

SOFTVÉROVÝ NÁSTROJ AKO SPOJENEC PRI SLEDOVANÍ VYKONANEJ ČINNOSTI

*Anarchia - želaná forma medziľudských vzťahov
založená na dobrovoľnej spolupráci bez toho, aby bola
vynucovaná alebo riadená inými.
(definícia podľa Wikipédie)*

Anton Balucha

Slovenská technická univerzita
Fakulta informatiky a informačných technológií
Ilkovičova 3, 842 16 Bratislava
xbaluchaa[zavináč]is.stuba[.]sk

Abstrakt. *Vývoj softvérových aplikácií prebieha v dnešných časoch najmä v tíme a hlavne vďaka tímovej práci. Doby, keď bol jednotlivec zodpovedný za celú aplikáciu a vývoj softvéru sa pomaly strácajú. Manažment jednotlivca bol pomerne jednoduchý, neboli potrebné rôzne pokročilé riešenia evidencie stavu práce, evidencie stavu projektu. Jednotlivec mal prehľad o všetkom. Čo však, keď máme tím? Ako budeme sledovať jeho prácu? Ako zistíme, kde stojíme s prácou? Aké nástroje nám môžu pomôcť, aby sme vedeli prácu správne zdeliť a neplytvali časom a schopnosťami jednotlivých členov tímu? Aké vlastnosti by mala aplikácia spĺňať, aby bolo možné rozdeliť prácu efektívne? Na tieto otázky sa pokúsim zodpovedať v tejto eseji.*

Kľúčové slová: *sledovanie úloh, vlastnosti aplikácie, kontrola ľudí*

Prečo sa venovať výberu správneho nástroja?

„Tím, tím, tím, tím. Už len to slovo samotné znie dobre.“ Tak hovorí Reynhol zo seriálu The IT Crowd. A podľa mňa sa za ním skrýva ešte viac. Vývoj a tvorba softvérových a informačných systémov v dnešnej dobe je možná práve vďaka tímu. Práve vďaka tejto skupine ľudí, ktorá dokáže skoncentrovať svoje úsilie na vytvorenie jedného výsledného produktu, ktorý naplňa ich cieľ. Cieľ, ktorý ich môže posúvať ďalej v oblasti prezentácie

vlastnej osobnosti a svojich schopností, v oblasti nápadov, ktoré prinesú svetu niečo nové, revolučné, ale hlavne naplnia svojou tvorivou činnosťou svoje peňaženky a bankové účty. Preč sú časy, keď o aplikácií, o spôsobe práce na projekte, vedení a organizovaní práce na projekte rozhodoval jednotliviec. Dnes, už dokonca aj na trochu väčších webových stránkach spolupracuje viacej ľudí.

Už kedysi, keď po zemi chodili mamuty, ľudia pracovali v tíme. Museli si spolu naplánovať lov, pripraviť si náradie, skoordinať sa a využiť schopnosti jednotlivých členov tímu na úspešné ulovenie nedeľného obeda.

Môžeme si tento príklad s lovením mamutov zobrať ako paralelu na proces tvorby softvérových a informačných projektov. Dnes si taktiež ľudia musia naplánovať svoju prácu, spôsob jej vykonania, vhodne sa zosynchronizovať a využiť svoje vedomosti a potenciál na úspešné ulovenie nedeľného obeda. A práve k tomu využívajú rozličné podporné prostriedky.

Spomínanými podpornými prostriedkami v minulosti bola buď veľmi veľká sila alebo správne nabrúsený oštep. S využitím spomínanej paralely aj dnes, rozličnými podpornými prostriedkami pre úspešné riadenie tímu a využitie ich schopností môže byť ako hrubá sila veľa kávy na zvýšenie koncentrácie a veľa ceruziek a papiera na ručnú tvorbu analýz a sledovanie tímu. Alebo ním môže byť rozumne zvolený nástroj, ktorého správne použitie nám pomôže efektívne skoordinať prácu v tíme a využiť jeho potenciál naplno.

V tejto práci sa teda venujem prostriedkom a ich vlastnostiam, ktoré dokážu sledovať jednotlivých členov tímu, ich postup v práci a pomocou ktorých dokážem vhodne skoordinať a presúvať ľudí ako figúrky na šachovnici projektu tak, aby sme naplno využili ich potenciál a schopnosti. Budeme sa pýtať, či je vhodné použiť daný nástroj, aký má prínos pre človeka, ktorý rozhoduje o tom, ako budú jeho ľudia na projekte ďalej pracovať.

Chceme sledovať. Čo potrebujeme?

Keď chceme sledovať ľudí, monitorovať ich činnosť, musíme sa spýtať, aké funkcie by predstavovaná aplikácia mala ponúkať. Musíme zistiť, aké vlastnosti je potrebné sledovať, aby sme boli schopní monitorovať tím a jednotlivých členov tímu.

Jednotlivé potrebné vlastnosti systému by sa dali podľa mňa rozdeliť do štyroch základných ucelených kategórií:

- zaznamenávanie úloh
- zaznamenávanie času
- identifikácia členov tímu
- iné vlastnosti projektu

Všetky rozličné iné vlastnosti a kategórie dokážeme vytvoriť a vygenerovať na základe týchto štyroch spomínaných kategórií.

Prvú kategóriu som určil ako zaznamenávanie úloh. Je to evidencia úloh, ktorým okrem samotného názvu a opisu úlohy pridávame aj iné rozširujúce údaje. Patrí sem napríklad priorita, dátum vzniku úlohy, vyhradený čas, autor úlohy, prípadne človek priradený na jej vykonávanie. Úlohám môžeme priradiť poznámky, môžeme ich hodnotiť.

Druhú kategóriu, zaznamenávanie času, tvorí evidencia času v rozličnej forme. Napríklad dátum a čas vytvorenia úlohy, dĺžka času potrebného na vykonanie úlohy, dĺžka času stráveného nad úlohou, dátum a čas ukončenia úlohy. Tieto časy sa potom priradujú jednotlivým úlohám.

Identifikácia členov tímu tvorí zoznam členov tímu, ich kontaktné údaje, e-mailové adresy. Jednotliví členovia tímu sa prihlasujú do aplikácie, pracujú s ňou a značia úlohy týkajúce sa projektu.

Iné vlastnosti projektu sú hlavne doplnkové vlastnosti, ktoré sa nie priamo dotýkajú predchádzajúcich kategórií a rozširujú funkcionality aplikácie. Ide napríklad o definovanie ceny práce, rozpočtu projektu. V rámci komunikácie s tímom sem zaraďujem aj diskusné fóra, posielanie správ prípadne súborové úložisko.

Ostatné vlastnosti a funkcionality sa zakladajú práve na týchto štyroch kategóriách. Napríklad generovanie grafov postupu práce na projekte vychádza z existujúcich zadaných údajov.

Rôznorodosť nástrojov

Podporné prostriedky pre riadenie softvérového projektu so zameraním sa na sledovanie úloh a využitie ľudských zdrojov je možné rozdeliť na niekoľko kategórií podľa toho, z akého uhlu pohľadu sa pozeráme na ich triedenie. Tieto pohľady nám umožňujú výber vhodných aplikácií pre potreby riadenia ľudí.

Triedenie z pohľadu nasadenia aplikácie

Z hľadiska nasadenia som rozdelil tieto podporné prostriedky rozdeliť do troch kategórií:

- jednopoužívateľské aplikácie
- klient – server aplikácie
- webové aplikácie

Jednopoužívateľské aplikácie predstavujú aplikácie nainštalované a používané jednou osobou na jednom počítači. Predstaviteľom takéhoto produktu je napríklad Microsoft Project.

Práve najväčšou nevýhodou jednopoužívateľskej aplikácie je obmedzené pôsobisko na jeden počítač, čo znamená vo väčšine prípadov samozrejme aj jedného používateľa. Toto riešenie však v rámci tímovej spolupráce, keď by všetci členovia zdieľali jeden počítač, nie je efektívne a ani prípustné. Hoci aplikácia môže mať prepracovanú správu dát o projekte, jej hodnotu znižuje práve jej využiteľnosť a prístupnosť.

Aplikácie typu klient – server predstavujú súbor dvoch aplikácií. Na serveri je nainštalovaná aplikácia, ktorá zabezpečuje zber, ukladanie a poskytovanie údajov o projekte. Na klientskom počítači je nainštalovaná druhá aplikácia, ktorá komunikuje s tou serverovou aplikáciou. Vymieňajú si údaje, ktorých forma a formát je vlastná aplikáciám samotným. Tieto údaje sú potom zobrazované klientskou aplikáciou. Predstaviteľom takejto kategórie je aj napríklad už spomínaný Microsoft Project.

Za veľkú nevýhodu aplikácií tejto kategórie považujem práve inštaláciu špeciálneho softvéru, ktorý jediný dokáže komunikovať so serverom. Hoci je vďaka serverovému

riešení umožnená spolupráca a prístup pre viacerých používateľov oproti jednopoužívateľským aplikáciám, používateľ je stále obmedzovaný na jeden počítač na jednom mieste, na ktorom má klientskú aplikáciu.

Ďalšiu kategóriu tvoria webové aplikácie. Webové aplikácie môžu samozrejme istou formou pripomínať klient – server aplikácie. Jednou z odlišností od nich je však to, že nepotrebujú inštaláciu špeciálneho softvéru, ktorý by komunikoval so serverom. Na komunikáciu s klientom využívajú webový prehliadač a obyčajné webové stránky. Používateľ pritom nie je obmedzovaný na používanie špeciálneho internetového prehliadača alebo nástroja. Dáva mu voľnú ruku v možnostiach pripojenia sa na webovú aplikáciu a využívania ich služieb. Predstaviteľmi takýchto aplikácií sú napríklad aplikácie dotProject, Jira, Mantis, Agilo, Jtrack.

Už len vďaka týmto vlastnostiam môžem povedať, že v rámci vývoja aplikácií a softvéru v tíme je jednoznačná výhoda webových aplikácií, ktoré sú prístupné pre viacerých členov tímu z ľubovoľného miesta. Nie sú obmedzované pôsobnosťou iba na jeden počítač a nekladú žiadne podmienky na inštaláciu špeciálneho softvéru. Toto riešenie je vhodné pre tímy, ktorých členovia pracujú v jednej firme na jednom mieste ako aj sú geograficky rozptýlení.

Triedenie z pohľadu finančného zaťaženia

Druhý spôsob rozdelenia aplikácií na sledovanie úloh je z hľadiska finančného zaťaženia či už na jednotlivé osoby alebo na firmy. Tu sa delia aplikácie do dvoch kategórií:

- komerčné
- voľne šíriteľné

Komerčné aplikácie predstavujú aplikácie, ktorých licenciu je poskytovaná používateľovi za určitý finančný obnos. Za tento finančný obnos získava samotnú aplikáciu a zákaznícku podporu v prípade problémov s aplikáciou. Do tejto kategórie spadá ako zástupca jednopoužívateľských aplikácií a aplikácií typu klient – server napríklad Microsoft Project. Zástupcom komerčnej webovej aplikácie je napríklad Jira.

Voľne šíriteľné aplikácie predstavuje kategóriu aplikácií za ktorých používanie sa neplatí, prípadne sa autorom aplikácie posielajú dobrovoľný finančný príspevok v ľubovoľnej výške. Príkladom aplikácií tejto kategórie je napríklad Mantis, dotProject alebo aj jTrack.

Definovať v tomto prípade výhody a nevýhody oboch typov kategórií aplikácií a jednoznačne určiť, ktoré riešenie je pre používateľa najlepšie, môže byť trochu problematické. Osobne by som však odporučil komerčné riešenie. Používateľ získava napríklad zákaznícku podporu alebo záplaty na prípadné chyby. Mnohé firmy, ktoré vyvíjajú tieto nástroje, kvôli čo najväčšiemu rozšíreniu svojej aplikácie majú veľmi dobré licenčné podmienky. Ceny nie sú nastavené na vysoké sumy, začínajú na malých cenách. Tu by som napríklad uviedol licenčné podmienky Jiri. Základná licencia Jiri pre desať používateľov na rok využiteľná pre komerčné účely je desať dolárov. Autori sľubujú, že týchto desať dolárov je odvedených na charitu. Pre študentov a na akademické použitie aplikácie je základná licencia poskytovaná zadarmo. Postačí vyplnenie formulára na ich

internetovej stránke. Po vyplnení a overení dát sprístupnia licenciu pre používanie nástroja.

Čo nám ponúkajú tieto nástroje?

Po získaní prehľadu, aké rôzne typy nástrojov existujú a aká je ich forma využitia, sa môžeme pýtať, čo nám všetko tieto nástroje môžu ponúknuť? Nebude predsa len lepší nástroj z kategórie jednopoužívateľskej aplikácie ako webový nástroj?

Prvým spomínaným nástrojom z kategórie jednopoužívateľských aplikácií bol **Microsoft Project**. Oproti ostatným nástrojom podľa mňa neponúka žiadne špeciálne funkcie, ktoré by jednoznačne vyjadrovali potrebu použitia daného nástroja. Čo by som však spomenul, tak zaujímavou funkciou, s ktorou som sa však nestretol pri ostatných aplikáciách, je tvorba rozpočtu podľa pridelenej práce a zdrojov. Aplikácia dokáže na základe pridelených úloh vypočítať hodnotu zadanej práce a projektu. Taktiež som si všimol, že sa vyzdvihuje výhoda používania Ganttovho diagramu, diagramu na znázornenie aktivít úloh na projekte, v tejto aplikácii. Ganttov diagram však už podporuje mnoho iných pokročilejších aplikácií a tak vyzdvihovanie Microsoft Project špeciálne kvôli tomuto diagramu, stráca postupne na význame. Príkladmi aplikácií využívajúce tento diagram môžu byť napríklad Open Workbench alebo GanttProject. Zaraďujú sa taktiež do kategórie jednopoužívateľských aplikácií. Microsoft Project však môže ťažiť z rozširiteľnosti produktov svojej materskej spoločnosti. Umožňuje napríklad integráciu úloh z produktu Microsoft Outlook. Posielanie pošty je taktiež zabezpečované pomocou Microsoft Outlook. Čo sa týka používania tohto programu, nakoľko je od Microsoftu, ktorý sa stará, aby všetky jeho aplikácie mali jednotný tvar a spôsob používania, je pomerne ľahké sa s ním oboznámiť. Avšak najväčšou nevýhodou je práve jeho zaradenie do jednopoužívateľských aplikácií. Mám síce prepojenie s e-mailovým klientom, sledovaním úloh, no tie nám stále neumožňujú zapojenie a integráciu viacerých ľudí naraz. Pre úspešnú tímovú prácu nie je vhodné, aby každý člen tímu mal nainštalovaný produkt u seba a iba do svojho si vkladal informácie o projekte, prípadne e-mailom ich preposielal ďalším členom tímu.

Ďalším spomínaným nástrojom je **jTrack**. Ide o pomerne malú webovú aplikáciu napísanú v jazyku Java. Je úplnom kontraste s aplikáciou Microsoft Project. Výhody jTrack nie sú v Microsoft Project a výhody Microsoft Project nie sú v jTrack. Aplikácia jTrack dokáže napríklad vytvárať úlohy s rozličnými doplňujúcimi dátami, ktoré môžeme počas vytvárania projektu zadefinovať. Ďalšou výhodou oproti Microsoft Project je to, že používa všetky výhody webových aplikácií – je dostupná z ktoréhokoľvek miesta a na jej používanie netreba inštalovať žiaden špeciálny softvér okrem bežného internetového prehliadača. Pre distribuovaný tím je tento nástroj oproti Microsoft Project jasnou voľbou. Samozrejme má aj svoje nevýhody. Aplikácia okrem vytvárania úloh a ich správy v podstate nedokáže nič. Nie je tu prítomné generovanie grafov, nie sú tu diskusné fóra, nie je možné definovať rozpočet projektu. Je preto vhodná na veľmi malé projekty, kde tieto pokročilé funkcionality nie sú potrebné. Vhodným prostredím na použitie sú napríklad školské projekty. Nevyžaduje sa tu pokročilé monitorovanie úloh, sledovanie financií, ale stačia jednoducho definované úlohy, ktoré treba splniť.



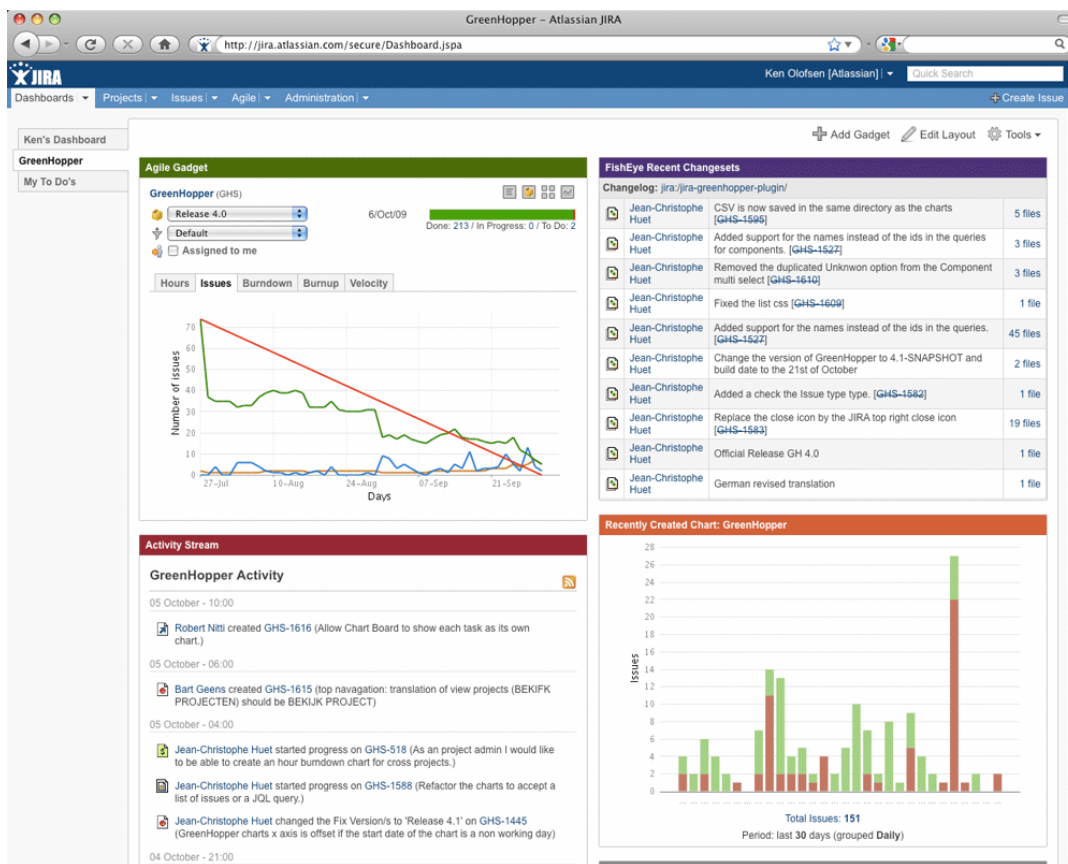
Obr. 1. Ukážka nastavení jednoduchého nástroja jTrack.

Taký prostredník medzi týmito dvoma kontrastnými aplikáciami tvorí webová aplikácia s názvom **Jira**. Ako webová aplikácia, tak ako aj jTrack, používa výhody platformy webu. Je prístupná z akéhokoľvek miesta a vhodná pre distribuované tímy. Čo sa však týka funkcionality, je na tom omnoho lepšie ako jTrack. Pomimo základných funkcií správy úloh obsahuje aj vytváranie súhrnných správ o projekte, generovanie grafov zobrazujúcich stav projektu ako napríklad Ganttov diagram alebo burndown diagram. Tento softvérový produkt je integrovateľný s ďalšími aplikáciami ako napríklad Wiki Confluence, ktorá slúži na spoločné vytváranie databázy znalostí daného tímu, komunikáciu tímu, vytváranie blogov a prezentáciu vlastných myšlienok. Ďalším nástrojom, s ktorým je možná integrácia je nástroj s názvom Greenhopper. Greenhopper pridáva možnosť spravovať projekt a jeho úlohy agilnými metódami. Je možné vďaka nemu ľahko spravovať zoznam úloh projektu, umožňuje plánovanie šprintov a sledovanie tímu počas celej doby vykonávania zadelených úloh. Okrem integrácie s existujúcimi projektmi umožňuje aplikácia vytvárať dodatočné moduly. Programátor za využitia existujúceho rozhrania môže pripraviť a upraviť funkcionality aplikácie na základe požiadaviek svojho tímu alebo firmy. Okrem vopred definovaných súhrnných správ a grafov tak má možnosť vytvárať nové kombinácie údajov a vytvárať nový náhľad na sledovanie zdrojov.

Výber vhodného softvérového prostriedku

Počas procesu výberu vhodného prostriedku je potrebné sa zamyslieť, ako si predstavujem prácu tímu, ako ho budem riadiť a ako by som asi chcel sledovať vykonané úlohy. Napríklad v prípade malého školského projektu je úplne zbytočné kúpiť komerčné riešenie, ktorého funkcionality sa naplno vôbec nevyužijú. Musíme sa zamyslieť, či je vyváženie medzi voľne dostupnou aplikáciou a námahou, ktorú vynaložíme na to, aby sme ju efektívne používali. Alebo siahneme po komerčných riešeniach, zaplatíme a budeme využívať overené a podporované softvérové produkty.

Na základe vlastností systému a vlastnej skúsenosti s ním by som ako vhodný nástroj na sledovanie úloh vybral Jiru. K jej výhodám, okrem spomínaných vlastností a funkcionality, patrí taktiež aj spomínaná licenčná politika firmy tvoriaca túto aplikáciu.



Obr. 2. Komplexnosť Jiri.

Záver

Ľudia v praveku si postupne vytvárali spoločenstvo loviace mamuty. Dokázali sa zoskupiť a spoločným úsilím uloviť potravu. Pri lovení platili isté pravidlá. Pravidlá, ktoré boli vhodným predpokladom na úspešné ukončenie akcie.

Tak ako ľudia v praveku, aj my tvoríme tímy. Tímy tvoriace softvér. Zoskupujeme sa za účelom dosiahnutia cieľa. K tomu cieľu nám tiež napomáhajú pravidlá. Pravidlá, ktorých dodržiavanie je vhodným predpokladom na úspešné ukončenie akcie.

Tak ako statočný praveký bojovník si vybral a naostril vhodnú zbraň, my sme si po dobrom zvážení vybrali vhodnú aplikáciu, ktorá nám môže pomôcť dokončiť úlohu. Úlohu predstavujúcu úspešné ukončenie projektu.

Tak ako nebojácny bojovník mal v zálohe niekoľko zbraní, aj my máme mnoho nástrojov, ktoré používajú mnoho techník riadenia ľudských zdrojov. Hoci poznáme väčšinou iba z počutia či čítania ich výhody respektíve slabiny, iba ich používaním dokážeme odhaliť ich najsilnejšie stránky a úspešne ich využívať.

Hoci mamuty už vyhynuli, tie súčasné softvérové mamuty sa postupne množia. A bude ich viac. Bude čo loviť. Je len na nás, aby sme vytiahli vhodné nástroje, nabrúsili ich a vďaka nim zdolali výzvy zaujímavých projektov.

Jedným z dôležitých prvých krokov pre úspešný boj je práve výber nástoja. V tejto eseji som predstavil tri zaujímavé nástroje určené pre rozličné skupiny bojovníkov. Ich vhodný výber môže mať dopad na úspešné ulovenie projektu.

Použitá literatúra

1. Campos A., Vale F., Alturas B.: Human resources metrics dashboard. In: *ACM Special Interest Group for Design of Communication archive*. ACM New York, NY, USA (2008) 257–262, ISBN: 978-1-60558-083-8.
2. Grinter R. E.: Using a configuration management tool to coordinate software development. In: *Proceedings of conference on Organizational computing systems*. ACM New York, NY, USA (1995) 168 – 177, ISBN: 0-89791-706-5.
3. Smart J. F.: *What issue tracking system is best for you?* (Uverejnené: 14. 03.2007) Dostupné na internete: <http://www.javaworld.com/javaworld/jw-03-2007/jw-03-bugs.html> (Citované: október 2010)
4. Benkovič P., Kuipers B. S.: *Tímová práca - mýty a realita*. (Uverejnené: 05.06.2008) Dostupné na internete: <http://podnikanie.etrend.sk/podnikanie-riadenie/timova-pracamyty-a-realita.html> (Citované: október 2010)

Annotation

Software tools as allies in process of watching performed work

In these days, development of software applications goes primarily thanks to team and team work. Times, when single person was responsible for whole application and for development of software applications are slowly disappearing. Management of single person was quite simple, there was no need to solve many different advanced solutions of work evidence or project status. One single person has summary about everything within project. But what if we have a team? How to watch their work? How to figure out with watching their work? How to find, where we stand with work? What are the opportunities about correct dividing of work between team members and not wasting the time and abilities of member of team?