stejně důležité je i co se *zamítulo* (a proč). Je fakt na budku po čtvrt roce znovu vymýšlet důvod, proč nepoužíváme „tento šíleně výhodný konstrukt“, když si každý v týmu pamatuje, že jsme jej jednoznačně odmítli, jenom nevíme proč. Zabírá to čas.

Zejména lenost je v týmu hřebík do rakve. Vždy bude několik členů pracovat „trochu“ více než ostatní, ale vedoucí týmu by měl udržovat vytížení všech členů na rozumné minimální výši. Bohužel, není nic otravnějšího, než lidi do práce honit.

Přehled práce na projektu

Pište si přehled práce na projektu. K čemu? Můj osobní názor je, že *slušného a normálního člověka bude motivovat, když někdo na projektu jeho kolega věnuje více času*. Minimálně se bude cítit blbě, že zatímco se fláká, kolega toho udělal daleko víc. Že to u všech tak nefunguje jsem tušil. Ale rozdíly které mohou vzniknout jsou opravdu žalostné.

Přehled práce by obsahovat určitě:

- datum
- kdo (jméno člena týmu)
- co (konkrétní činnost, kterou dělal; pokud má nějaký výsledek, tak jaký)
- jak dlouho (čas v hodinách, my používali nejmenší jednotku půlhodinu)

Není špatné rozdělit činnosti do několika skupin. Například Analýzy, Implementace, Dokumentace, Komunikace se zákazníkem apod. Na závěr práce na projektu je díky těmto kategoriím možné nakreslit zajímavé grafy o rozložení práce v čase, vytíženost jednotlivých členů týmu nad různými činnostmi a mnohem více.

Co je však důležitější:

- Pište přehled práce na projektu ihned od počátečních schůzek.
- Ustanovte si *formát* takového seznamu práce a dodržujte jej. My jsme používali plaintext, jeden soubor pro každého člena týmu. Ačkoli jsme na začátku stanovili formát datumu, oddělovačů a formát číslic, ve finále jsme měli pět zmatlaných seznamů, z nichž si byly formátově podobné jen dva. Excel není špatná volba, ale špatně se verzuje (starší verze Office) a navíc pokud nemají všichni MS Office, ale používají třeba OpenOffice, může docházet k problémům při ukládání různých verzí sešitu.

- Pokud v průběhu práce na projektu zjistíte, že se předepsaný formát mění, zasáhněte ihned. Opravte dosavadní chyby, ujasněte si v týmu jak se má výkaz práce psát. My jsme nechali anarchii ve výkazu práce dojít tak daleko, že potom jenom přepsání (převedení) dat do použitelné podoby zabralo téměř dvě hodiny. To se může zdát jako drobnost, ale někdo to dělat musí a je to maximální vopruz.

Co je nejdůležitější:

- Vyhodnocujte přehled práce projektu *průběžně*. Ještě jednou, vyhodnocujte přehled práce na projektu *průběžně*! Třeba jednou za měsíc, nebo jednou za dva měsíce. Přinese to následující výhody:
 - o dodržuje se formát přehledu práce
 - o víte kdo se fláká

V seznamu práce na projektu se velmi rychle objeví, kdo jak na projektu pracuje. Seznam však vyhodnocujte s rozumem.

- Pokud někdo oznámil, že bude čtrnáct dní na horách, nemůže dostat po měsíci kopr, že se flákal. Na druhou stranu pokud se někdo tváří že dělá a pak vyjde najevo, že za celý měsíc odpracoval tolik jako kdyby byl čtrnáct dní na horách, je něco špatně.
- Ne všechny činnosti jdou dělat pořád. Pokud je někdo specialista na programování a nemá zrovna do čeho píchnout, určitě se to v přehledu práce projeví. Na druhou stranu toto je prostor pro vedoucího týmu, aby přiřadil dotyčnému jinou, paralelní činnost.
- Lidé jsou různí. Někdo za dvě hodiny naprogramuje zlaté tele, jiný si za dvě hodiny nerozmyslí ani datovou strukturu. S tím počítejte, je to prostě fakt. Oba budou mít výkaz práce stejný, ale výsledky budou diametrálně odlišné.

A pár slov závěrem k tomuto tématu: připravte se na to, že někdo odvede opravdu znatelně méně práce, než někdo jiný. Jenže až na konci projektu sečtete čas a zjistíte, že někdo strávil prací na projektu **čtyřikrát** méně času než někdo jiný, je už pozdě. Vyhodnocujte průběh prací průběžně.

Studium v zahraničí

Jeden člen našeho týmu odjel zhruba v polovině harmonogramu projektu studovat do Německa. Musím říct, že výkon týmu tím nijak netrpěl. Dopředu jsme si domluvili pár pravidel:

- před tím, než bude schůzka týmu, pošle kolega seznam toho co od minulé schůzky udělal, co plánuje na další týden, případně další dotazy
- pravidelně čte zápisy ze schůzek a pokud jim nerozumí, ozve se mailem, skype nebo ICQ

Práce na projektu mimo tým vyžaduje jistě značnou disciplínu a ne každý jí je schopen. Pokud bude však váš kolega v zahraničí cílevědomý a schopný komunikovat po některém ze zvolených kanálů, určitě je možné projekt včas a úspěšně dokončit.

Rozdělení práce bylo v našem případě jednoduché, protože náš kolega programoval grafický front-end v Javě a do jeho práce mu nikdo z nás nezasahoval. Pouze v pozdější fázi přibýlo propojení s několika dialogy programovanými jiným členem týmu, ale to vše snadno řešili mailem nebo po skype.

Verzování

O tom, že se má používat verzovací systém bylo již v jiných zkušenostech z projektů napsáno mnohé. Vyberte si váš oblíbený systém pro správu verzí (nebo takový který někdo z týmu ovládá a má s ním zkušenosti) a používejte jej. Předtím bych ale doporučil následující kroky:

- Stanovte si pevná pravidla, co do reporsitory ukládat a co ne (například zdrojové kódy ano, binárky ne)

- Kdo verzování ovládá, může ostatním práci s reporsitory ukázat, plus ještě lépe sepsat.
- Instalaci verzovacího systému neodkládejte a mějte jej k dispozici co nejdříve – můžete sdílet zápisy schůzek, výkaz prací, různé materiály potřebné k projektu apod.
- Přesvědčte se, že každý reporsitory umí ovládat! Nejen stahovat nové verze, ale i ukládat svoje. Naučte se navzájem jak mazat správně soubory, jak z reporsitory exportovat soubory jinam apod. Pro někoho to jsou drobnosti, ale pro jiné členy týmu to může být španělská vesnice. A zbytečně se později ztrácí čas na řešení problémů s nekonzistencí reporsitory a nebo ještě horší.

Dokumentace

Dokumentace projektu je možná důležitější než dílo samotné. Špatně dokumentovaný projekt, byť sebelepší, nemá šanci na úspěšné obhájení. Dokumentaci bude dobrý oponent týden zkoumat a pročítat. Když budete mít projekt na nestandardní platformě, nezbude mu nic jiného, a proto si dá s pročením a pochopením všeho možná docela práci. Pokud budete mít projekt, který je instalovatelný na běžném PC, bude oponent potřebovat jistě instalační manuál bez jediné chyby, uživatelský apod. Ať tak nebo tak, dokumentace je potřebná a není možné ji jakýmkoli způsobem zanedbat.

Počítejte s tím, že dokumentace musí být „kvalitně typograficky upravená“. Nejsem specialista na typografii, proto si níže uvedené rady nejlépe ověřte. Popisují zde to, co jsme použili my v dokumentaci našeho projektu. Velmi doporučuji předmět Aplikační software, ve kterém se základní typografická pravidla vyučují.

Kvalitně typograficky upravená znamená mimo jiné:

- Používání pevných mezer (za jednopísmenným předložkami a někde s uvádí i za spojkami „a“ a „i“). Na konci řádku nesmí být samostatná jednopísmenná předložka.
- Typograficky korektní font. Tzn. patkové písmo, pro nadpisy bezpatkové.
- Odsazování odstavců, první odstavec za nadpisem odsazený být nemusí. (Viz úprava bakalářské práce) apod.
- Dokumentace musí mít obsah, dále ideálně seznam obrázků a tabulek (jsou-li nějaké).
- Pokud používáte odkazy z literatury, je nezbytné zahrnout i seznam literatury. Pozor, v seznamu literatury by neměly být uvedeny knihy, na něž z textu neexistuje odkaz. Pro formátování seznamu literatury existuje česká norma. Jinak vypadá zapsána vydaná kniha, jinak článek ve sborníku, článek z časopisu, citace z diplomové práce, elektronický odkaz (webová stránka, webový archiv). Toto vše je potřeba projít a korektně zapsat.
- Seznam pojmů (glossary) je nezbytný, používáte-li velkou řadu zkratk, které nemusí být čtenáři jasné. Náš projekt byl z oblasti mobilních komunikací, které obsahují opravdu veliké množství zkratk. Proto jsme slovníček do dokumentace začlenili. Pokud je dokumentace složena z více dokumentů, vložte slovníček do každé z nich, usnadňuje to orientaci v dokumentu. Navíc v HTML dokumentaci je překlikávání do slovníčku velmi příjemné.
- Číslování stránek je minimum, v záhlaví stránky může být uveden název a číslo kapitoly.
- Seznamy (jako je tento) musí být všechny jednotné. Buď je položkou věta, která začíná velkým písmenem a končí tečkou, nebo je to text začínající malým písmenem. V druhém případě je možné „věty“ oddělovat středníkem a za středníkem začíná slovo malým písmenem.
- Tabulky i obrázky by měly být popsány, stejně tak příklady nebo ukázky kódu.
- Vytisknutá dokumentace by neměla obsahovat sirotky – poslední řádek odstavce umístěný na následující stránce, než kde odstavec začíná. A už vůbec by se neměli

vyskytovat „parchanti“ – začátek odstavce tvoří jeden řádek na jedné stránce, konec odstavce je na druhé stránce a jedná se opět jen o jeden řádek.

- Nadpisy kapitol musí být svázány s následujícím odstavcem.
- Popisy tabulek nebo obrázků musí být svázány s objektem.
- Nejsem si jistý, jestli je to tak typograficky správně, ale zarovnání od strany ke straně (jako v knížkách) vypadá minimálně lépe než „ježatá“ pravá strana.

Dokumentace by měla být kvalitně stylisticky upravena:

- bez gramatických chyb
- nepoužívat chodící slovesa: program běží, toto jde udělat, když to nechodí apod.
- viz není zkratka, ale sloveso; proto se za něj nepíše tečka
- neopakovat výrazy ve větě nebo v několika větách za sebou (výjimky jsou možné)
- nepoužívat informatický slang, nebo i vžití výrazy: filesystem je souborový systém, buttonek je tlačítko, user je uživatel; ale naproti tomu firewall není ohnivá brána
- používejte česká, všem srozumitelná slovesa – „mečovat“ od anglického „to match“ sice zná každý matfyzák, ale komise se bude tvářit že toto slovo vidí poprvé v životě
- nepoužívejte zkratky, pokud nejsou nezbytně nutné; zejména zkratky vytvořené vámi v rámci práce na projektu jsou často zbytečné

Dejte si záležet i na formátu odevzdané práce

- Dnes není problém nechat práci vytisknout na kvalitní papír. Pokud obsahuje dokumentace obrázky, měly by být vytištěny v barvě.
- Oboustranný tisk je možný, ale měl by být odůvodněn větším rozsahem dokumentace.
- CD s projektem je dobré potisknout. Ne že by CD popsané fixem bylo důvodem k zamítnutí projektu, ale prostě vypadají lépe disky potištěné. Pokud nemáte možnost potisku (hodně novějších tiskáren tuto vlastnost má), poohlédněte se po CD Lightscribe. Za pomoci ne zrovna speciální vypalovačky je možné na vrchní vrstvu vypálit laserem monochromatickou grafiku včetně textu. Jejich kvalita je velmi dobrá.
- Všechny dokumenty, které jsou součástí dokumentace, svažte. Nám se osvědčila umělohmotná kroužková vazba.
- Pokud máte dokumentů v rámci dokumentace více, umístěte je společně, například do šanonu. Na plastovou kroužkovou vazbu se prodávají průhledné umělohmotné nástavce, díky kterým je možné dokument do šanonu vložit. Také na CD existuje řada obálek s „dírami“ pro vložení do šanonu.
- Samostatné papíry (úvodní stránka, obsah šanonu) jsme vložili do průhledných košílek.
- Netiskněte programátorskou dokumentaci vygenerovanou z doxy. Je to naprosto zbytečné a bez překlíkávání mezi částmi textu ztrácí dokumentace význam.

Domluvte se ještě před zahájením psaní dokumentace na formátu jednotlivých logických částí textu (styly, resp. například použité tagy). Pravidla si sepište, umístěte do sdíleného úložiště a dodržujte je napříč celou dokumentací, nebo všemi dokumenty v rámci dokumentace. Pravidla by měla uvádět jakým textem budete psát

- názvy souborů, cesty, nebo jejich části
- výpisy programů, příkazy zadávané na konzoli (pkill..)
- tagy a tributy xml souborů
- názvy jiných počítačů
- názvy proměnných a parametrů (mohou se vyskytnout i v uživatelské dokumentaci!)
- chybové kódy
- prosté zvýraznění textu
- názvy položek menu, texty na tlačítkách, text zadávaný uživatelem do GUI, text vypsáný v okně, nadpisy oken...
- poznámky
- uvozovky (tag quote, nebo klasické uvozovky), co se smí do uvozovek vkládat

- nadpisy, které nejsou nadpisy částí dokumentace
- odkazy na jiné části dokumentu, jiné dokumenty dokumentace, nadpisy kapitol, odkazy na literaturu, odkazy do slovníčku
- klávesové zkratky
- názvy tříd, metod, interface v programátorské dokumentaci

Používejte obrázky. V uživatelské dokumentaci grafického rozhraní jsou téměř nutností. Lepší než popisovat jednotlivé prvky grafického rozhraní pouze textem, je vložit obrázek ovládacího prvku, aby si čtenář dovedl udělat představu o čem se mluví. Pěkný nápad je rozdělit obrázek uživatelského rozhraní na několik částí, ty zvýraznit, a popsat je podrobněji. Obrázky by neměly sloužit jako pouhá výplň textu a „natažení“ dokumentace. Zejména diagramy, message-flow, popis architektury nebo diagram nasazení mají své nezastupitelné místo a podle nich se vše lépe vysvětluje.

Snažte se psát dokumentaci tak, aby byla srozumitelná i v případě, že ji někdo otevře v polovině a začte se do ní. Pokud má dokument nějaké „prerekvizity“, se kterými by se měl čtenář seznámit před samotným čtením, určitě je zmiňte v úvodu dokumentu.

Nebráňte se tomu rozdělit dokumentaci na několik logicky odůvodněných částí. Zejména pokud víte, jak se projekt nebo jeho části budou používat, přizpůsobte tomu i strukturu dokumentace. Náš projekt byl složen ze dvou autonomních modulů (jakoby samostatných programů), které budou jistě využity samostatně i v budoucnu a proto jsme pro každý z nich psali zvlášť uživatelskou i programátorskou dokumentaci. Celkově se naše dokumentace skládala ze sedmi dokumentů, další (programátorské dokumentace vygenerované z doxy) byly vypáleny na přiloženém CD. Jednotlivé dokumenty:

- Vývojová dokumentace
- Instalační manuál
- Modul 1: Uživatelská dokumentace
- Modul 2: Uživatelská dokumentace
- GUI: Uživatelská dokumentace
- Programátorská dokumentace databázových struktur
- Analýza (popis algoritmu diskretní simulace, kterou jsme využili pro odhad možné zátěže na část našeho projektu)

Využijte možnosti vypůjčit si CD s projektem starších projektů. Podívejte se, jak mají členěnou dokumentaci jiní, například z tématicky příbuzného projektu. Hodně vám to pomůže s představou, co by měla dokumentace obsahovat.

Napsané kusy dokumentace je dobré v týmu číst křížově – jeden napíše, druhý zkontroluje. Vzájemně si pak napíšete připomínky a autor je podle své úvahy opraví. Není nic neobvyklého, že autor textu své překlepy nebo chyby nevidí. Po nějaké době najdete jistě v týmu funkci „češtinář“, kolegu, který má pro sloh a opravování gramatických chyb dobré oko. Ještě je vhodné najít v týmu „typografa“, který si dá zase práci s úpravou dokumentu. Ten kdo dokument kontroluje, by měl kontrolovat i korektní používání tagů a zvýrazňování textu.

Pište dokumentaci průběžně. O komentářích do kódu ani nemluvně, ale i uživatelské dokumentace nebo zejména vývojovou dokumentaci není vhodné nechat na poslední chvíli. Dokumentaci opravdu za týden nenapíšete. My jsme začali dokumentovat čtyři měsíce před odevzdáním projektu, tedy téměř v polovině projektu, a končili jsme „akorát“.

Dokumentace – software

Stanovte si formát a software, ve kterém budete projekt dokumentovat. Formát by měl být verzovatelný. Na dokumentaci se budou ideálně podílet všichni členové týmu a proto je vhodné dokumentaci umístit do reporsitory.

My jsme po delší debatě zvolili pro psaní dokumentace DocBook. Má výhodu v tom, že se jedná o XML, má rozumnou nápovědu a jeden člen týmu psal v DocBooku svou bakalářskou práci. Sepsal nám tak způsob použití a export do HTML. Bohužel export do PDF je poměrně netriviální, zvláště ve chvíli, kdy potřebujete do PDF nějaké úpravy. Původní naše myšlenka vyexportovat dokumentaci do HTML a pak tyto stránky vytisknout naráží na to, že výsledek naprosto neodpovídá typografickým pravidlům.

Konverzi DocBook do PDF jsme nakonec horko těžko, leč úspěšně, provedli čtyři dny před odevzdáním projektu. Použili jsme pdfjadetex, ale hojně jsme upravovali DSSSL styly, které jadetex používá. Pro export do HTML jsme využili saxon.

K výběru software a formátu dokumentace bych doporučil následující:

- vyberte si jakýkoli chcete
- vyzkoušejte si před samotným psaním dokumentací, že umíte formát exportovat do HTML (pro verzi na CD) a hlavně do PDF (pro verzi na tisk) při zachování všech typografických pravidel
- sjednoťte si v týmu používané tagy, nebo formát textu pro různé pojmy a dodržujte je!
- zkoušejte průběžně generovat HTML i PDF, abyste předešli pracnému opravování chyb při exportu; například to že docbooku v seznamu (listitem) musí být uveden odstavec (para) jsem zjistil až při exportu do PDF a opravoval to tak ve všech dokumentech, vnořených seznamech apod.
- neváhejte se zeptat ostatních řešitelských týmů, v čem dokumentují oni a zauvažujte nad jejich variantami (např. wiki, rovnou latex a jiné)

Příště bych sáhnul po formátu latex, nebo opět docbook, ale s pečlivou přípravou PDF exportu. Finálový textový výstup bych zvolil opět z PDF, maximum typografie je v ...texu prostě zadarmo.

Odevzdání projektu

Před odevzdáním projektu nastává tradiční jedno-dvou týdenní kolečko, ve kterém chodíte spát ve čtyři ráno a vstáváte v osm. Tedy minimálně pokud už tou dobou nesurfujete a neklidíte se pracujícím kolegům svého týmu z cesty.

Na webu projektové komise je doporučení ukončit projekt čtrnáct dní před plánovaným odevzdáním projektu. V té době byste měli mít už vytištěnou, opravenou dokumentaci a vytvořené instalační CD. Sice budete ještě čtrnáct dní nacházet různé chyby v dokumentaci, apod., ale máte čas s tím něco dělat. Pokud uzavřete dokumentaci a CD vypálíte dvanáct hodin před odevzdáním, budete v dokumentaci a na CD nacházet chyby také, ale můžete tak akorát pokrčit rameny a jít spát.

Po odevzdání projektu je čas na kontaktování oponenta projektu. Pokud budete mít štěstí, bude vaším oponentem komunikativní a na mail reagující člověk, který si dílo rád nechá předvést, popsat, vysvětlit... To je jistě výhoda. V případě, že je váš projekt určen pro nestandardní platformu, zkuste si dojednat s oponentem předvedení tam kde je vaše dílo nainstalované.

Licence

Ihned v úvodu nemusíte mít jasno v tom, jak bude výsledný projekt licencován. Je však dobré mít licenci zmíněnu v odevzdané dokumentaci projektu. Určitě uveďte licence všech použitých balíčků, software třetích stran atd.

„Užití vytvořeného softwarového díla se řídí platnými ustanoveními Autorského zákona v posledním platném znění. Na vytvořené softwarové dílo se přitom hledí jako na školní dílo ve smyslu § 60 zákona 121/2000 Sb. (Autorský zákon).“ (citace z Pravidel pro projekty)

Počítejte s tím, že pokud budete chtít projekt někomu prodat, musíte brát v úvahu předchozí odstavec.

Obhajoba – prezentace

Prezentace je jediné předvedení projektu, které členové projektové komise uvidí. Tomu by měla odpovídat také příprava na prezentaci. Pokud strávíte rok nad projektem, byla by velká škoda pohrbit všechnu námahu kvůli nekvalitní prezentaci!

Na přípravě prezentace stačí týden mezi odevzdáním projektu a obhajobou. K prezentaci bych měl následující rady:

- Úvodní slide by měl obsahovat seznam řešitelů, zvýrazněné jméno přednášejícího, jméno vedoucího projektu, organizaci pro kterou projekt vyvíjíte (pokud to tak je).
- Představení projektu musí být stručné, jasné a krátké. Je třeba jasně říct, co projekt dělá, k čemu je určen, k čemu se hodí.
- Nezabíhejte do implementačních detailů – co a jak dělá může být odpověď na dodatečné otázky, ale rozhodně ne součástí prezentace.
- Prodejte maximálně všechny význačné rysy projektu – řekněte proč je váš projekt dobrý, co zajímavého přináší, čím je výjimečný.
- Nenechte se rozhodit tím, že se členové komise možná budou bavit mezi sebou, smát se, nedávat pozor. Vyberte si jednoho dva posluchače a mluvte k nim, občas přejeďte pohledem i ostatní.

Prezentaci si určitě vyzkoušejte nanečisto. Je to otravné a nikomu se do toho nechce, ale dokud si skutečně nepustíte stopky a nevyzkoušíte trefit se *přesně* do časového limitu, nemůžete mít odhad na to jak dlouhá prezentace bude. Náš časový limit byl dvacet minut i s předvedením software. To opravdu není moc. Měli jsme 14 slide, ale na téměř každém byl obrázek – popisy obrázků ubírají značně čas. Prezentace na 15 minut + 5 minut předvedení software není místo na improvizaci; připravte si vhodné a zajímavé slovní obraty, ale nemluvte z paměti.

Je vhodné, aby byl prezentující slušně oblečen. Komise může přijít „v tepláčkách“, ale už svým zjevem dáváte najevo váš přístup k prezentaci jako takové. Pokud budete popisovat obrázky, je laserové ukazovátko nutností. Střídání prezentujících (např. dva členové týmu) je možné, ale musí být sehraní, vzájemně se doplňovat a musí mít (opět) prezentaci několikrát vyzkoušenou.

Předvedení software je nepříjemné z toho důvodu, že může prostě zkolabovat. S tím musíte počítat a proto je dobré si předvádění jednotlivých funkcí software dobře rozmyslet, naplánovat a vyzkoušet. Pokud má GUI deset obrazovek, vyplatí se možná podrobně popsat jenom tři. Obecně je předvedení software silně závislé na druhu projektu, který budete dělat. V žádném případě ale nejsou vyhovující videa nebo screenshoty. Pro řešitele projektu by možná výhodné byly, komisi to ale nepřesvědčí a na krátké filmy moc koukat nechce.

Otázky, které po prezentaci zřejmě přijdou, nechte tázajícího dopovědět celé, neskákejte mu do řeči. Získáte tak čas odpověď si promyslet. Neváhejte se obrátit na kompetentní členy vašeho týmu, pokud je dotaz moc specializovaný a vy „jenom“ předvádíte.

Jak poznat toho kdo se surfuje a co s ním

„Nechození na schůzky obvykle silně koreluje se surfváním“. (Citace z jiných zkušeností projektu). Bohužel musím konstatovat, že i surfař je schopen se účastnit všech schůzek. Na schůzku přijde (dokonce včas), ale je duchem mimo, do debaty se nezapojí, ostatním řekne co všechno neudělal, slíbí co všechno do příště udělá, schůzku si odsedí a má na týden zase pokoj.

Nevyplňování přehledu práce na projektu či jeho ignorování je poměrně jasný příznak. Pokud na něčem strávím dvanáct hodin denně, asi se budu chtít i „pochlubit“ a svou práci zapíšu. Zapisování zpětně nefunguje, to si dotyčný bude vysloveně hodiny „cucat z prstu“.

Surfař moc nekomunikuje a když už, tak mizerně (viz část Komunikace).

A co s tím? Těžká rada. Vyhodit jej by bylo řešením, ale připravíte se o člena týmu, který konec konců nějakou práci zastane. Je-li surfař jen líný, ale jinak dobrý programátor, nebo má velké zkušenosti, je zde jistě prostor pro menší či větší nátlak nebo motivaci. Kdyby ale práci vysloveně sabotoval, nebo pokud byste jej neviděli dva měsíce vůbec, pak je to jistě důvod na vyhazov. Vedoucí týmu bude mít ze surfaře dva šedé vlasy navíc, někteří členové týmu práci navíc. Zejména vedoucí projektu by se měl zajímat o to, jak jednotliví členové pracují. A říct vedoucímu projektu, že jeden z vás se „veze“, není žalování, ale nutné zlo.

Myslím si, že surfař v týmu je částečně i vinou vedoucího týmu. Ten by měl pravidelně vyhodnocovat průběh prací na projektu a podle něj i sledovat, kdo je jak vytížen. Přehození některých povinností na jiné členy týmu může pomoci vyrovnávat zátěž na jednotlivé členy týmu a třeba i urovnat surfování některého z nich.

Zkuste surfaře motivovat co nejdříve. Píšu motivovat, protože jakmile dojde na výhrůžky, je už pozdě. A také – není *čím* vyhrožovat. Maximálně tak vyhozením z týmu, což může být i motivace, ale nemusí. Může se stát, že surfaře z týmu vyhodit prostě nemůžete, a to z různých důvodů (například je váš nadřízený v práci, jako jediný má k dispozici nezbytné součásti projektu apod.). Pokud si je toho surfař vědom a hodlá v surfování pokračovat, čekají vás veselé měsíce.

Závěrem

Neberte žádné ze zde zmíněných zkušeností jako dogma, jediné možné řešení nebo jediný správný přístup. Jedná se o *mé* zkušenosti z projektu a jako takové bych je *já* poskytnul svým spolužákům, které softwarový projekt teprve čeká, nebo na něm již pracují.

Pročtěte si co nejvíce zkušeností z projektů ostatních. Jsou tam cenné rady a i řada okruhů, na které jsem si nevzpomněl, nebo které se našeho projektu netýkaly.

A nakonec ještě drobnost – neváhejte se zeptat. Ať už projektové komise, lidí co už mají projekt za sebou, ostatních řešitelských týmů, nebo kamarádů. Nikdo vás přímo do prýč nepošle a předejdete řadě nejasností nebo nejistot.

Poděkování

Děkuji Katce Dufkové za neuvěřitelné pracovní nasazení a spoustu kvalitně odvedené práce. Byla mi velkou motivací a skvělou kolegyní, se kterou jsem spolupracoval moc rád.

Děkuji mé slečně a mé rodině za trpělivost, kterou se mnou za rok práce na projektu často měli.