



Kapitola 22 – Projektový manažment

Kapitola 23 – Plánovanie projektu

Riadenie softvérových projektov



- ✧ Zaoberá sa činnosťami súvisiacimi so zabezpečením dodania softvéru včas a podľa plánu a v súlade s požiadavkami organizácií vyvíjajúcich a obstarávajúcich softvér.
- ✧ Projektový manažment je potrebný, pretože vývoj softvéru vždy podlieha rozpočtovým a časovým obmedzeniam, ktoré stanovuje organizácia, ktorá softvér vyvíja.

Kritériá úspešnosti



- ✧ Doručte softvér zákazníkovi v dohodnutom čase.
- ✧ Udržujte celkové náklady v rámci rozpočtu.
- ✧ Dodávajte softvér, ktorý spĺňa očakávania zákazníka.
- ✧ Udržujte koherentný a dobre fungujúci vývojový tím.

Rozdiely v správe softvéru



✧ Produkt je **nehmotný**

- Softvér nie je vidieť ani sa ho nemožno dotknúť. Manažéri softvérových projektov nemôžu vidieť pokrok jednoduchým pohľadom na artefakt, ktorý sa vytvára.

✧ Mnohé softvérové projekty sú „**jednorazové**“ **projekty**

- Veľké softvérové projekty sa zvyčajne v niektorých smeroch líšia od predchádzajúcich projektov. Dokonca aj manažéri, ktorí majú veľa predchádzajúcich skúseností, môžu mať problém predvídať problémy.

✧ Softvérové procesy sú variabilné a organizačne špecifické

- Stále nemôžeme spoľahlivo predpovedať, kedy konkrétny softvérový proces pravdepodobne povedie k problémom s vývojom.

Faktory ovplyvňujúce riadenie projektu



- ✧ Veľkosť spoločnosti
- ✧ Zákazníci softvéru
- ✧ Veľkosť softvéru
- ✧ Typ softvéru
- ✧ Organizačná kultúra
- ✧ Procesy vývoja softvéru

Tieto faktory znamenajú, že projektoví manažéri v rôznych organizáciách môžu pracovať celkom odlišným spôsobom.

Univerzálne manažérske činnosti



✧ *Plánovanie projektu*

- Za plánovanie sú zodpovední projektoví manažéri. Odhadovanie a plánovanie vývoja projektu a pridelovanie ľudí k úlohám.

✧ *Riadenie rizík*

- Projektoví manažéri posudzujú riziká, ktoré môžu ovplyvniť projekt, monitorujú tieto riziká a prijímajú opatrenia, keď nastanú problémy.

✧ *Riadenie ľudí*

- Projektoví manažéri si musia vybrať ľudí do svojho tímu a zaviesť spôsoby práce, ktoré vedú k efektívnemu tímovému výkonu.

(Ďalšie) Manažérske činnosti



✧ *Nahlasovanie*

- Projektoví manažéri sú zvyčajne zodpovední za podávanie správ o priebehu projektu zákazníkom a manažérom spoločnosti vyvíjajúcej softvér.

✧ *Písanie návrhu*

- Prvá fáza softvérového projektu môže zahŕňať napísanie návrhu na získanie zákazky na vykonanie diela. Návrh popisuje ciele projektu a spôsob jeho realizácie.



Plánovanie projektu

Plánovanie projektu



- ✧ **Plánovanie projektu** zahŕňa rozdelenie práce na časti a pridelenie ich členom projektového tímu, predvídanie problémov, ktoré môžu nastať, a prípravu predbežných riešení týchto problémov.
- ✧ **Projektový plán**, ktorý sa vytvára na začiatku projektu, sa používa na informovanie projektového tímu a zákazníkov o tom, ako bude práca vykonaná, a na pomoc pri hodnotení pokroku na projekte.

Etapy plánovania



- ✧ Vo fáze návrhu , keď sa uchádzate o zákazku na vývoj alebo poskytovanie softvérového systému.
- ✧ Počas fázy spustenia projektu , keď musíte naplánovať, kto bude na projekte pracovať, ako bude projekt rozdelený na prírastky, ako sa budú prideľovať zdroje v rámci vašej spoločnosti atď.
- ✧ Pravidelne počas celého projektu , keď svoj plán upravíte na základe získaných skúseností a informácií z monitorovania postupu prác.

Plánovanie návrhu



- ✧ Plánovanie môže byť potrebné iba s rámcovými požiadavkami na softvér.
- ✧ Cieľom plánovania v tejto fáze je poskytnúť zákazníkovi informácie, ktoré sa použijú pri stanovovaní ceny za systém.
- ✧ Cena projektu zahŕňa odhad, koľko bude stáť vývoj softvéru, pričom sa zohľadnia faktory, ako sú náklady na zamestnancov, náklady na hardvér, náklady na softvér atď.

Plánovanie spustenia projektu



- ✧ V tejto fáze viete viac o systémových požiadavkách, ale nemáte informácie o návrhu alebo implementácii
- ✧ Vytvorte dostatočne podrobný plán na rozhodovanie o rozpočte projektu a personálnom obsadení.
 - Tento plán je základom pre pridelovanie zdrojov projektu
- ✧ Plán spustenia by mal definovať aj mechanizmy monitorovania projektu
- ✧ Pre agilný vývoj je stále potrebný plán spustenia, aby bolo možné prideliť zdroje na projekt

Plánovanie vývoja



- ✧ Plán projektu by sa mal pravidelne upravovať podľa toho, ako projekt postupuje a vy viete viac o softvéri a jeho vývoji
- ✧ Harmonogram projektu, odhad nákladov a riziká sa musia pravidelne revidovať

Plánom riadený vývoj



- ✧ **Plánom riadený alebo plánom založený vývoj** je prístup k softvérovému inžinierstvu, kde je proces vývoja podrobne naplánovaný.
 - Plánom riadený vývoj je založený na technikách riadenia inžinierskych projektov a je „tradičným“ spôsobom riadenia veľkých projektov vývoja softvéru.
- ✧ **plán projektu** , ktorý zaznamenáva prácu, ktorá sa má vykonať, kto ju bude vykonávať, harmonogram vývoja a pracovné produkty.
- ✧ Manažéri využívajú plán na podporu rozhodovania o projekte a ako spôsob merania pokroku.

Plánom riadený vývoj – klady a zápory



- ✧ Argumenty v prospech prístupu založeného na pláne sú, že včasné plánovanie umožňuje dôsledné zohľadnenie organizačných problémov (dostupnosť zamestnancov, iné projekty atď.) a že potenciálne problémy a závislosti sa odhalia skôr, než sa projekt začne / kým sa projekt rozbehne.
- ✧ Hlavným argumentom proti plánom riadenému vývoju je, že mnohé skoré rozhodnutia musia byť revidované kvôli zmenám v prostredí, v ktorom sa má softvér vyvíjať a používať.

Projektové plány



✧ V plánoch riadenom rozvojovom projekte **projektový plán** stanovuje **zdroje (kto)** dostupné pre projekt, **rozpis prác (čo)** a **harmonogram (kedy)** vykonávania práce.

✧ Plán úsekov

- Úvod
- Organizácia projektu
- Analýza rizík
- Požiadavky na hardvérové a softvérové zdroje
- Rozpis práce
- Harmonogram projektu
- Mechanizmy monitorovania a podávania správ

Doplňky projektového plánu



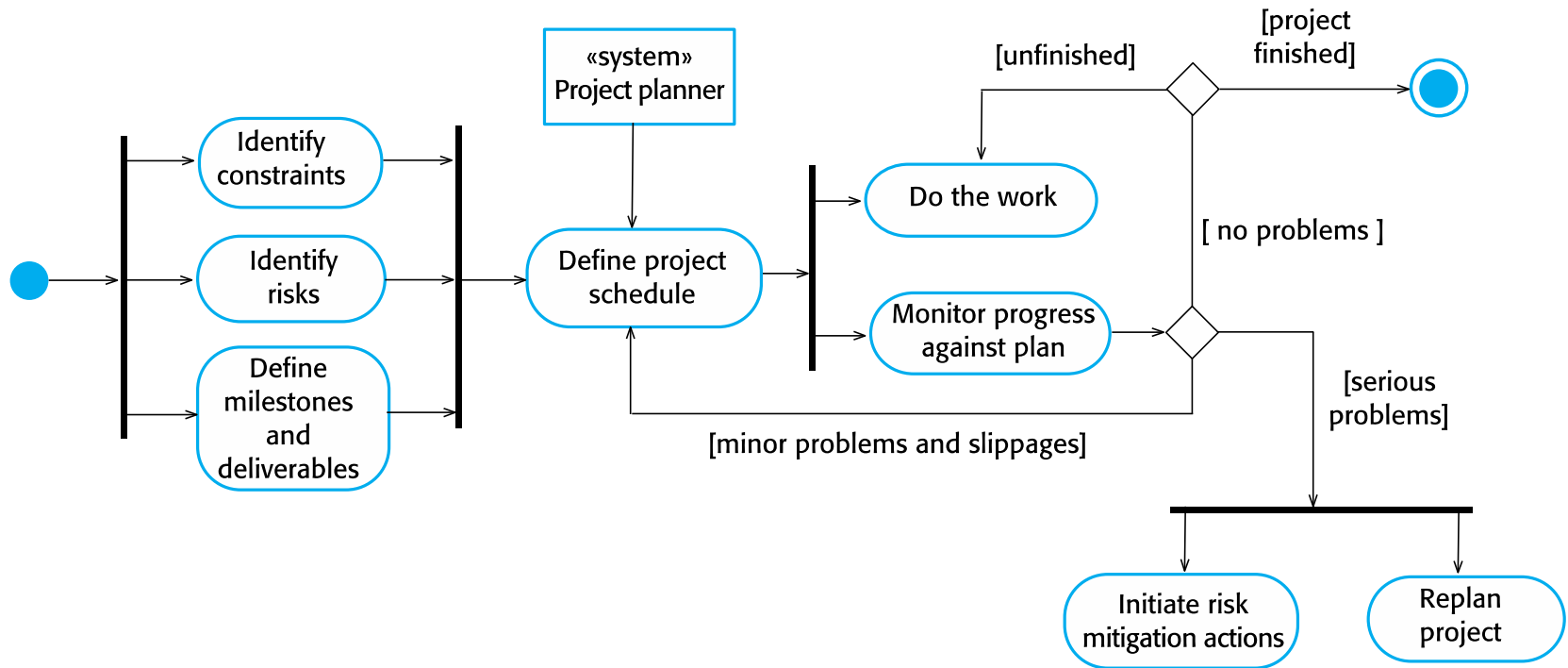
Plán	Popis
Plán riadenia konfigurácie	Opisuje postupy a štruktúry riadenia konfigurácie, ktoré sa majú použiť.
Plán nasadenia	Popisuje, ako bude softvér a súvisiaci hardvér (ak je potrebný) nasadený v prostredí zákazníka. To by malo zahŕňať plán migrácie údajov z existujúcich systémov.
Plán údržby	Predpovedá požiadavky na údržbu, náklady a úsilie.
Plán kvality	Popisuje postupy a štandardy kvality, ktoré sa použijú v projekte.
Validačný plán	Opisuje prístup, zdroje a plán používaný na overenie systému.

Proces plánovania



- ✧ Plánovanie projektu je iteratívny proces, ktorý sa začína vytvorením počiatočného plánu projektu počas fázy spustenia projektu.
- ✧ Zmeny plánu sú nevyhnutné.
 - Keď bude počas projektu k dispozícii viac informácií o systéme a projektovom tíme, mali by ste plán pravidelne revidovať, aby odrážal zmeny požiadaviek, harmonogramu a rizík.
 - Zmena obchodných cieľov vedie aj k zmenám v projektových plánoch. Keď sa obchodné ciele zmenia, môže to ovplyvniť všetky projekty, ktoré potom možno bude potrebné preplánovať.

Proces plánovania projektu



Plánovacie predpoklady



- ✧ Pri definovaní plánu projektu by ste mali robiť skôr realistické ako optimistické predpoklady.
- ✧ Počas projektu sa vždy vyskytnú nejaké problémy a vedú k omeškaniu
- ✧ Vaše počiatočné predpoklady a plánovanie by preto mali zohľadňovať neočakávané problémy.
- ✧ Mali by ste zahrnúť nepredvídané udalosti, aby ak sa niečo pokazilo, aby váš harmonogram doručenia nebol vážne narušený.

Plánovanie projektu

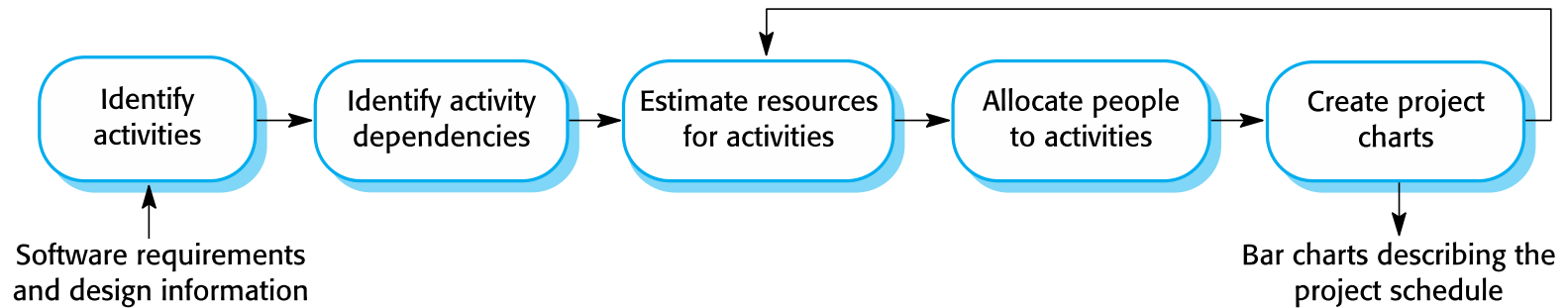


- ✧ **Plánovanie projektu** je proces rozhodovania o tom, ako bude práca v projekte organizovaná ako samostatné úlohy a kedy a ako sa tieto úlohy vykonajú.
- ✧ Odhadujete kalendárny čas potrebný na dokončenie každej úlohy, potrebné úsilie a kto bude pracovať na úlohách, ktoré boli identifikované.
- ✧ Musíte tiež odhadnúť zdroje potrebné na dokončenie každej úlohy, ako je miesto na disku požadované na serveri, čas potrebný na špecializovaný hardvér, ...



- ✧ Rozdeľte projekt na úlohy a odhadnite čas a zdroje potrebné na dokončenie každej úlohy.
- ✧ Organizujte úlohy súbežne, aby ste optimálne využili pracovnú silu.
- ✧ Minimalizujte závislosti úloh, aby ste sa vyhli oneskoreniam spôsobeným čakaním jednej úlohy na dokončenie inej.
- ✧ Plánovanie závisí od intuície a skúseností projektových manažérov.

Proces plánovania projektu



Problémy s plánovaním



- ✧ Odhadnúť náročnosť problémov a tým aj náklady na vývoj riešenia je ťažké.
- ✧ Produktivita nie je úmerná počtu ľudí pracujúcich na úlohe.
- ✧ Pridaním ľudí do neskorého projektu je to neskôr z dôvodu režijných nákladov na komunikáciu.
- ✧ Neočakávané sa vždy stane. Uvažujte aj nepredvídateľné veci pri plánovaní.

Prezentácia harmonogramu



- ✧ Grafické zobrazenie
- ✧ Zobrazuje rozdelenie projektu na úlohy. Úlohy by nemali byť príliš malé. Mali by trvať asi týždeň alebo dva.
- ✧ Na základe kalendára
 - Stĺpcové grafy sú najbežnejšie používaným znázornením plánov projektov. Zobrazujú rozvrh ako aktivity alebo zdroje v závislosti od času.
- ✧ Siete aktivít
 - Zobrazuje závislosti úloh

Projektové aktivity



- ✧ Projektové aktivity (úlohy) sú základným plánovacím prvkom. Každá aktivita má:
- trvanie v kalendárnych dňoch alebo mesiacoch,
 - úsilie , ktoré ukazuje počet osobo-dní alebo osobo-mesiacov na dokončenie práce (PD vs MD)
 - termín , do ktorého má byť činnosť ukončená,
 - definovaný koncový bod, ktorým môže byť dokument, uskutočnenie kontrolného stretnutia, úspešné vykonanie všetkých testov atď.

Míľniky a výstupy



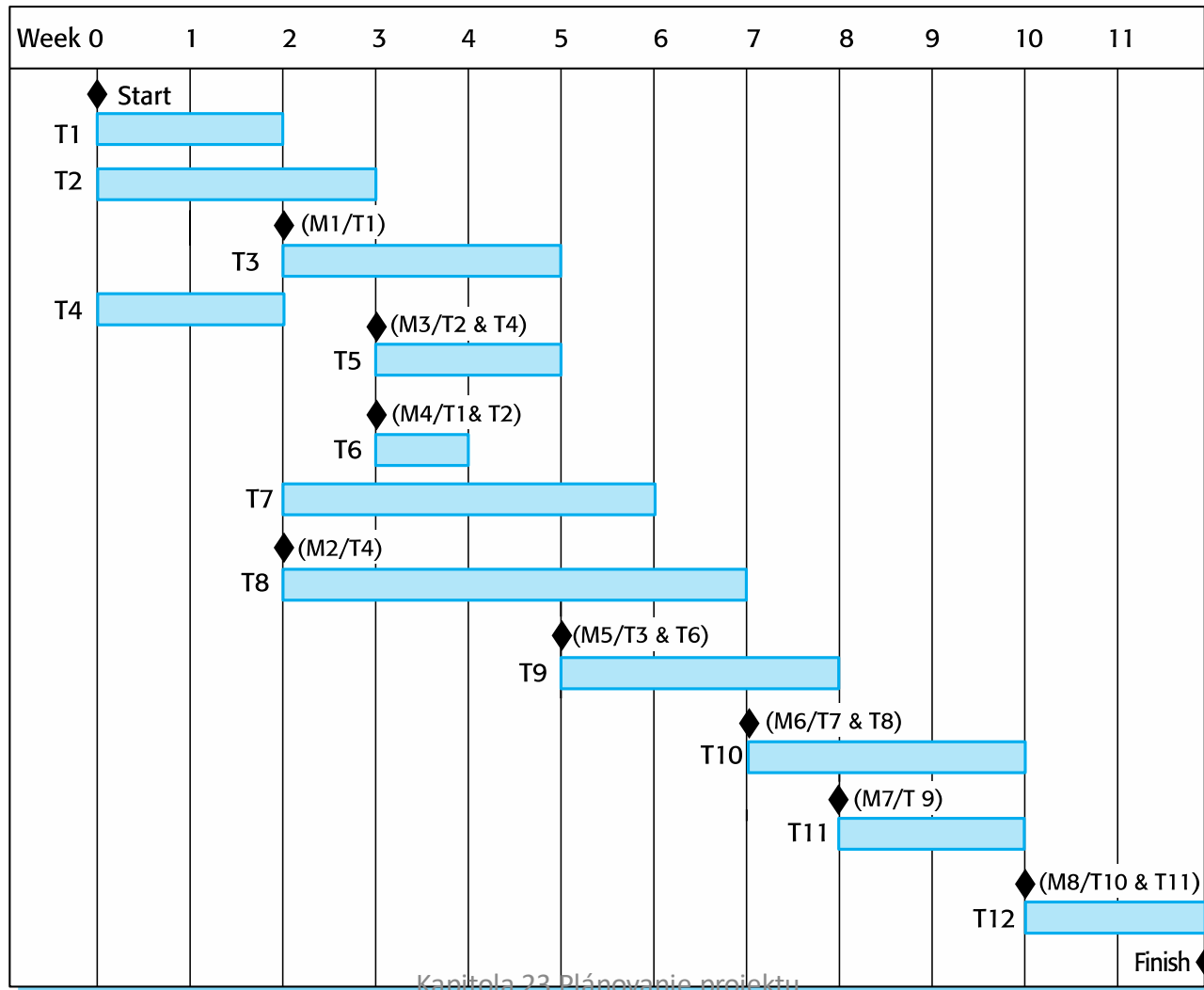
- ✧ Míľniky sú body v harmonograme, podľa ktorých môžete hodnotiť pokrok, napríklad odovzdanie systému na testovanie.
- ✧ Dodávky sú funkčné produkty, ktoré sa dodávajú zákazníkovi, napr. aj dokument požiadaviek na systém.

Úlohy , trvanie a závislosti

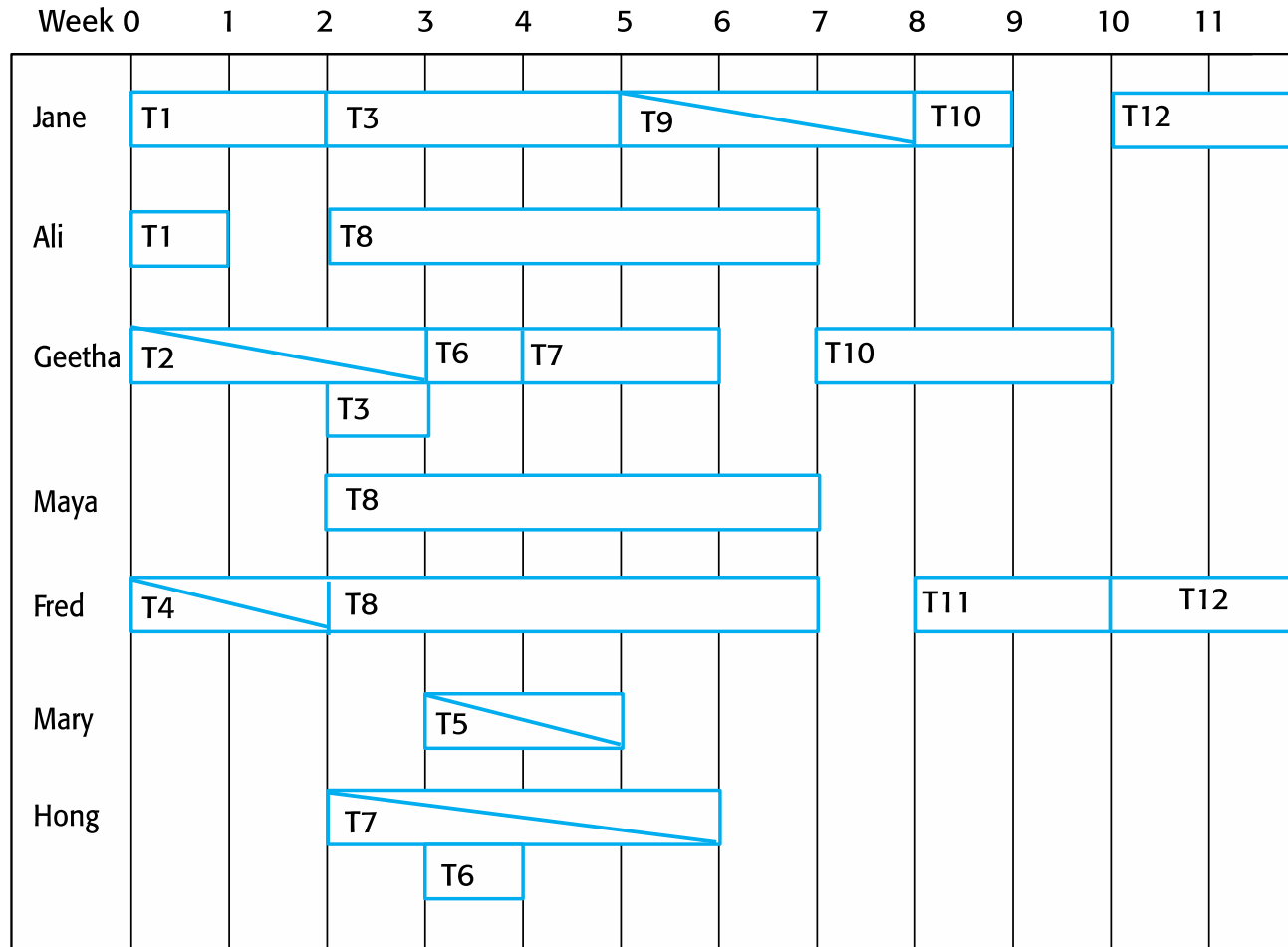


Úloha	Úsilie (osobo-dni)	Trvanie (dni)	Závislosti
T1	15	10	
T2	8	15	
T3	20	15	T1 (M1)
T4	5	10	
T5	5	10	T2, T4 (M3)
T6	10	5	T1, T2 (M4)
T7	25	20	T1 (M1)
T8	75	25	T4 (M2)
T9	10	15	T3, T6 (M5)
T10	20	15	T7, T8 (M6)
T11	10	10	T9 (M7)
T12	20	10	T10, T11 (M8)

Stĺpcový graf aktivity (Gantt)



Tabuľka pridelovania zamestnancov





- ✧ Agilné metódy vývoja softvéru sú iteratívne prístupy, pri ktorých sa softvér vyvíja a dodáva zákazníkovi postupne.
- ✧ Na rozdiel od plánom riadených prístupov sa funkčnosť týchto prírastkov neplánuje vopred, ale rozhoduje sa o nich počas vývoja.
 - Rozhodnutie o tom, čo zahrnúť do prírastku, závisí od pokroku a od priorít zákazníka.
- ✧ Priority a požiadavky zákazníka sa menia, preto má zmysel mať flexibilný plán, ktorý sa týmto zmenám dokáže prispôbiť.

Etapy agilného plánovania



- ✧ **Plánovanie vydania**, ktoré sa pozerá dopredu na niekoľko mesiacov a rozhoduje o funkciách, ktoré by mali byť súčasťou vydania systému.
- ✧ **Iteračné plánovanie**, ktoré má krátkodobý výhľad a zameriava sa na plánovanie ďalšieho prírastku systému. Zvyčajne ide o 2-4 týždne práce pre tím.

Prístupy k agilnému plánovaniu



✧ Plánovanie v Scrume

✧ Založené na riadení nevybavených úloh

- Veci, ktoré treba robiť s dennými kontrolami pokroku a problémov

✧ Plánovacia hra

- Pôvodne vyvinutý ako súčasť Extreme Programming (XP)
- Závisí od príbehov používateľov ako miery pokroku v projekte

Plánovanie založené na príbehoch



- ✧ Plánovacia hra je založená na používateľských príbehoch, ktoré odrážajú funkcie, ktoré by mal systém obsahovať.
- ✧ Projektový tím si príbehy prečíta a prediskutuje a zoradí ich podľa času, ktorý si myslia, že implementácia príbehu bude trvať.
- ✧ Príbehom sú priradené „body úsilia“ odrážajúce ich veľkosť a náročnosť implementácie
- ✧ Meria sa počet bodov úsilia implementovaných za deň, čím sa získa odhad „rýchlosti“ tímu
- ✧ To umožňuje odhadnúť celkové úsilie potrebné na implementáciu systému

Plánovacia hra





Riadenie rizík



- ✧ Riadenie rizík sa zaoberá identifikáciou rizík a zostavovaním plánov na minimalizáciu ich vplyvu na projekt .
- ✧ Manažment softvérových rizík je dôležitý kvôli neistotám spojeným s vývojom softvéru.
 - Tieto neistoty pramenia z voľne definovaných požiadaviek, zmien požiadaviek v dôsledku zmien potrieb zákazníkov, ťažkostí pri odhadovaní času a zdrojov potrebných na vývoj softvéru a rozdielov v individuálnych zručnostiach.
- ✧ Musíte predvídať riziká, chápať vplyv týchto rizík na projekt, produkt a podnikanie a podniknúť kroky, aby ste sa týmto rizikám vyhli.

Klasifikácia rizika



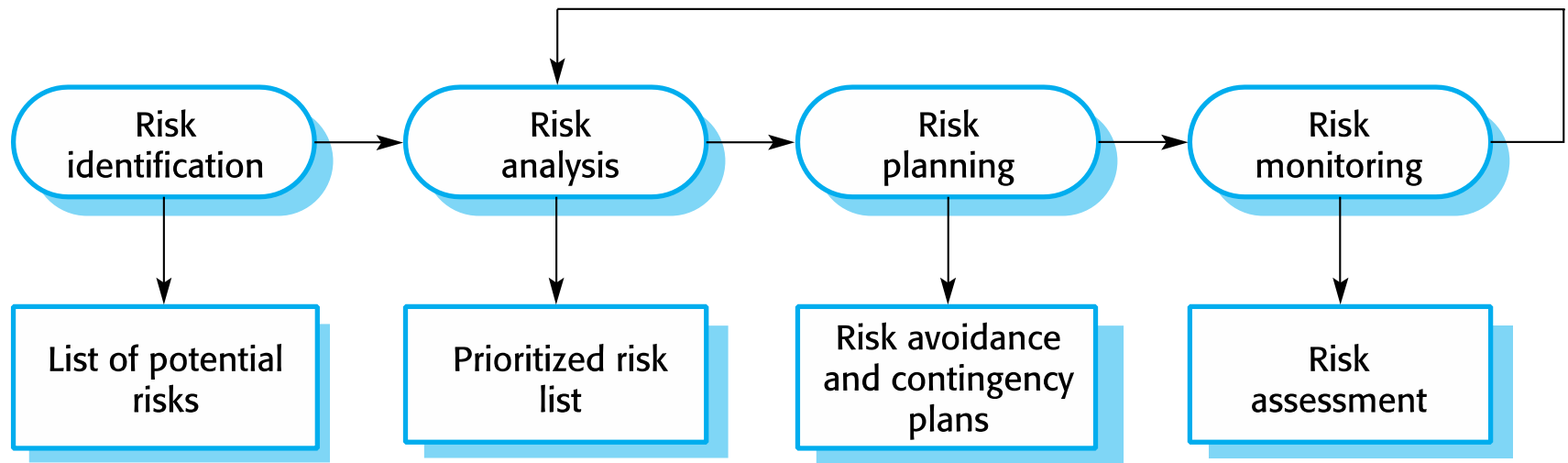
- ✧ Existujú dva rozmery **klasifikácie rizika**
 - Druh rizika (technické, organizačné, ..)
 - čo je ovplyvnené rizikom:
- ✧ *Riziká projektu* ovplyvniť plán alebo zdroje ;
- ✧ *Riziká produktu* ovplyvniť kvalitu alebo výkon vyvíjaného softvéru ;
- ✧ *Podnikateľské riziká* ovplyvňujú organizáciu, ktorá vyvíja alebo obstaráva softvér.

Príklady rizík projektu , produktov a podnikania



Riziko	Ovplyvňuje	Popis
Fluktuácia zamestnancov	Projekt	Skúsení pracovníci opustia projekt ešte pred jeho dokončením.
Zmena manažmentu	Projekt	Dôjde k zmene organizačného manažmentu s odlišnými prioritami.
Nedostupnosť hardvéru	Projekt	Hardvér, ktorý je pre projekt nevyhnutný, nebude dodaný podľa plánu.
Požiadavky sa menia	Projekt a produkt	V požiadavkách dôjde k väčšiemu počtu zmien, než sa predpokladalo.
Oneskorenia špecifikácií	Projekt a produkt	Špecifikácie základných rozhraní nie sú k dispozícii podľa plánu.
Veľkosť podhodnotená	Projekt a produkt	Veľkosť systému bola podcenená.
Nedostatočný výkon nástroja CASE	Produkt	CASE nástroje, ktoré podporujú projekt, nefungujú tak, ako sa očakávalo.
Zmena technológie	Podnikanie	Základná technológia, na ktorej je systém postavený, je nahradená novou technológiou.
Konkurencia produktov	Podnikanie	Konkurenčný produkt je uvedený na trh pred dokončením systému.

Proces riadenia rizík



Proces riadenia rizík



✧ Identifikácia rizika

- Identifikujte projektové, produktové a obchodné riziká;

✧ Analýza rizík

- Posúdiť pravdepodobnosť a dôsledky týchto rizík;

✧ Plánovanie rizika

- Vypracujte plány na zabránenie alebo minimalizovanie účinkov rizika;

✧ Monitorovanie rizík

- Monitorujte riziká počas celého projektu;

Identifikácia rizika



- ✧ Môže ísť o tímové aktivity alebo na základe skúseností jednotlivých projektových manažérov.
- ✧ Na identifikáciu rizík v projekte možno použiť kontrolný zoznam spoločných rizík
 - Technologické riziká .
 - Organizačné riziká .
 - Ľudské riziká.
 - Riziká z požiadaviek .
 - Riziká z odhadov .

Príklady rôznych typov rizík



Typ rizika	Možné riziká
Odhad	Čas potrebný na vývoj softvéru je podhodnotený. (12) Miera opravy defektu je podhodnotená. (13) Veľkosť softvéru je podhodnotená. (14)
Organizačné	Organizácia je reštrukturalizovaná tak, že za projekt je zodpovedný iný manažment. (6) Finančné problémy organizácie si vynucujú zníženie rozpočtu projektu. (7)
Ľudia	Nie je možné získať zamestnancov s požadovanými zručnosťami. (3) Kľúčoví zamestnanci sú v kritických časoch chorí a nedostupní. (4) Požadované školenie pre zamestnancov nie je k dispozícii. (5)
Požiadavky	Navrhujú sa zmeny požiadaviek, ktoré si vyžadujú veľké prepracovanie dizajnu. (10) Zákazníci nedokážu pochopiť vplyv zmien požiadaviek. (11)
Technológia	Databáza používaná v systéme nedokáže spracovať toľko transakcií za sekundu, ako sa očakáva. (1) Opätovne použiteľné softvérové komponenty obsahujú chyby, ktoré znamenajú, že ich nemožno opätovne použiť podľa plánu. (2)
Nástroje	Kód generovaný nástrojmi na generovanie softvérového kódu je neefektívny. (8) Softvérové nástroje nemôžu spolupracovať integrovaným spôsobom. (9)

Analýza rizík



- ✧ Posúďte pravdepodobnosť a závažnosť každého rizika.
- ✧ Pravdepodobnosť môže byť veľmi nízka, nízka, stredná, vysoká alebo veľmi vysoká.
- ✧ Rizikové dôsledky môžu byť katastrofálne, vážne, tolerovateľné alebo bezvýznamné.

rizík a príklady



Riziko	Pravdepodobnosť	Účinky
organizácie si vynucujú zníženie rozpočtu projektu (7).	Nízka	Katastrofálne
Nie je možné získať zamestnancov so zručnosťami požadovanými pre projekt (3).	Vysoká	Katastrofálne
Kľúčoví zamestnanci sú chorí v kritických časoch projektu (4).	Mierne	Vážne
Chyby v opakovane použiteľných softvérových komponentoch sa musia opraviť skôr, ako sa tieto komponenty opätovne použijú. (2).	Mierne	Vážne
Navrhujú sa zmeny požiadaviek, ktoré si vyžadujú veľké prepracovanie dizajnu (10).	Mierne	Vážne
Organizácia je reštrukturalizovaná tak, že za projekt je zodpovedný iný manažment (6).	Vysoká	Vážne
Databáza používaná v systéme nedokáže spracovať toľko transakcií za sekundu, ako sa očakáva (1).	Mierne	Vážne

Typy rizík a príklady



Riziko	Pravdepodobnosť	Účinky
Čas potrebný na vývoj softvéru sa podceňuje (12).	Vysoká	Vážne
Softvérové nástroje nemožno integrovať (9).	Vysoká	Znesiteľné
Zákazníci nedokážu pochopiť vplyv zmien požiadaviek (11).	Mierne	Znesiteľné
Požadované školenie pre zamestnancov nie je k dispozícii (5).	Mierne	Znesiteľné
Miera opravy defektov je podhodnotená (13).	Mierne	Znesiteľné
Veľkosť softvéru je podhodnotená (14).	Vysoká	Znesiteľné
Kód generovaný nástrojmi na generovanie kódu je neefektívny (8).	Mierne	Bezvýznamný

Plánovanie rizika



- ✧ Zvážte každé riziko a vytvorte stratégiu na riadenie tohto rizika.
- ✧ **Stratégie vyhýbania sa**
 - Pravdepodobnosť vzniku rizika je znížená;
- ✧ **Stratégie minimalizácie**
 - Zníži sa vplyv rizika na projekt alebo produkt;
- ✧ **Pohotovostné plány**
 - Ak riziko vznikne, pohotovostné plány sú plány na riešenie tohto rizika;

Čo keby otázky



- ✧ Čo ak je chorých viacero inžinierov súčasne?
- ✧ Čo ak ekonomický pokles povedie k zníženiu rozpočtu projektu o 20 %?
- ✧ Čo ak je výkon softvéru s otvoreným zdrojovým kódom nedostatočný a jediný odborník na tento softvér s otvoreným zdrojovým kódom odíde?
- ✧ Čo ak spoločnosť, ktorá dodáva a udržiava softvérové komponenty, zanikne?
- ✧ Čo ak zákazník nedoručí upravené požiadavky tak, ako sa predpokladalo?

Stratégie , ktoré pomáhajú riadiť riziko



Riziko	Stratégia
Organizačné finančné problémy	Pripravte inštruktážny dokument pre vrcholový manažment, ktorý ukáže, ako projekt veľmi dôležitým spôsobom prispieva k cieľom podnikania, a uvedie dôvody, prečo by škrtý v rozpočte projektu neboli nákladovo efektívne.
Problémy s náborom	Upozorniť zákazníka na možné ťažkosti a možnosť omeškania; skúmať nákupné komponenty.
Ochorenie personálu	Preorganizujte tím tak, aby sa práce viac prekrývali a ľudia si preto navzájom rozumeli.
Chybné komponenty	Vymeňte potenciálne chybné komponenty za zakúpené komponenty so známou spoľahlivosťou.
Zmeny požiadaviek	Odvodit' informácie o sledovateľnosti na posúdenie vplyvu zmeny požiadaviek; maximalizovať informácie ukryté v dizajne.

Stratégie , ktoré pomáhajú riadiť riziko



Riziko	Stratégia
Organizačná reštrukturalizácia	Pripravte informačný dokument pre vrcholový manažment, ktorý ukáže, ako projekt veľmi dôležitým spôsobom prispieva k cieľom podnikania.
Výkon databázy	Preskúmajte možnosť nákupu výkonnejšej databázy.
Podcenený čas vývoja	Preskúmajte nákupné komponenty; preskúmať použitie generátora programov .

Monitorovanie rizík



- ✧ Pravidelne vyhodnocujte každé identifikované riziko, aby ste sa rozhodli, či sa stáva menej alebo viac pravdepodobným.
- ✧ Posúďte tiež, či sa účinky rizika zmenili.
- ✧ Každé kľúčové riziko by sa malo prediskutovať na poradách vedenia.

Ukazovatele rizika



Typ rizika	Potenciálne ukazovatele
Odhad	Nedodržanie dohodnutého harmonogramu; neodstránenie nahlásených závad .
Organizačné	Organizačné klebety; nedostatok činnosti zo strany vrcholového manažmentu.
Ľudia	Zlá morálka zamestnancov; zlé vzťahy medzi členmi tímu; vysoká fluktuácia zamestnancov.
Požiadavky	Mnoho požiadaviek mení požiadavky; sťažnosti zákazníkov.
Technológia	Oneskorené dodanie hardvéru alebo podporného softvéru; veľa hlásených technologických problémov.
Nástroje	Neochota členov tímu používať nástroje; sťažnosti týkajúce sa nástrojov CASE; požiadavky na výkonnejšie pracovné stanice.



Riadenie ľudí

Riadenie ľudí



- ✧ Ľudia sú najdôležitejším aktívom organizácie.
- ✧ Úlohy manažéra sú v podstate zamerané na ľudí. Bez pochopenia ľudí bude manažment neúspešný.
- ✧ Zlé riadenie ľudí je dôležitým prispievateľom k zlyhaniu projektu.

Faktory riadenia ľudí



✧ Dôslednosť

- So všetkými členmi tímu by sa malo zaobchádzať porovnateľným spôsobom bez obľúbencov alebo diskriminácie.

✧ Rešpekt

- Rôzni členovia tímu majú rôzne zručnosti a tieto rozdiely by sa mali rešpektovať.

✧ Inklúzia

- Zapojte všetkých členov tímu a uistite sa, že sa berú do úvahy názory ľudí.

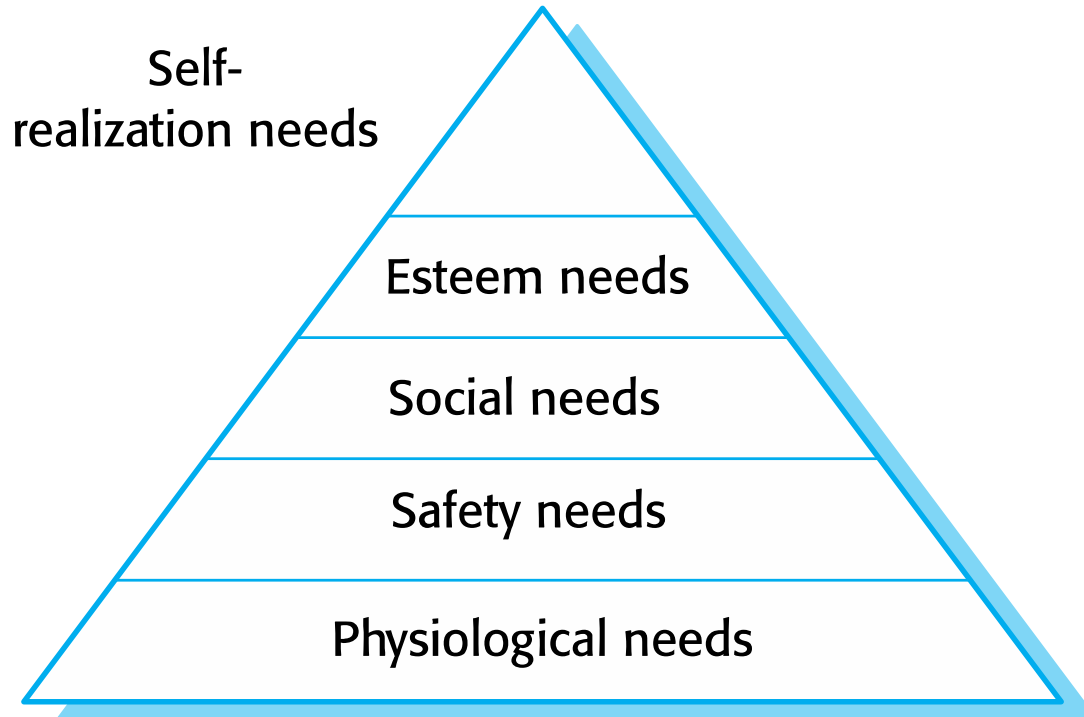
✧ Čestnosť

- Vždy by ste mali byť úprimní v tom, čo v projekte ide dobre a čo zle.



- ✧ Dôležitou úlohou manažéra je motivovať ľudí pracujúcich na projekte .
- ✧ Motivácia znamená organizáciu práce a pracovného prostredia tak, aby povzbudzovala ľudí k efektívnej práci.
 - Ak ľudia nebudú motivovaní, práca, ktorú robia, ich nebude zaujímať. Budú pracovať pomaly, budú častejšie robiť chyby a nebudú prispievať k širším cieľom tímu alebo organizácie.
- ✧ Motivácia je zložitý problém, ale zdá sa, že existujú rôzne typy motivácie založené na:
 - Základné potreby (napr. jedlo, spánok atď.);
 - Osobné potreby (napr. rešpekt, sebaúcta);
 - Sociálne potreby (napr. byť akceptovaný ako súčasť skupiny).

Hierarchia ľudských potrieb



Potreba uspokojenia



- ✧ V skupinách vývoja softvéru nie sú základné fyziologické a bezpečnostné potreby problémom.
- ✧ Sociálna
 - Poskytovať komunálne zariadenia;
 - Umožnite neformálnu komunikáciu napr. prostredníctvom sociálnych sietí
- ✧ Úcta
 - Uznanie úspechov;
 - Primerané odmeny.
- ✧ Sebarealizácia
 - Školenie – ľudia sa chcú dozvedieť viac;
 - Zodpovednosť.

Typy osobnosti



- ✧ Hierarchia potrieb je takmer určite prílišným zjednodušením motivácie v praxi.
- ✧ Motivácia by mala brať do úvahy aj rôzne typy osobnosti:
 - Úlohovo orientovaní ľudia, ktorí sú motivovaní prácou, ktorú robia. V softvérovom inžinierstve.
 - Orientovaný na interakciu ľudí, ktorí sú motivovaní prítomnosťou a konaním spolupracovníkov.
 - Sebaorientovaný ľudí, ktorých motivuje predovšetkým osobný úspech a uznanie.

Typy osobnosti



✧ Orientovaný na úlohy.

- Motiváciou pre vykonanie práce je samotná práca;

✧ Sebaorientovaný.

- Práca je prostriedkom na dosiahnutie cieľa, ktorým je dosiahnutie individuálnych cieľov – napr. zbohatnúť, hrať tenis, cestovať atď.;

✧ Orientovaný na interakciu

- Hlavnou motiváciou je prítomnosť a činy spolupracovníkov. Ľudia chodia do práce, pretože do práce chodia radi
-

Rovnováha motivácie



- ✧ Jednotlivé motivácie sa skladajú z prvkov každej triedy.
- ✧ Rovnováha sa môže meniť v závislosti od osobných okolností a vonkajších udalostí.
- ✧ Ľudia však nie sú motivovaní len osobnými faktormi, ale aj tým, že sú súčasťou skupiny a kultúry.
- ✧ Ľudia chodia do práce, pretože ich motivujú ľudia, s ktorými pracujú.



Tímová práca

Tímová práca



- ✧ Väčšina softvérového inžinierstva je skupinová činnosť
 - Plán vývoja väčšiny netriviálnych softvérových projektov je taký, že ich nemôže dokončiť jedna osoba pracujúca sama .
- ✧ Dobrá partia je súdržná a má tímového ducha. Zainteresovaní ľudia sú motivovaní úspechom skupiny, ako aj vlastnými osobnými cieľmi.
- ✧ Skupinová interakcia je kľúčovým determinantom skupinového výkonu.
- ✧ Flexibilita v zložení skupiny je obmedzená
 - Manažéri musia robiť to najlepšie, čo môžu s dostupnými ľuďmi.

Skupinová súdržnosť



- ✧ V súdržnej skupine členovia považujú skupinu za dôležitejšiu ako ktorýkoľvek jednotlivec v nej.
- ✧ Výhody **súdržnej skupiny** sú:
 - Normy kvality skupiny môžu vytvoriť členovia skupiny.
 - tímu sa od seba navzájom učia a spoznávajú prácu toho druhého ; Znižujú sa zábrany spôsobené nevedomosťou.
 - Vedomosti sa delia. Kontinuita môže byť zachovaná, ak člen skupiny odíde.
 - Podporuje sa refaktoring a neustále zlepšovanie. Členovia skupiny spoločne pracujú na poskytovaní vysokokvalitných výsledkov a riešení problémov, bez ohľadu na jednotlivcov, ktorí pôvodne vytvorili dizajn alebo program.

Efektívnosť tímu



✧ Ľudia v skupine

- V projektovej skupine potrebujete zmes ľudí, pretože vývoj softvéru zahŕňa rôzne činnosti, ako je vyjednávanie s klientmi, programovanie, testovanie a dokumentácia.

✧ Organizácia skupiny

- Skupina by mala byť organizovaná tak, aby jednotlivci mohli prispievať podľa svojich najlepších schopností a aby úlohy mohli byť splnené podľa očakávania.

✧ Technická a manažérska komunikácia

- Dobrá komunikácia medzi členmi skupiny a medzi tímom softvérového inžinierstva a ostatnými zainteresovanými stranami projektu je nevyhnutná.

Výber členov skupiny



- ✧ Úlohou manažéra alebo vedúceho tímu je vytvoriť súdržnú skupinu a organizovať svoju skupinu tak, aby mohli efektívne spolupracovať.
- ✧ To zahŕňa vytvorenie skupiny so správnou rovnováhou technických zručností a osobností a organizáciu tejto skupiny tak, aby jej členovia efektívne spolupracovali.

Zostavenie tímu



- ✧ Možno nebude možné vymenovať ideálnych ľudí na prácu na projekte
 - Rozpočet projektu nemusí umožňovať využitie vysoko platených zamestnancov;
 - Zamestnanci s príslušnými skúsenosťami nemusia byť k dispozícii;
 - Organizácia môže chcieť rozvíjať zručnosti zamestnancov na softvérovom projekte.
- ✧ Manažéri musia pracovať v rámci týchto obmedzení, najmä ak je nedostatok vyškoleného personálu.

Zloženie skupiny



- ✧ Problematická môže byť skupina zložená z členov, ktorí zdieľajú rovnakú motiváciu
 - Orientácia na úlohy – každý chce robiť to svoje;
 - Orientácia na seba – každý chce byť šéf;
 - Orientované na interakciu – príliš veľa chatovania, málo práce.
- ✧ Efektívna skupina má rovnováhu všetkých typov.
- ✧ To môže byť ťažké dosiahnuť, softvéroví inžinieri sú často orientovaní na úlohy.
- ✧ Ľudia orientovaní na interakciu sú veľmi dôležití, pretože dokážu odhaliť a zmierniť vzniknuté napätie.

Skupinová organizácia



- ✧ Spôsob, akým je skupina organizovaná, ovplyvňuje rozhodnutia, ktoré táto skupina robí, spôsoby výmeny informácií a interakcie medzi vývojovou skupinou a externými účastníkmi projektu.
 - Medzi kľúčové otázky patria:
 - Mal by byť projektový manažér technickým vedúcim skupiny?
 - Kto sa bude podieľať na prijímaní dôležitých technických rozhodnutí a ako sa budú robiť?
 - Ako sa budú riešiť interakcie s externými zainteresovanými stranami a vrcholovým manažmentom spoločnosti?
 - Ako môžu skupiny integrovať ľudí, ktorí nie sú spolu?
 - Ako možno zdieľať vedomosti v rámci skupiny?

Skupinová organizácia



- ✧ **Malé skupiny** softvérového inžinierstva sú zvyčajne organizované neformálne bez pevnej štruktúry.
- ✧ Pri **veľkých projektoch** môže existovať hierarchická štruktúra, v ktorej sú rôzne skupiny zodpovedné za rôzne podprojekty .
- ✧ **Agilný vývoj** je vždy založený na neformálnej skupine na princípe, že formálna štruktúra bráni výmene informácií