

Text k přednášce do IVS 2015/2016 na téma
**Generování programové dokumentace z kódu, plánování, sdílení dat
v týmu**

Dokumentace

- Uživatelská příručka je určena pro uživatele, což může být sekretářka, která absolvovala třeba zemědělskou univerzitu. Pokud jí program vypíše chybové hlášení, že pole pro číslo musí vyplnit, protože hodnota jeho parametru required je nastavena na true, nebude vědět, co má dělat. Tento příklad jsem si nevymyslel, ale pochází z jednoho programu od ministerstva kultury. Pokud se takovéto texty vyskytnou i v uživatelské příručce, je to špatné.
- Pokud bude mít uživatelská příručka 800 stran, v rámci pracovní doby ji není možné přečíst. Obvykle se jí každý lekne a ani ji neotevře a když bude tištěná, ještě bude zavazet, dokud neskončí někde v archivu.
- Referenční manuál již nemůže být stručný, protože jeho cílem je popsat všechny detaily programu z hlediska uživatele a správce, který má program instalovat a konfigurovat. Uživatel ani správce však obvykle nepotřebují vědět, jak je program napsaný. To je často součástí firemního tajemství, které by se nemělo zveřejňovat. Co by však v referenčním manuálu mělo být, jsou závislosti na externích knihovnách a programech, včetně uvedení konkrétních verzí (často se uvádí konkrétní otestovaná verze doplněná textem „a vyšší“). Pokud pak něco nefunguje, lze nahlédnout do referenčního manuálu a zjistit, co je třeba doinstalovat. U programů s otevřenými zdrojovými texty je vhodné uvádět i verzi využitého překladače, protože pro novější verzi mohou být nutné úpravy.
- Programová dokumentace slouží pro vývojáře, kteří budou program udržovat či vylepšovat. Ti potřebují vědět, k čemu slouží která část programového kódu či ve které části je implementována daná funkce.
- Když už jsou v programovém kódu komentáře, manuální psaní programové dokumentace by znamenalo vypisovat vybrané části komentářů. Obvykle navíc nemá smysl dokumentovat každý řádek kódu, ale stačí vybírat komentáře větších bloků, funkcí nebo dokonce celých souborů. Takový výběr komentářů lze provést i automaticky a dokumentaci nechat vygenerovat strojově.

Doxygen

- K tomu můžeme využít Doxygen. Je to univerzální zdarma dostupný nástroj, který podporuje celou řadu programovacích jazyků a podpora dalších přibývá i ve formě různých vstupních filtrů, které některé konstrukce jednoho jazyka převedou do jiného podobného jazyka tak, aby se správně vyextrahovaly komentáře.
- Výstupem je dokumentace, která může být v jednom či více z podporovaných formátů. Nejčastěji se využívá HTML nebo PDF.
- Pokud si instalujete i nástroj dot, doxygen může s jeho využitím vygenerovat grafy závislostí, volání funkcí apod.

Komentáře

- Jak jsem již říkal, dokumentace se generuje především z komentářů. Pokud bychom měli kód bez komentářů, vygenerovaly by se sice seznamy funkcí a nějaké grafy, žádná rozumná dokumentace by však nevznikla.
- Navíc je třeba vzít v úvahu to, že počítač sice pozná, že komentář je v daném souboru, ale ne vždy je schopen rozpoznat, k čemu patří a zda je pro dokumentaci důležitý. Pokud komentář umístíme několik řádků nad funkcí, už není jasné, zda patří k této funkci, nebo jestli je to jenom nějaká poznámka, např. k celému souboru, popis chybějící funkcionality apod. Počítač navíc v bloku textu nerozpozná, co je popis funkce, co je popis jejích parametrů apod. Je tedy potřeba komentáře upravit tak, aby byly pro strojovou analýzu

vhodné. Protože pro komentáře není dostupný všeobecný standard, musíme se vždy držet pravidel pro generátor dokumentace, který chceme využít.

Komentáře

- Než se však začneme zabývat přesnou formou komentářů pro Doxygen, je vhodné říci něco málo o dobrých zvycích při komentování. Velké firmy či projekty mají obvykle stanovena svoje striktní pravidla pro psaní komentářů, kde je řečeno, jak co komentovat a o kolik mezer odsazovat. Já zde tolik do detailů nepůjdu, ale řeknu pouze několik obecných pravidel, která je vždy dobré dodržovat.
- Jedním ze základních pravidel je, že komentářů by mělo být raději více, než méně. Některé konstrukce, které se mohou zdát jasné, jiný vývojář může chápat jenom obtížně. Nemá však smysl psát až příliš mnoho komentářů, protože kód potom může být nepřehledný.
- Komentář by měl mít vždy přínos k pochopení řešeného problému. Pokud se na řádku přiřítá jednička, můžeme okomentovat k čemu a proč. Pokud bychom však napsali pouze že se jedná o přiřetení jedničky, je to k ničemu a spíše to znepřehledňuje.
- Komentování každého řádku kódu by bylo zbytečné, ale celý soubor je třeba komentovat vždy a to jak hlavičkou, která říká, co obsahuje, tak i zápatím, které umožňuje zjistit, že je soubor celý. Komentovat je třeba i každou třídu v objektově orientovaném programování, proceduru, funkci apod.
- Názvy proměnných by měly být samopopisné, což znamená, že z názvu ihned poznáme, k čemu slouží. Příliš dlouhé názvy však znepřehledňují kód a u pomocných lokálních proměnných a počítadel jsou často zbytečné. Pokud však využijeme proměnné s nejasnými názvy, je třeba jejich deklaraci či první výskyt komentovat.
- Komentovat bychom měli také všechno, co není zcela jasně zřejmé a pochopitelné z názvů volaných funkcí, využitých proměnných apod.

Komentáře

- Pokud využíváme vnořené závorky, které ohraničují velké bloky kódu, pouhé odsazení pro přehlednost nestačí. Je-li otevírací závorka o 3 obrazovky výš a nad ní je obdobná struktura ještě několikrát, nejen že může být zdlouhavé hledání otevírací závorky, kdy dlouze jedeme kurzorem, ale můžeme i přejít a udělat zbytečnou chybu, kdy např. připišeme řádek kódu do jiného bloku či funkce. Takové závorkové struktury, jako např. tato (viz slajd) je tedy vhodné komentovat a to buď zkrácenou kopií řádku, který závorku otevírá (např. název funkce či podmínka cyklu), nebo komentářem k tomuto řádku.

Komentáře pro Doxygen

- Jak již bylo řečeno na začátku, pro generátor dokumentace je nutné komentáře upravit, resp. psát je už od začátku tak, abychom je potom mohli využít. Protože v kódu máme celou řadu komentářů, nejprve je třeba určit, které komentáře má Doxygen extrahovat. Tyto jsou určeny tím, že jsou umístěny v tzv. dokumentačních blocích. Podporovaných syntaxí bloku je více, jednou z nejvyužívanějších je varianta pro C a Javu.
- Uvnitř bloku potom můžeme využívat různé příkazy pro Doxygen, které vysvětlím dále.
- Abychom u deklarací proměnných nemuseli využívat zdlouhavé komentářové bloky, existuje k tomuto účelu i jednořádková zkrácená varianta komentáře.
 - Tato jednořádková varianta je stále blokový (tedy nikoliv řádkový) komentář.
 - Častou chybou studentů je nesprávné použití tohoto typu komentáře pro komentování něčeho, co do generované dokumentace nemá smysl dávat (např. komentář k pomocné proměnné použité pouze na několika následujících řádcích uvnitř funkce).

Komentování souborů (1/2)

- Každý soubor by měl mít komentář, ze kterého bude patrné k jakému projektu patří, kdo jej vytvořil, co obsahuje, o jakou verzi se jedná a kdy byl naposledy změněn. Některé z těchto informací Doxygen získá z konfigurace a bylo by tedy zbytečné a nepřehledné je do dokumentace umisťovat vícekrát. Proto můžeme hlavičku rozdělit na 2 části, kde 1. není korektní dokumentační blok a Doxygen ji tedy ignoruje. Druhou část hlavičky již upravíme

pro Doxygen. Příkazem `@file` řekneme, že daný blok komentáře patří k celému souboru. `@brief` je potom stručný komentář. Pokud by byl potřeba i delší komentář, uvedli bychom jej volně dovnitř dokumentačního bloku. Můžeme uvést i další příkazy, pomocí kterých uvedeme autory a další informace relevantní k souboru.

- Proč vlastně vkládat 1. část hlavičky, když ji Doxygen ignoruje? Motivačním příkladem, k čemu lze tuto hlavičku využít, může být případ z praxe, kdy došlo k poškození souborového systému diskového pole. Miliony souborů sice zůstaly nepoškozené, ale adresářová struktura byla značně poškozená a vše tak bylo pouze v jediném adresáři a často pojmenované pouze číslem. Zdrojové soubory s dobře napsanými hlavičkami bylo možné automatizovaně přejmenovat, vytrdit a obnovit složky různých projektů. Pro zbytek souborů pak bylo možné pouze manuálně hledat podle něčeho uvnitř (po paměti, podle obsahů již nalezených souborů apod. - tedy místo 5 min. třeba 3 dny práce a to pouze pro jediný z desítek programů, co zde byly uloženy).

Komentování souborů (2/2)

- Konec souboru označíme rovněž komentářem, aby bylo zřejmé, že soubor je kompletní.

Komentování funkcí

- Funkce komentujeme dokumentačním blokem nad jejich deklarací. Vždy je třeba komentovat, k čemu funkce slouží, její parametry a návratovou hodnotu. Volitelně můžeme uvést i autora a další informace, mezi které patří např. zjištěné neopravené chyby a nedoručené problémy. Doxygen nám potom umožňuje generovat seznam chyb a seznam úkolů k řešení.

Další příkazy

- Doxygen podporuje i celou řadu dalších příkazů pro komentování prvků objektově orientovaného programování, vytváření odkazů v rámci dokumentace apod. Tyto si můžete podle potřeby najít v manuálu.

Generování dokumentace

- Doxygen je konzolová aplikace, což znamená, že se spouští z příkazové řádky. Nejprve vytvoříme tzv. Doxyfile, což je soubor s konfigurací, a následně program spustíme jednoduchým příkazem. Pokud se zdrojový kód programu změní, vygenerování nové verze dokumentace lze snadno provést opakovaným spuštěním.

Doxyfile

- K vytvoření konfiguračního souboru existuje i nástroj s GUR. V tomto nástroji si konfiguraci můžete naklikat, uložit výsledný Doxyfile i spustit Doxygen.
- Starší verze tohoto nástroje neměly všechny volby do detailů, čímž byla negativně ovlivněna kvalita vygenerované dokumentace.
- V aktuální verzi lze využít průvodce se základními volbami a následně doladit nastavení v záložce Expert. Dokumentace jednotlivých položek se v této záložce zobrazuje při najetí myši.

Doxywizard

- Průvodce základními volbami nastaví pouze malé množství voleb a některé výchozí volby nejsou příliš dobré. Po využití tohoto průvodce bez následného doladění je kvalita dokumentace znatelně nižší.

Doxywizard

- Nevýhodou je, že oproti procházení Doxyfile, kde jsou komentáře ihned nad položkami a lze rychle procházet celým souborem, je zde nutné jezdit myší, aby se nám komentáře postupně po jednom zobrazovaly, a přepínat sekce. Výhodou je, že soubory lze vybírat pomocí systémového dialogu a neuděláme zde syntaktickou chybu v Doxyfile.

Sekce Doxyfile

- V šabloně souboru vygenerované Doxygenem je obsaženo spoustu komentářů, které Vám pomohou se v něm zorientovat. Je rozdělen na řadu sekcí, které lze rozdělit do 3 skupin: základní nastavení, nastavení výstupu a další sekce, což bychom mohli nazvat i ostatní.

Mezi další sekce patří např. volby spojené s preprocesorem, což je předzpracování pro programovací jazyky, které nejsou přímo podporovány.

Základní nastavení (1/3)

- Než začneme procházet jednotlivé sekce se základními nastaveními, zmíním nastavení kódování, které je velice důležité. Nastavit lze kódování zdrojových textů programu i samotného Doxyfile. U zdrojových textů se nejedná jenom o možnost využití diakritiky v komentářích, ale i o textové řetězce vypisované programem a převzaté do dokumentace v rámci možnosti procházení zdrojovým textem.
- A nyní již volby spojené s projektem. Mezi ty základní patří název projektu a verze, která se skrývá pod označením PROJECT_NUMBER.
- Nastavit je potřeba také umístění pro generovanou dokumentaci a jazyk komentářů, abychom v dokumentaci neměli např. „class obrazek“ ale „třída obrazek“ apod.
- U jazyka se na chvíli zastavím, protože mezi časté dotazy studentů patří, v jakém jazyce komentáře a identifikátory (názvy proměnných, funkcí, ...) psát.

Plně české zdrojové texty omezují možnost znovupoužití cizími programátory a obecně nejsou doporučovány ani preferovány. Lze je využít spíše u tvorby pro vlastní potřebu, u které jste si jistí, že je nikdy nebudete chtít sdílet s programátory z ciziny, nebo u aplikací, které nikdy (lze-li tohle slovo vůbec říci jistě) nebudou spravovat cizinci (např. česká webová aplikace pro malé zahradnictví XY). Při cvičných školních projektech je to také často tolerováno a v programech určených pro výuku, kde studentům ukazujete zdrojové texty, to může být i žádoucí, protože lépe pochopitelné jsou texty v mateřském jazyce.

Pokud se nejedná o mezinárodní projekt, neumíte dobře anglicky, a čeština není explicitně zakázaná, můžete psát identifikátory v angličtině a komentáře v češtině. Cizinec to pak může číst jako zdrojový text bez komentářů, což je při vhodně zvolených názvech proměnných často možné, a tam, kde se v kódu neorientuje, může využít překladač.

Ve firemní praxi to však často nemusí být tolerováno (sice v některé ryze české firmě může být, ale slyšel jsem několik historek typu "ryze česká firma - tak si uděláme přijímací pohovor v angličtině", takže bych na to nespolehal) a musíte psát komentáře i identifikátory anglicky. Česky jsou pak jen texty, které může vidět uživatel, a ty jsou obvykle umístěny v samostatném souboru, který bývá hlavní součástí lokalizačního balíčku.

- Můžeme zde povolit i optimalizaci pro některý programovací jazyk apod.

Základní nastavení (2/3)

- Konfigurační volby spojené se sestavením (dokumentace) umožňují zvolit, které programové konstrukce chceme dokumentovat a zda chceme generovat i seznam úkolů, chyb apod.
- Nastavení varování a zpráv o zpracování umožňuje vygenerovat s dokumentací i soubor, ve kterém budou obsaženy problémy zjištěné při generování. Tento bychom měli využívat, abychom např. vlivem překlepu neměli v dokumentaci chybu nebo v ní něco nechybělo. Odhalí však pouze omezené množství chyb, takže dokumentaci je vhodné vždy alespoň přehledově zkontrolovat.
 - Pokud soubor není prázdný, v dokumentaci pravděpodobně bude chyba. Pokud je prázdný, neznamená to, že tam žádná chyba není (často tam chyby jsou).
 - Typickou chybou jsou chybějící stručné komentáře k souborům, třídám, balíčkům apod.
 - v seznamech pak chybí položky v pravém sloupci. Nejčastěji je problém u jazyků, kde je v 1 souboru 1 třída a vývojář si neuvědomí, že tam mají být 3 - 4 komentáře (1. pro celý soubor, 2. pro celý soubor Doxygen, 3. pro třídu v tom souboru a případně i 4. pro balíček, který se dává do 1 souboru v balíčku).

Základní nastavení (3/3)

- Mezi poslední důležitou část základních nastavení patří nastavení vstupu, kde zvolíme zkoumané přípony souborů, jejich umístění, výjimky apod.

Nastavení výstupu

- Druhou skupinou nastavení je nastavení výstupu. Některé volby jsou zde společné pro různé výstupní formáty. Mezi tyto volby patří např. možnost umístit zdrojové kódy se zvýrazněnou syntaxí přímo do dokumentace, takže se od popisu funkce můžete proklikat až k jejímu obsahu.
- Doxygen může vygenerovat i abecední rejstřík tříd apod.
- Pro jednotlivé výstupní formáty jsou v Doxyfile umístěny samostatné sekce, které umožňují povolit daný formát a případně nastavit některá jeho specifika.
- Máme-li nástroj dot, můžeme povolit a nastavit i generování grafů. Je však důležité zvolit správný formát výstupních obrázků, aby dokumentace neměla desítky MB.

Ukázka výstupu (jednotlivé slajdy)

- Nyní zde mám ukázkou vygenerované dokumentace. Záměrně jsem vybral program, jehož komentáře jsou dobře napsané, ale mohly by být i lépe.
- Nyní se podívejme na hierarchii tříd. Seznam vypadá pěkně, ale hodily by se stručné popisy jednotlivých tříd. Ty však bohužel v komentářích chybí.
- Pokud se podíváme na konkrétní třídu, vidíme seznam jejích metod. Stručné komentáře opět chybí, takže účel metody se dozvíme až když na ni klikneme a podíváme se na detaily.
- V detailním popisu už je vše v pořádku. Pokud by nám však uvedené informace nestačily, je zde ještě detailnější zobrazení.
- Můžeme totiž využít toho, že Doxygen umí udělat ze zdrojových textů součást dokumentace. Jistě Vám neuniklo, že zde chybí komentář pro celý soubor, protože ten, který zde je, není pro Doxygen.
- Nyní se vraťme k detailnímu zobrazení informací o metodě. Jistě jste si všimli grafů volání, na kterých je názorně zobrazeno, které metody tato metoda volá či může volat a kterými metodami je volána.
- Kromě grafů volání Doxygen vygeneroval i graf hierarchie tříd, který však pro tento program a jazyk C++ není příliš dobrý.
- Nyní zde mám ještě obrázky z dokumentace lépe komentovaného programu, který je v jazyce Java.
- Na úvodní stránce nebylo vidět nic zvláštního, ale v seznamu tříd ihned vidíme, že jsou uvedeny i stručné popisy.
- Pro jednotlivé třídy zde máme vygenerovány i diagramy tříd. Tyto diagramy znáte z Úvodu do softwarového inženýrství, takže je nebudu více popisovat. Jak vidíte, nemáte-li návrhové dokumenty, můžete k získání diagramů pro neznámé zdrojové texty s výhodou využít Doxygen.

Sestavení týmu a prvotní komunikace v týmu

- Ve škole i v budoucím zaměstnání se můžete dostat do situace, kdy musíte sestavit tým lidí k řešení projektu.
- Ne vždy si však můžete zvolit členy svého týmu podle svých představ a ne vždy máte možnost ihned zjistit, který člen je zkušený a pracovitý a který nikoliv.
- Tým lidí je navíc často ustaven pomocí elektronických prostředků a členové nikdy nebudou sedět ve společné kanceláři.
- Vhodným přístupem a komunikací můžete omezit negativní dopad činnosti „nepříliš dobrých“ členů týmu.
- Základem je, že ihned po sestavení týmu zjistíte informace o jeho neznámých členech.

Komunikace

- Ke komunikaci existuje mnoho přístupů a nástrojů. Velké firmy mají svoje komunikační standardy a pravidla, která je nutné dodržovat. Kdybychom měli probírat příklady z jednotlivých firem, překročili bychom čas na tuto prezentaci. Nebudeme se tedy zabývat komplexním prostředím ve velkých firmách, ale situací, která nastává ve škole při práci

- na projektech do jednotlivých kurzů či v praxi při práci na živnostenský list.
- Situace je obvykle taková, že dostanete za úkol sestavit tým. Ve škole z lidí, kteří chodí do daného kurzu, v praxi z lidí, se kterými jste už někdy spolupracovali, a z lidí, na které dostanete v rámci zakázky kontakt. Každý však může mít jiné komunikační zvyklosti a využívat jiné komunikační kanály. Např. grafická může preferovat telefon a e-mail, pomocný programátor ICQ a specialista na danou oblast je zvyklý na diskusní fóra.
- Pokud Vy preferujete diskutování na nějakém fóru, obvykle nejste v pozici, kdy můžete někoho dalšího nutit, aby se na toto fórum registroval a poskytl tak jeho správci nějaké osobní údaje, byť jich chce pouze nezbytné minimum.
- Dohoda na komunikačních kanálech a způsobu komunikace je tedy v této situaci především otázkou kompromisu, na kterém je nutné se dohodnout.

Příklad

- Pro další pokračování v tomto tématu si stanovme příklad situace, které se tyto informace týkají. Mohl bych sice zvolit třeba příklad z práce na živnostenský list, ale raději zvolíme situaci, která Vám bude bližší a do které se můžete dostat třeba už v průběhu příštího nebo některého z následujících semestrů.
- Mějme povinný předmět studia, ve kterém dostanete zadaný týmový projekt. Každý tým musí mít 4 nebo 5 osob, ale vy máte pouze 2 kamarády, takže máte 3-členný tým. Nebo máte 5 kamarádů a jednoho přece nevyhodíte, tak se rozdělíte na 2 týmy.
- V každém případě ve Vašem týmu zůstanou volná místa a na ta se Vám někdo zapíše přes IS. Řekněme, že se přihlásí 2 lidé, o kterých nic nevíte.
- Také předpokládejme, že kamarády znáte z přednášek či ze střední školy, ale v týmu jste s nimi nikdy nepracovali. Komunikaci tedy bude nutné dohadovat se všemi.

Prvotní komunikace

- Pokud se chceme dohodnout na komunikačních kanálech, základních pravidlech pro komunikaci a dalších věcech, které je v týmu potřeba vyřešit, jako např. rozdělení zodpovědností a volba vedoucího, vždy je nejlepší uspořádat osobní schůzku. Pokud však má každý člen týmu jiný (časový) rozvrh, může být dohoda na společném místě a termínu schůzky problematická. Navíc často známe pouze e-mailové adresy či telefonní čísla ostatních členů týmu.
- Pokud má tým více než 5 lidí, začíná být dohadování pomocí telefonátů a e-mailů zdoluhavé a nepřehledné. Ne každý je ihned ochotný všem ostatním poslat, které dny v týdnu má čas a na která místa je ochoten přijít. I pokud se tyto informace vymění, stále je třeba, aby někdo zjistil, kde se časová okna překrývají a jedno z nich vybral tak, aby byli ostatní spokojeni. Pokud se navíc okna nepřekrývají, bude někdo muset ustoupit a udělat si čas. Pokud se toto nepodaří dostatečně rychle, bude další týden a dohadování může začít znovu od začátku s jinými rozvrhy.
- Dohodu lze výrazně urychlit využitím jednoduché webové služby, která umožní hlasování o termínu schůzky, případně i dalších problémech. Mezi takovéto nástroje patří např. Doodle, agreeAdate nebo NeedToMeet..
- Pokud je potřeba složitější dohadování, lze využít nějakého fóra nebo knihy návštěv, ale ne každý bude ochoten se někde registrovat. Na veřejně přístupném fóru však nelze řešit libovolný problém. Řešením může být umístění fóra nebo alespoň jednoduché knihy návštěv chráněné sdíleným heslem do prostoru na nějakém hostingu zdarma. Instalace může zabrat 10 min. a všichni se mohou rychle a efektivně dohodnout na první schůzce i na dalších případných problémech, které musejí být před touto schůzkou vyřešeny.

Doodle

- Zde vidíte ukázkou hlasování na Doodle. V záhlaví tabulky jsou nabídnuté termíny a na řádcích jména osob. Pro založení hlasování stačí zadat název události, termíny a svoje jméno a e-mail. Celé hlasování pak lze provést bez zadání jediného citlivého osobního údaje, protože stačí zadat pouze jméno či přezdívkou a vybrat si z možností. U každého

termínu můžete zvolit, zda je dobrý, přijatelný nebo nevhodný. Pod hlasování lze připsávat i textové komentáře.

- Lze nastavit i striktní hodnocení termínů (ano či ne), ale pak hlasující často raději vybírají „ne“ a může se stát, že žádný termín nebude vhodný.
- Je-li to možné, je třeba vypsát dostatečné množství termínů. Pro 5 lidí výběr ze 2 možností pravděpodobně nedopadne dobře, protože se obvykle najde alespoň 1 člověk, který si z nich nevybere.

Doodle

- Po tom, co již někdo hlasoval, je v zápatí zobrazené vyhodnocení pro jednotlivé termíny.

Doodle

- Ten, kdo hlasování založil, jej na závěr uzavře, při čemž dle výsledků hlasování zvolí nejvhodnější termín.

Příklad

- Nyní se vraťme k našemu příkladu. Všichni členové týmu mají stejné zadání a nikdo není předem určený jako vedoucí týmu. Začíná tedy jakási hra na nervy, kdy všichni čekají, až se někdo ozve. Ve výhodě můžou být ti členové týmu, kteří se znají, ale stále je pravděpodobné, že budou čekat, až se jim ozvou zbývající 2.
- Toto chování je fatální chyba, protože se nesmyslným čekáním ztrácí čas na práci na projektu. Komunikovat by měli začít všichni a ihned, protože to je jediný způsob, jak rychle začít. Každý by tedy měl poslat e-mail všem ostatním, ve kterém uvede detailnější kontaktní informace, nabídku termínů schůzky či odkaz na Doodle, případně odkaz na nabízený lepší komunikační kanál, např. neveřejné fórum, které spravuje. Následně by se měl využívat nejlepší nabídnutý komunikační kanál, případně pokud je více stejných, měl by se využít první nabídnutý.
- V našem příkladu máme členy týmu, které si pro anonymitu nazveme písmeny od začátku abecedy. Nejrychleji svůj e-mail poslal pan C, který všem zaslal svoje ICQ, telefonní číslo a časy, ve které by příští týden mohl přijít na schůzku do školy seřazené podle vhodnosti.
- Následně reagoval B, ale ten poslal jenom ICQ a jeden termín, který si ze seznamu pana C vybral. To je od něj sice hezké, ale jeden vybraný termín je k ničemu, protože ostatním se nemusí hodit. Správně měl poslat všechny termíny, ve které může přijít, nebo alespoň ty, ve kterých se shoduje s C.
- Pan A potom kromě jiného poslal i odkaz na Doodle a odkaz a heslo na zaheslovanou knihu návštěv, kterou někam instaloval. Pokud někdo z Vás přemýšlí, proč posílal časy i odkaz na Doodle, tak to není, protože je aktivní blbec, ale protože nechce zdržovat v případě, že ostatní budou posílat e-maily a Doodle z nějakého podivného důvodu nevyužijí. Navíc může být využita jiná služba než Doodle a může vyžadovat zadání nějakého osobního údaje, na které nemusí přistoupit každý.

Příklad

- Každý by měl nyní odpovědět na první přijatý e-mail a na každý další, na který je třeba reagovat.
- Když už je možnost přejít na lepší, tedy především přehlednější komunikační kanál, všichni by to měli udělat. Měli by tedy hlasovat na Doodle.
- Také by následně měli pravidelně kontrolovat knihu návštěv, jestli v ní nejsou nějaké nové příspěvky.
- Jistě si teď říkáte, proč se otravovat s forem či knihou návštěv, která není zdaleka nejlepším komunikačním prostředkem, když můžeme vše vyřešit hlasováním na Doodle a komentáři pod ním. Kdyby se všichni chovali jako pan C nebo A, byla by dohoda možná i přes e-maily a mohl by být téměř zbytečný i Doodle. Pokud se však najde někdo, kdo z nějakého důvodu na Doodle odmítne hlasovat nebo bude potřeba urychleně vyřešit nějaký problém, jehož řešení nechcete zveřejnit, a nejbližší termín schůzky bude až příští týden, nastanou problémy.

- Obvykle pak dojde na spoustu e-mailů, dohoda je pomalá a nepřehledná a přemýšlíte, kdo kterým e-mailem na co reagoval a co se vlastně dohodlo. Někdo nemusí být ochotný poslat všechny rozsahy časů ani je vybrat v hlasování, ale může vybrat pouze 2 termíny, obdobně jako náš pan B. Počet e-mailů a reakcí na ně potom narůstá a ne každý reaguje na poslední zasláný e-mail, takže v některých e-mailech není celá historie a ztrácí se souvislosti. Složitě řečeno, ale stále jednoduché proti tomu, co by Vás čekalo v té spoustě e-mailů.
- Jednoduchou knihu návštěv, která je na přibližně 200 řádků kódu a funguje i bez databáze, stáhnete a nainstalujete za 10 min. se základními znalostmi webu a poskytuje lepší ochranu informací, než pouhá neznalost adresy na konkrétní hlasování na Doodle. Může to tedy být použitelné záložní či doplňkové řešení, které umožní rychle zvládnout aktuální situaci.

První schůzka týmu

- První schůzka týmu by měla být vždy osobní. Přítomnost přes telekonferenci je stále částečně neosobní, což má negativní vliv na důvěru k ostatním členům týmu.
- Na schůzce je třeba dohodnout alespoň následující: Každý tým musí mít vedoucího. Jedná se o osobu, která zodpovídá především za dodržování termínů a popohání ty, kteří je nedodržují. Vedoucí týmu musí mít stále přehled o tom, kdo a na čem pracuje a zda vůbec někdo pracuje. Místo vedoucího není nějaký privilegovaný post, ale spíše velice nepříjemná práce, kdy někoho popoháníte, aby neměl problémy s termínem, a on Vám za to nadává. Měl by to být nejspolehlivější člen týmu, který je ochoten se této práci ujmout. Pokud nikdo nechce být vedoucím, je třeba losovat mezi spolehlivými členy týmu.
- Po volbě vedoucího je třeba dohodnout stabilní komunikační kanály, pravidla pro komunikaci a prostředky pro sdílení dat v týmu. Těmito se budeme zabývat v následujících částech této přednášky.
- Nakonec lze předběžně rozvrhnout práce na projektu, rozdělit počáteční úkoly apod. Bude záležet především na tom, kolik bude na schůzku času a jak rychle se podaří dohodnout na základních věcech.

Komunikační kanály

- Mezi základní nejčastěji využívané komunikační kanály patří telefon, e-mail, instant messaging, diskusní fóra, chaty, ale také projektová wiki a další webové služby.

Telefon

- Nejrychlejším kanálem je obvykle mobilní telefon. Ne každý je však ochoten poskytnout ostatním členům týmu své číslo, protože je nezná. Je to určitá prevence před obtěžováním, která však ihned na začátku zavádí do týmu nedůvěru a nepříjemnou atmosféru. Úspěch projektu je přitom závislý především na dobré týmové spolupráci.
- Motivací ke sdělení telefonního čísla může být i to, že kolegové Vás mohou upozornit na nějakou náhlou událost či na to, že zapomenete na nějaký důležitý termín.
- Už jsem byl několikrát v situaci, kdy se na někoho čekalo kvůli tomu, že si spletl nebo špatně zapamatoval termín. Dokonce i u státnic se nejednou stalo, že měl někdo za 15 min. přijít na řadu a stále nebyl ve škole. Obvykle se jednalo o posledního člověka, který počítal svůj čas od konce, na kterém je rezerva, místo od začátku. Kdybychom neměli k dispozici jeho telefonní číslo, jeho pozdní příchod pro něj mohl mít fatální následky (k tomuto je nutno poznamenat, že dnes již telefonní čísla studentů v IS FIT nejsou dostupná kvůli ochraně osobních údajů, takže takto už nikomu nepomůžeme).
- Využití telefonu by však mělo být minimální a měl by se využívat pouze pro neodkladné záležitosti. Neobtěžovat ostatní častými a zbytečnými telefonáty patří k ohleduplnému a serióznímu jednání člena týmu.

E-mail

- E-mail je pravděpodobně nejdostupnější komunikační prostředek. Bohužel je však také jedním z nejpomalejších. I přesto, že mají obě komunikující osoby spuštěného poštovního klienta, může výměna e-mailů trvat i desítky minut. Pokud je někdo zvyklý vybírat poštu pouze ráno nebo náhodně 2x týdně, může být doba čekání na odpověď nepřijatelná.

- Proto je u důležitých zpráv výhodné využívat doručenkou, která umožní zjistit, zda už daná osoba e-mail přijala. Zaslání doručenkou lze sice přeskočit, nicméně při týmové spolupráci je toto jednání zcela nevhodné.
- Pokud řešíme týmový projekt, je navíc vhodné zasílat každou zprávu v kopii všem ostatním členům týmu a při odpovídání využívat funkci „Odpovědět všem“, která zašle kopii odpovědi i příjemcům kopie původní zprávy. Předejdeme tím zdržování a různému přeposílání zpráv se slovy „tenhle řekl tohle a tenhle tamto“. Ve větších projektech se využívají mailing listy, což jsou seznamy příjemců a specializované nástroje, které s nimi pracují (např. <http://www.gnu.org/software/mailman/index.html>). Když pak pošlete e-mail na speciální adresu, následně jej dostanou všechny ostatní osoby na seznamu a nemůže se stát, že na někoho zapomenete.
- Musím zde upozornit také na to, že reakci na e-mail vždy píšeme nad citovaný text e-mailu, na který reagujeme, nebo daný e-mail rozčleníme a na jednotlivá témata reagujeme přímo pod nimi. Pokud svoji reakci připsáte na konec, autor původního e-mailu musí skrolovat přes svůj text až na konec k Vaší odpovědi. Pokud si vyměníte několik e-mailů a oba budete psát na konec, budete se zbytečně otravovat skrolováním. Pokud byste se chtěli skrolování vyhnout tím, že vymažete původní text e-mailu, je to ještě mnohem horší, protože je třeba hledat, čeho se vlastně reakce týká. Zachování celé historie odpovědí je tedy také součástí dobrých zvyklostí.

Instant messaging

- Existují různé sítě, jako ICQ, Jabber, AIM apod. Do různých sítí se lze připojovat různými klienty a některé sítě navíc umí přenos do jiných sítí. Pokud obě komunikující osoby nevyužívají stejnou síť a stejného klienta, mohou nastávat různé problémy. Autorizace se nemusí zdařit, může docházet k její ztrátě a nutnosti opakování. Může to také vypadat, že je někdo offline, ale ve skutečnosti má jen rozdílného klienta, který jinak vysílá status. Místo smajlíka se zobrazuje nesrozumitelný text, přenosy souborů často nefungují apod.
- Některé sítě či klienti nepodporují chat 3 a více osob současně a při chatu s různými klienty mohou nastávat různé problémy s vypadáváním osob z místnosti apod. Jednou jsme při řešení projektu diskutovali a když jsem se jednoho z členů týmu zeptal na jeho názor, zjistil jsem, že zprávy z chatu mu už asi 5 minut vůbec nechodí. Jenom mu bylo divné, že tak dlouho nikdo nic nepíše. Instant messaging lze tedy využít spíše při komunikaci 2 osob, které společně řeší nějaký podúkol, nebo při dotazech na konkrétní osobu, které se netýkají celého projektu. V těchto případech má velké výhody v rychlosti a jednoduchosti.
- Pokud je tímto kanálem uskutečněno nějaké důležitější rozhodnutí, je vhodné si jej poznamenat a zpřístupnit ostatním přes jiný komunikační kanál, např. fórum nebo e-mail. Obvyklým problémem totiž je, že historie komunikace se ukládá pouze do klienta, který toto podporuje a je pro uchování záznamů komunikace nastaven. Potom někdo přijde k jinému PC a k potřebným informacím se nedostane.

VoIP / Skype / Google Hangouts

- VoIP, Skype či Google Hangouts nám umožňuje telefonovat přes Internet. Kromě standardních hovorů jsou zde i možnosti videohovorů, audiokonferencí s více účastníky a videokonferencí.
- Není-li možná osobní schůzka, lze ji částečně nahradit i schůzkou přes Skype. První schůzka týmu by však měla být vždy osobní.
- Nevýhodou však je, že některý člen týmu nemusí mít potřebné vybavení, tedy mikrofon, sluchátka, webkameru a dostatečně rychlé připojení k Internetu, nebo jej má pouze doma či pouze v práci. Zde bych chtěl upozornit na to, jak strašně nepříjemné a otravné je, když má někdo mikrofon a reproduktory, tedy nepoužívá náhlavní soupravu se sluchátky. Daný člověk má sice pohodlí a neslyší žádný problém, ale ten, kdo je na druhé straně, slyší svoji ozvěnu, což jej po chvíli totálně otráví. Pokud nepoužívá náhlavní soupravu žádný z účastníků hovoru, přes ozvěny a pískání nelze vůbec hovořit. Využití náhlavní soupravy tedy jednoznačně patří k ohleduplnému chování. Je-li u jednoho PC více lidí, nebo z jiného

důvodu nelze využít náhlavní soupravu, je potřeba ve chvílích, kdy nemluvíte, ztlumit mikrofon. Ušetří se tím i šířka pásma pro přenos, což může u více účastníků na pomalejších linkách znamenat opravdu znatelné zlepšení. Ztlumené mikrofony však zpomalují reakční dobu a musíte tedy počítat s větší délkou hovoru. Také je potřeba dávat ostatním prostor a více času na reakci.

- Velmi důležité je dělat si poznámky. Je třeba si vypsát důležitá rozhodnutí, úkoly a další potřebné informace. Po 2 hodinách dohadování už si jinak na polovinu dohodnutých věcí nevzpomenete a záznam se obvykle nepořizuje. I když však záznam budete mít, není v něm možné jednoduše vyhledávat a jeho kompletní procházení je časově náročné. Málo kdo má navíc nervy na to se 2 hodiny poslouchat ze záznamu.

Diskusní fóra

- Diskusní fóra jsou asi nejlepším prostředkem pro komunikaci v týmu. Historie komunikace se zaznamenává a informace jsou odkudkoliv přístupné všem členům týmu. Fórum je sice pomalejší než ICQ, ale může být rychlejší než e-mail (pokud jej členové týmu sledují). Některá fóra navíc umožňují upozorňování na důležité zprávy e-mailem či přes mobil.
- Je však velmi důležité zachovat přehlednost. K tomuto účelu je vhodné využití vláken a moderátora, který dohlíží na dodržování stanovených pravidel pro komunikaci na fóru. Nejsou-li k dispozici vlákna, řešení více problémů se prolíná a rychle se ztrácí kontext a přehled. Není-li fórum moderované, porušení pravidel může snížit přehlednost fóra např. rozptěněním jednoho problému do více vláken apod.
- Na fóru je také nutné udržet emoce, vyjádřit svůj názor, respektovat názor druhého, ale nezačít hádku. Hádky, která již nepřinášejí žádné konstruktivní připomínky, odsunuje do historie a zatemňuje původně řešený problém. Pokud se 2 uživatelé pohádají, zbylých 5 členů týmu už nemusí být schopných zjistit, co se vlastně řešilo. Při hádce by měl vždy zasáhnout moderátor.
- Pro práci na projektu byste měli vždy využít neveřejné fórum. Využijete-li veřejné fórum, může jej např. pomocí nějakého vyhledávače najít konkurence a využít Vaše řešení, které zde nalezne přehledně s výhodami i nevýhodami.

Projektová wiki

- Projektová wiki je jakousi encyklopedií projektu. Lze v ní shromáždit veškeré informace o projektu a následně z ní čerpat informace při tvorbě dokumentace i zkušenosti s řešenými problémy pro práci na jiných projektech.
- Každý projekt by měl mít hlavní stránku, která bude obsahovat základní informace, jako je název, řešitelé, základní informace o zadání a cílech projektu, informace o celkovém stavu projektu apod.
- Na dalších stránkách by potom měly být detailní informace o jednotlivých řešených problémech, podproblémech, dosažených cílech, vytvořených součástech programu apod.
- Vše by mělo mít dokumentační charakter, tzn. že by se z wiki stránky nemělo stát diskusní fórum, ve kterém se důležité informace ztratí v nerelevantních otázkách a odpovědích.

Příklad wiki stránky bakalářské práce

- Jako ukázkou wiki stránky projektu jsem zvolil stránku jedné zdařilé bakalářské práce. Jak vidíte, na začátku jsou nejdůležitější informace, následuje cíl práce, úvod do řešené problematiky, informace o řešení a výsledky.

Další webové služby

- Existuje celá řada dalších webových nástrojů pro komunikaci, z nichž některé jsou i velmi sofistikované a specializované na týmovou spolupráci.
- Některé nástroje jsou však pouhou kombinací výše uvedených (případně s nějakými vylepšeními). Za zmínku stojí např. Facebook, který kombinuje fórum, chat, úložiště souborů a další jednoduché služby. Pokud jsou všechny nástroje v jednom, může to být velká výhoda. Nicméně v případě Facebooku je méně pohodlné a přehledné procházení historie, což může být značná nevýhoda.

- Nevýhodou je, že na využití takového nástroje se musejí dohodnout všichni členové týmu a musejí se s ním seznámit. Seznámení s takovým nástrojem a jeho efektivním využitím také přináší určitou režii, která zkracuje čas na samotnou práci na vytvářeném produktu.

Web projektu

- U projektů, u kterých je potřeba informovat veřejnost a sponzory, je nutné vytvořit web projektu. Můžeme jej však vytvořit u jakéhokoliv projektu, i když pro příliš malé projekty se jeho tvorba nemusí vyplatit.
- Web projektu však nemusí být pouze stránky s informacemi pro veřejnost. Můžeme vytvořit celý portál, na kterém lze agregovat webové služby pro podporu práce na projektu. Může zde být diskusní fórum, chat, úložiště dokumentů apod. Lze vytvořit i sekce pro zadávání pracovních výkazů a sledování stavu projektu.
- Vždy je však třeba pečlivě zvažovat, které informace budou veřejně přístupné a ke kterým se dostanou pouze přihlášení členové týmu, případně sponzoři či jiní uživatelé.

Příklad

- K webu projektu zde mám obrázek, jak vypadal web jednoho z projektů řešených v magisterském studijním programu. Kromě veřejných informací obsahoval chat, diskusní fórum, ankety pro hlasování o problémech, úložiště dokumentů a pracovní výkazy. Jeho značnou nevýhodou bylo, že diskusní fórum nepodporovalo vlákna.
- Na profesionální web projektu se můžete podívat např. na <http://decipher-research.eu/>

Nástroje

- Web projektu samozřejmě nemusíme psát manuálně pro každý projekt. Jednoduché fórum, chat a úložiště dokumentů lze napsat rychle (web, který jste viděli na předchozím slajdu jsem vytvořil za cca 40 hod.). S větším množstvím integrovaných služeb by však rozsah prací na webu projektu mohl překonat rozsah práce na samotném produktu.
- Pro vytvoření webu projektu máme k dispozici hotové webové aplikace, které integrují fóra, wiki, systém pro správu verzí, plánovač a další potřebné nástroje.
- Mezi tyto aplikace patří např. Redmine a Trac. Oba tyto systémy jsou vyvíjeny s otevřeným zdrojovým kódem a můžete si je tedy stáhnout, vyzkoušet, případně i začít využívat na školních projektech, i když na školní projekt může být takový nástroj jako kanón na vrabce.

Pravidla pro komunikaci

- V týmu je také třeba stanovit určitá pravidla pro komunikaci, která budou všichni členové dodržovat. Tato pravidla zahrnují především to, které komunikační kanály budou využívány k jakým účelům a minimální frekvence využívání těchto kanálů. Je třeba stanovit minimální frekvenci kontroly pošty a prohlížení diskusního fóra, např. 1x denně v pracovní dny a pokud je využíván instant messaging, tak i nějaké minimální časové rozsahy, kdy budou všichni dostupní online.

Příklad

- Nyní se vraťme k našemu příkladu. Protože A, B a C jsou kamarádi, jako vedoucí by měl být zvolen jeden z nich, protože je menší pravděpodobnost, že projekt vzdá a kamarády v tom nechá. Nikdo z nich však o vedení nemá výraznější zájem. Protože A na začátku nabídl knihu návštěv, B a C ho zvolí vedoucím. Pokud by vedení nepřijal, mohli by losovat.
- A nabídl, že vytvoří web projektu a umístí na něj chat a fórum. Ostatní souhlasili a následně se dohodli na těchto pravidlech:
- Hlavním komunikačním kanálem bude fórum na webu projektu. Každý jej navštíví nejméně 1x za každý pracovní den, přičemž se stejnou frekvencí se bude vybírat i pošta. Moderátorem fóra bude vedoucí projektu.
- Pro elektronické diskuse mezi více členy současně bude využíván chat na webu projektu, termíny diskusí budou domlouvány na fóru.
- Pro komunikaci 2 osob pracujících na podúkolech bude využito ICQ. Každý člen týmu, který zrovna pracuje na nějakém úkolu, by měl být při práci online.

- Telefon se bude využívat pouze výjimečně. Pouhé prozvonění telefonu je považováno za výzvu k příchodu na ICQ či návštěvě fóra.
- Pokud někdo bude nemocný, nebo z jiného důvodu nebude moci daná pravidla dodržet, na fóru se omluví a uvede, proč a do kdy nebude komunikovat, případně proč nějakou dobu nekomunikoval. Pokud je to možné, omluva by měla být zveřejněna předem. Pokud by někdo byl v nemocnici a hrozilo, že se již projektu nezúčastní, měl by se ozvat vedoucím, např. prostřednictvím telefonu či vzkazu přes třetí osobu.
- Jistě Vám neuniklo, že zde není uveden Skype ani projektová wiki. Je to proto, že projektová wiki pro takovýto projekt není potřeba a pánové D a E řekli, že Skype nevyužívají.
- Všimněte si také nutnosti omluvy. Některé z Vás jistě napadlo, že pravidlo je příliš přísné a nikomu není nic do toho, proč jste nemohli komunikovat. Za jeden den nepřítomnosti se samozřejmě není třeba omlouvat, ale pokud někdo jeden nebo dva týdny nekomunikuje, může to znamenat, že projekt vzdal, nebo že mu na něm nezáleží a moc času mu nevěnuje. Při práci na jednom projektu se mi stalo, že jeden kolega řešil svůj podúkol 4 týdny s tím, že po prvních dvou týdnech teprve odpověděl na e-mail s tím, že na tom pracuje a polovinu už má hotovou. Čtvrtý týden, když už se opět dlouho neozýval, jsem mu telefonovala a s hrůzou jsem zjistil, že celou dobu nedělal nic a projekt vzdal.

Obsah - Prostředky pro sdílení dat v týmu

- Od komunikace, která se zaměřuje na domlouvání a výměnu zpráv, přejdeme ke sdílení dat v týmu. Data jsou vytvářené dokumenty, zdrojové texty programů, testovací data a další soubory, které jsou při práci na projektu potřebné.

Prostředky pro sdílení dat v týmu

- Pomineme-li sdílení dat přenášením po fyzických nosičích, které je už spíše nereálné, a různé peer-to-peer sítě a další řešení, která nejsou k tomuto účelu nejběžnější, máme 3 různé možnosti pro sdílení dat v týmu. FTP, využití webových služeb a systémy pro správu verzí.

FTP

- FTP je zcela jistě nejdostupnějším řešením. Stačí založit účet na nějakém hostingu zdarma, který nechce příliš mnoho osobních údajů, rozeslat přihlašovací údaje a výměna dat může začít.
- Pokud však pomocí FTP sdílíme něco, co se v průběhu práce na projektu mění, brzy nastanou problémy. Ztrácíme totiž předchozí verze souborů a pokud někdo něco pokazí, již nemusí být cesta zpět. Navíc si 2 členové týmu mohou stáhnout soubor, jeden z nich jej upraví a aktualizuje a potom jej aktualizuje druhý, čímž zahodí změny od toho prvního. FTP také není vhodné pro souběžný přístup, takže pokud budou zapisovat 2 osoby současně, výsledek může být různý.
- I přes všechny nevýhody lze FTP při sdílení dat využít. Hodí se především pro soubory, které jednorázově někdo vytvoří a dále už je ostatní jenom stahují, jako např. zápisy a záznamy ze schůzek, sady testovacích dat apod. Vzhledem k efektivnímu způsobu přenosu je vhodné i pro velké soubory a výhodou je i možnost stažení souboru přes HTTP, tzn. možnost někomu zaslat odkaz, který si rychle bez hledání otevře v prohlížeči.
- Možnost přístupu přes HTTP je však současně i velkou nevýhodou. Pokud si u souborů nenastavíte vhodná oprávnění, můžete veřejně zpřístupnit i utajované informace, které se mohou hodit konkurenci. To, že konkurence nezná adresu, nemusí být překážkou, protože pokud daný web nemá žádnou indexovou stránku, webserver může automaticky zobrazovat seznam souborů, který následně zaindexuje vyhledávač.

Web

- Na webu lze nalézt řadu služeb pro sdílení souborů. Soubor si uložíte a rozesíláte odkazy, nebo využijete nějaké sdílené úložiště, které funguje jako FTP přes web. Takové úložiště lze

vytvořit i na webu projektu.

- Sice existují i sofistikovanější řešení, ale obvykle se skutečně jedná o nějakou dobu FTP přes web. Z toho plynou i stejné nevýhody jako u FTP, ke kterým se navíc často přidává výrazné omezení velikosti souboru.

Příklad

- Zde mám opět příklad z reálného projektu. Jedná se o velice jednoduché úložiště, které však pro daný účel bylo dostačující. Všimněte si odkazů pro vymazání souboru. Nepříjemná situace může nastat ve chvíli, kdy někdo použije stahovač na všechny odkazy se záměrem stažení celého obsahu úložiště, aniž by přemýšlel o tom, že jsou tam i odkazy pro smazání souboru. Tuto situaci jsem v daném projektu zažil a viníka usvědčil až denní záznam z logu webserveru.

Systém pro správu verzí

- FTP ani jednoduchá webová služba tedy nejsou vhodné ke sdílení souborů, které se mění, a tím ani ke sdílení zdrojových textů programu. K tomuto účelu existuje mnohem lepší a sofistikovanější nástroj, kterým je systém pro správu verzí.
- Tento systém umožňuje vyhledání libovolné předchozí verze souboru, prohlížení provedených změn, řešení konfliktů apod.
- Není však vhodný pro velké objemy dat, u kterých není potřeba uchovávat předchozí verze. Pokud do adresáře se zdrojovými texty programu uložíte 1000 souborů s 2 GB testovacích dat, kolega, který je zrovna připojený přes mobil, kde platí za přenesená data, nejprve začne aktualizovat svoji verzi a stahovat. Za chvíli zjistí, co se děje, a začne nadávat. Potom bude muset vybírat, co si chce aktualizovat a co ne, aby si tu hrůzu nemusel stahovat.
- Některé systémy pro správu verzí navíc s velkými soubory neumějí pracovat, takže uložení souboru se nemusí zdařit a celý systém pro správu verzí může zkolabovat.
- Těmito systémy se bude zabývat i celá příští přednáška.

Další služby pro sdílení souborů

- Existují i další služby, které lze pro sdílení souborů využít. Mezi tyto patří např. Dropbox, Google Drive nebo TeamDrive. Obdobně jako systém pro správu verzí umějí uchovat předchozí verze souborů apod., nicméně pro sdílení zdrojových textů programu obvykle mají oproti systému pro správu verzí různé nevýhody – např. obtížnější získání předchozí verze souboru či prohlížení změn, chybí popisy změn apod. Uchování historie také může být omezené, případně se za její delší uchování více platí.

Příklad

- Opět se vraťme k týmu z našeho příkladu. Pro sdílení dat si zvolili FTP, přičemž využijí adresář na serveru s webem projektu, a systém pro správu verzí. Jediné problémy byly, který systém pro správu verzí zvolit a kdo jej nainstaluje. První problém bude rozebrán v příští přednášce a druhý bude záležet na možnostech a prostředcích členů týmu. V nejhorším případě lze zjistit, jaké prostředky jsou dostupné ve škole, což udělá vylosovaný člen týmu.

Časové rozložení práce na projektu

- U každého projektu máme obvykle daný termín, do kterého jej musíme dokončit nebo dosáhnout určitého dílčího cíle. Předběžné rozvržení prací na projektu nám umožní získat přehled o tom, kolik času máme na kterou činnost a jak intenzivně je třeba pracovat.
- Obvyklé je, že v prvních fázích projektu jde práce pomaleji a ke konci není dostatek času. Ve všech fázích je proto třeba uvažovat i rezervy, přičemž u prvních fází projektu by tyto rezervy měly být menší a u posledních větší. Při plánování rezervy je třeba zohlednit i složitost dané fáze projektu.
- Samotné plánování sice u malých projektů můžeme vyřešit dohodou nebo na papíře, ale u větších projektů je již třeba využít nějaké podpůrné prostředky. Jedním z nejčastěji využívaných programů pro podporu plánování prací na projektu je Microsoft Project.
- Ať už však plán vytvoříte jakkoliv, není to něco, co se odloží bokem a začne se pracovat.

Plán je potřeba průběžně sledovat a v případě potřeby (např. když už je jasné, že je nereálný) jej upravit. Častou chybou je také to, že plán sleduje pouze vedoucí. Členové týmu potom nevidí, kdy mají začít pracovat a kdy mají mít práci hotovou. Když pak někomu vedoucí pošle e-mail, že už je v polovině času na svou práci, a daný člověk teprve z tohoto e-mailu zjistí, že už měl dávno pracovat, už může být pozdě. Plán by proto měl být vystavený někde, kde jej jednotliví členové týmu vidí, a každý by měl sledovat jeho aktuální stav. U větších projektů jsou pak kromě plánu projektu i plány jednotlivých fází či menších částí a řešitelé mohou mít přístup pouze k částem, které se jich týkají. Využívají se také plánovací kalendáře vygenerované z plánu projektu apod..

- Velmi důležité je také to, aby byl plán ve všech částech dostatečně podrobný. Pokud plán nevytváří specialista, který má s touto činností zkušenosti, je dobré, když plán vytvářejí a případně průběžně upřesňují všichni členové týmu. Lze tak předejít situaci, kdy plán vytvoří vedoucí, který si svoji práci, které dobře rozumí, rozvrhne podrobně a ostatním jen velmi zhruba. Pro ostatní členy týmu je pak obtížnější sledovat stav práce na jejich částech projektu i stav práce na těch částech, na které svou práci navazují.
- Nyní Vám ukážu pouze velmi krátkou ukázkou využití Microsoft Project a pokud by Vás tato problematika zaujala, můžete si zapsat kurs řízení projektů informačních systémů, ve kterém se probírá detailně. Nebude-li tento kurs otevřen, můžete si v navazujícím magisterském studiu zapsat kurs management projektů.

MS Project

- Zde vidíte vytváření nového projektu. Vytvoříte soubor MS Project, vložíte projekt mezi úkoly, aby se stal kořenem stromu úkolů, a nastavíte, od kdy se bude plánovat.

Ganttův diagram

- Na tomto obrázku je Ganttův diagram, který je základním diagramem pro plánování projektů. Osa x je časová osa a na ose y jsou jednotlivé úkoly. V diagramu vidíte termíny, dobu trvání jednotlivých úkolů, vztahy mezi úkoly, přiřazené osoby a prostředky apod. Kromě úkolů jsou v diagramu i milníky, což jsou body v čase a mají tedy nulovou dobu trvání. Jsou to okamžiky, ve kterých se má zkontrolovat splnění nějaké podmínky, dokončení úkolu nebo fáze projektu.

Kalendář

- K dispozici je také plánovací kalendář.

Seznam zdrojů

- Pokud je někdo přetížený, zjistíte to i v seznamu zdrojů. Kromě lidských zdrojů mohou být také materiální a investiční. Lidské zdroje potom mohou být abstraktní či konkrétní. Abstraktní zdroj může být skupina lidí určitého zaměření.

Diagram zdrojů

- V diagramu zdrojů vidíte vytížení jednotlivých zdrojů v jednotlivých dnech. Přehledně zde vidíte přesčasy a chybné přiřazení úkolů, kdy má někdo pracovat na 200 a více procent.

Sledování stavu projektu

- MS Project umožňuje také sledování stavu projektu. V základním zobrazení vidíte, které úkoly jsou splněny.

Sledovací Ganttův diagram

- Při sledování projektu obvykle nestačí vědět, které úkoly jsou splněné, ale je potřeba vědět, v jakém stavu jednotlivé úkoly jsou. K tomu slouží sledovací Ganttův diagram.

Rozdělování práce a zodpovědností v týmu

- Ve firmě je rozdělení práce často dané managementem nebo znalostmi jednotlivých členů týmu. Tým má obvykle určité jádro, což je skupina dobře spolupracujících lidí s aktivním přístupem a ostatní členy, kteří byli k týmu přiděleni a jejich přístup k práci není předem známý.
- Máte-li možnost si práci rozdělit mezi sebou a jste-li v jádře týmu, vždy je dobré aby práce

na prvních fázích projektu dostali členové z jádra týmu. I když se může jednat o jednodušší podúkoly, začátek, kdy ještě nic není hotové, bývá nejtěžší a slabšímu členovi týmu může trvat dlouho. Rychlý začátek projektu je důležitý, aby se nenabíral skluz do dalších fází projektu.

- Pokud slabší člen týmu svoji práci neudělá, někdo z jádra týmu ji obvykle musí udělat za něj. Pokud se jedná o pozdější fázi projektu, kdy už má daný člen týmu svoji práci hotovou a v projektu není časový skluz, je to zvládnutelnější, než když má někdo dělat svoji práci a současně práci z předchozí fáze, na kterou měl navazovat.

Plánování problémů

- Při práci na projektu mohou nastat různé problémy: někdo onemocní, dá výpověď, nebude se mu chtít do práce a bude zdržovat apod. Problémy mohou být i technického charakteru, kdy někdo nebude mít na čem pracovat nebo dojde ke ztrátě dat.
- Na tyto problémy je třeba být připraven a počítat s nimi již od prvotního plánování prací na projektu. V této fázi se však zatím nejedná o problémy, ale o pouhá rizika.
- Rizika navíc v průběhu projektu vznikají a zanikají, takže nestačí je pouze jednou naplánovat, ale je třeba s nimi stále pracovat. Touto problematikou se zabývá disciplína nazvaná řízení rizik, kterou se můžete naučit v rámci volitelného kurzu řízení projektů informačních systémů.

Jednoduché řešení pro malý tým a projekt

- Pokud neumíte s riziky pracovat, nebo se to v rámci malého projektu pro 4 – 6 lidí nevyplatí, můžete si pouze vytvořit obecné časové rezervy. Rezervu vytvoříte např. tak, že na začátku některý spolehlivý člen týmu nedostane žádnou práci. Ostatní tak mají více práce a daný člen týmu může pomáhat tomu, kdo to zrovna potřebuje. Pokud někdo onemocní, automaticky převezme jeho práci (pokud neonemocní sám).
- Také je výhodné počítat rezervu na celý projekt a při plánování stanovit termín dokončení před oficiálním termínem.
- Pokud se v některé fázi projektu dostanete do předstihu, je dobré posunout následující termíny o tento předstih a zvětšit tak rezervu na konci projektu. Pokud se další fáze zpozdí, můžete termíny posunout zpět na výchozí. Takto lze korigovat chyby v plánování rezerv na jednotlivé fáze projektu.

Shrnutí

- Několik hlavních myšlenek z této prezentace:
- Dobré zvyklosti při psaní komentářů nám umožní vygenerovat programovou dokumentaci ze zdrojových textů.
- Dobrá a efektivní komunikace v týmu je základem týmové spolupráce.
- Pro sdílení dat v týmu je třeba zvolit vhodné prostředky a využívat je k vhodným účelům.