

Managementul proiectului

Obiective

Obiectivul de baza este discuta despre managementul proiectelor software și despre două activități de management importante, și anume

- gestionarea riscurilor și
- managementul persoanelor.

Vom învăța următoarele:

- a cunoaște principalele sarcini ale managerilor de proiect software;
- de a înțelege noțiunea de gestionare a riscurilor și unele dintre riscurile care pot apărea în proiectele software;
- de eviedența factorii care influențează motivația personală și ce anume ar putea însemna aceștia pentru managerii de proiect software;
- de a înțelege problemele cheie care influențează munca în echipă, cum ar fi echipa compoziție, organizare și comunicare.

Avem trei compartimente:

1. Managementul riscurilor
2. Gestionarea oamenilor
3. Munca în echipă

Managementul proiectelor software este o parte esențială a ingineriei software trebuie gestionate deoarece ingineria software profesională este întotdeauna **supusă bugetului organizațional și constrângerile de programare**. Sarcina managerului de proiect este de a asigura că proiectul software îndeplinește și depășește aceste constrângeri, precum și livrarea software de înaltă calitate.

O bună gestionare nu poate garanta succesul proiectului. Cu toate acestea, o gestionare defectuoasă duce de obicei la eșecul proiectului: software-ul poate fi livrat târziu, costă mai mult decât se estimează inițial sau nu îndeplinesc așteptările clienților.

Criteriile de succes pentru managementul proiectelor variază în mod evident de la proiect la proiectdar, pentru majoritatea proiectelor, obiectivele importante sunt:

1. Livrați software-ul către client la ora stabilită.
2. Mențineți costurile globale în limita bugetului.
3. Furnizați software care satisface așteptările clientului.
4. Mențineți o echipă de dezvoltare fericită și care funcționează bine.

Aceste obiective nu sunt unice pentru ingineria software, ci sunt obiectivele tuturor ingineriei. Cu toate acestea, ingineria software este diferită de alte tipuri de ingineria într-**o serie de moduri care fac managementul software-ului deosebit de dificil.** Unele dintre aceste diferențe sunt:

1. **Produsul este intangibil. Un manager** al unei construcții navale sau al unei construcții civile poate vedea produsul în curs de dezvoltare. Dacă un program alunecă, efectul asupra produsului este vizibil - părți ale structurii sunt evident neterminate. Software-ul este intangibil, nu poate fi văzut sau atins. Managerii de proiect software nu pot vedea progresul uitându-se pur și simplu la artefactul care este construit. Mai degrabă, se bazează pe alții pentru a produce dovezi pe care le pot folosi pentru a revizui progresul muncii.

2. **Proiectele software mari sunt deseori proiecte „unice”** Proiectele software mari sunt de obicei diferite în anumite privințe de proiectele anterioare, prin urmare, chiar și managerii celor care au un corp mare de experiență anterioară le poate fi greu de anticipat. În plus, schimbări tehnologice rapide în calculatoare și comunicările pot face experiența unui manager învechită. Lecții învățate din proiecte anterioare s-ar putea să nu fie transferabile către proiecte noi.

3. **Procesele software sunt variabile și specifice organizației** Ingeria proceselor de dezvoltare pentru unele tipuri de sistem, cum ar fi poduri și clădiri, este bine înțeleasă. Cu toate acestea, procesele software variază destul de semnificativ de la o organizație la alta la altul. Deși s-au înregistrat progrese semnificative în standardizarea proceselor și îmbunătățire, încă nu putem prezice în mod fiabil când un anumit software proces poate duce la probleme de dezvoltare. Acest lucru este valabil mai ales atunci când proiectul software face parte dintr-un proiect mai larg de inginerie de sisteme.

Din cauza unor probleme, **nu este surprinzător faptul că unele proiecte software întârzie,** depășește bugetul și întârzie programele. Sistemele software sunt adesea noi și tehnice inovatoare. Proiecte de inginerie (cum ar fi sisteme noi de transport) care sunt inovatoare.

De multe ori au și probleme de programare. Având în vedere dificultățile implicate, este posibil remarcabil faptul că atât de multe proiecte software sunt livrate la timp și la buget! Este imposibil să scrieți o descriere standard a postului pentru un manager de proiect software. Locul de muncă variază enorm în funcție de organizație și de produsul software

Cu toate acestea, majoritatea managerilor își asumă responsabilitatea la un moment dat pentru unele sau toate următoarele activități:

1. **Planificarea proiectului** Managerii de proiect sunt responsabili de planificare, estimare și planificarea dezvoltării proiectelor și atribuirea oamenilor la sarcini. Ei supraveghează munca pentru a se asigura că este efectuată la standardele și monitorizarea cerute progrese pentru a verifica dacă dezvoltarea este la timp și în limita bugetului.

2. **Raportarea** Managerii de proiect sunt de obicei responsabili pentru raportarea asupra progresului unui proiect către clienți și către managerii companiei în curs de dezvoltare

Trebuie să poată comunica la o serie de niveluri, de la informații tehnice detaliate la rezumatele conducerii. Trebuie să scrie documente concise, coerente, care abstractizează informațiile critice din detalii rapoartele proiectului. Trebuie să fie capabili să prezinte aceste informații în timpul dezvoltării.

3. **Managementul riscurilor** Managerii de proiect trebuie să evalueze riscurile care pot afecta proiectul, monitorizați aceste riscuri și luați măsuri atunci când apar probleme.

4. **Managementul persoanelor** Managerii de proiect sunt responsabili pentru gestionarea unei echipe detrebuie să aleagă oameni pentru echipa lor și să stabilească modalități de lucrare duc la performanța eficientă a echipei.

5. **Documentarea propunerii** Prima etapă a unui proiect software poate presupune scrierea unei propuneri pentru a câștiga un contract pentru a realiza o lucrare. Propunerea descrie obiectivele proiectului și modul în care acesta va fi realizat. Acesta include de obicei costurile și programează estimări și justifică de ce ar trebui atribuit contractul de proiect pentru o anumită organizație sau echipă. Scrierea de propuneri este o sarcină critică ca supraviețuire din multe companii de software depinde de a avea suficiente propunerii acceptate și contractele atribuite. Nu pot exista linii directoare stabilite pentru această sarcină; **documentarea de propuneri** este o abilitate pe care o dobândești prin practică și experiență.

1. Managementul riscurilor

Managementul riscurilor este unul dintre cele mai importante sarcină de muncă pentru un manager de proiect managementul implică anticiparea riscurilor care ar putea afecta programul proiectului sau calitatea software-ului dezvoltat și apoi luarea de măsuri pentru a evita aceste riscuri

Riscurile pot amenința proiectul, software-ul dezvoltat, sau organizația. Prin urmare, există trei categorii de riscuri conexe:

1. **Riscurile proiectului** Riscurile care afectează programul sau resursele proiectului. Un exemplu de un risc de proiect este pierderea unui proiectant experimentat. Găsirea unui înlocuitor proiectantului cu abilități și experiență adecvate poate dura mult timp și, de preferință, proiectarea software-ului va dura mai mult timp.

2. **Riscuri de produs** Riscuri care afectează calitatea sau performanța software-ului Un exemplu de risc al produsului este **eșecul unei componente achiziționate să funcționeze conform așteptărilor.** Acest lucru poate afecta performanța generală a sistemului astfel încât să fie mai lent decât se aștepta.

3. **Riscuri comerciale** Riscuri care afectează organizația care dezvoltă sau achiziționează software. De exemplu, un concurent care introduce un produs nou este o afacere introducerea

unui produs competitiv poate însemna că ipotezele despre vânzările de produse software existente poate fi extrem de optimist.

Desigur, aceste tipuri de risc se suprapun. Dacă un programator cu experiență părăsește un proiect acesta poate fi un risc al proiectului deoarece, chiar dacă sunt înlocuite imediat, programul va fi afectat. Este nevoie inevitabil de timp pentru un nou membru al proiectului să înțeleagă munca care a fost făcută, astfel încât acestea să nu poată fi imediat productive.

Prin urmare, livrarea sistemului poate fi întârziată. Pierderea unui membru al echipei poate fi, de asemenea, un risc de produs, deoarece un înlocuitor poate să nu fie la fel de cu experiență și așa ar putea face erori de programare. În cele din urmă, poate fi o afacere risc, deoarece experiența programatorului poate fi crucială pentru a câștiga noi contracte.

Ar trebui să înregistrați rezultatele analizei de risc în planul de proiect împreună cu un analiza consecințelor, care stabilește consecințele riscului pentru proiect, produs și afaceri. Gestionarea eficientă a riscurilor facilitează gestionarea probleme și să se asigure că acestea nu duc la buget sau program inacceptabil alunecare.

Riscurile specifice care pot afecta un proiect depind de proiect și organizational mediu în care se dezvoltă software-ul. Totuși, acolo sunt, de asemenea, riscuri frecvente care nu sunt legate de tipul de software dezvoltat iar acestea pot apărea în orice proiect. Unele dintre aceste riscuri comune sunt prezentate în Figura de mai jos

Risk	Affects	Description
Staff turnover	Project	Experienced staff will leave the project before it is finished.
Management change	Project	There will be a change of organizational management with different priorities.
Hardware unavailability	Project	Hardware that is essential for the project will not be delivered on schedule.
Requirements change	Project and product	There will be a larger number of changes to the requirements than anticipated.
Specification delays	Project and product	Specifications of essential interfaces are not available on schedule.
Size underestimate	Project and product	The size of the system has been underestimated.
CASE tool underperformance	Product	CASE tools, which support the project, do not perform as anticipated.
Technology change	Business	The underlying technology on which the system is built is superseded by new technology.
Product competition	Business	A competitive product is marketed before the system is completed.

Figure 22.1
Examples of common project, product, and business risks

An outline of the process of risk management is illustrated in Figure 22.2. It involves several stages:

1. *Risk identification* You should identify possible project, product, and business

Gestionarea riscurilor este deosebit de importantă pentru proiectele software din cauza incertitudinii inerente cu care se confruntă majoritatea proiectelor. Acestea provin din definiții slabe cerințe, modificări ale cerințelor datorate schimbărilor în nevoile clienților, dificultăți în estimarea timpului și a resurselor necesare dezvoltării software-ului și a diferențelor în abilitățile individuale. Trebuie să anticipați riscurile; să înțelegeți impactul acestor riscuri pentru proiect, produs și afacere și luați măsuri pentru a le evita

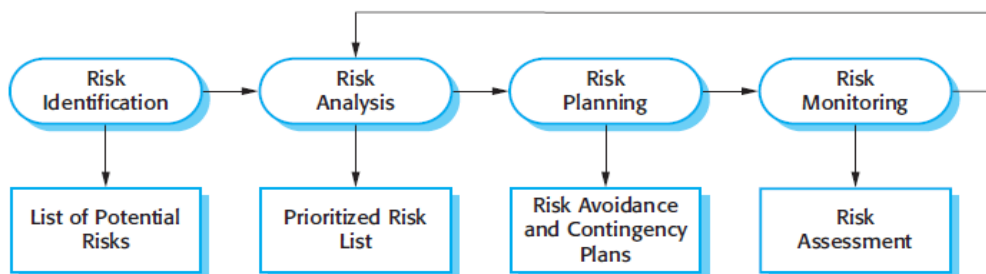
Este posibil să trebuiască să elaborați planuri de urgență, astfel încât, dacă există riscuri, să vă puteți lua măsuri de recuperare imediată.

O schemă a procesului de gestionare a riscurilor este ilustrată în Figura 22.2 presupune mai multe etape:

1. **Identificarea riscului** Ar trebui să identificați posibilul proiect, produs și afacere riscuri.
2. **Analiza riscurilor** Ar trebui să evaluați probabilitatea și consecințele acestor riscuri.
3. **Planificarea riscului** Ar trebui să faceți planuri pentru a aborda riscul, fie evitându-l, fie minimizarea efectelor sale asupra proiectului.
4. **Monitorizarea riscului** Ar trebui să evaluați în mod regulat riscul și planurile dvs. de risc atenuați și revizuiți-le atunci când aflați mai multe despre risc.

Ar trebui să documentați rezultatele procesului de gestionare a riscurilor într-un plan de management al riscurilor. Acesta ar trebui să includă o discuție asupra riscurilor cu care se confruntă proiectul, o analiză a acestor riscuri și informații despre cum propuneți să gestionați riscul dacă pare să fie o problemă.

Procesul de gestionare a riscurilor este un proces iterativ care continuă pe tot parcursul. Odată ce ați întocmit un plan inițial de gestionare a riscurilor, monitorizați pentru a detecta riscurile emergente. Pe măsură ce sunt disponibile mai multe informații despre riscuri, trebuie să reanalizați riscurile și să decideți dacă prioritatea riscului s-a schimbat poate fi necesar să vă schimbați planurile de evitare a riscurilor și de gestionare a contingențelor.



a) Identificarea riscului

Identificarea riscurilor este prima etapă a procesului de gestionare a riscurilor cu identificarea riscurilor care ar putea reprezenta o amenințare majoră pentru ingineria software proces, software-ul dezvoltat sau organizația de dezvoltare. Identificarea riscului poate fi un proces de echipă în care o echipă se reunește pentru a face brainstorming posibil riscuri.,

managerul de proiect poate pur și simplu să-și folosească experiența identifică riscurile cele mai probabile sau critice.

Ca punct de plecare pentru identificarea riscurilor, se poate face o listă de verificare a diferitelor tipuri de risc

Există cel puțin șase tipuri de risc care pot fi incluse într-o listă de verificare a riscurilor:

1. **Riscuri tehnologice** Riscuri care decurg din tehnologiile software sau hardware care sunt utilizate pentru dezvoltarea sistemului.
2. **Riscuri pentru oameni** Riscuri asociate cu persoanele din echipa de dezvoltare.
3. **Riscuri organizaționale** Riscuri care decurg din mediul organizational unde se dezvoltă software-ul.
4. **Riscurile instrumentelor** Riscurile care decurg din instrumentele software și alte programe de asistență folosit pentru dezvoltarea sistemului.
5. **Riscuri de cerințe** Riscuri care decurg din modificări ale cerințelor clienților iar procesul de gestionare a cerințelor se schimbă.
6. **Riscuri estimate** Riscuri care decurg din estimările de management ale resursele necesare construirii sistemului.

Câteva exemple de riscuri posibile în fiecare dintre aceste categorii

Risk type	Possible risks
Technology	The database used in the system cannot process as many transactions per second as expected. (1) Reusable software components contain defects that mean they cannot be reused as planned. (2)
People	It is impossible to recruit staff with the skills required. (3) Key staff are ill and unavailable at critical times. (4) Required training for staff is not available. (5)
Organizational	The organization is restructured so that different management are responsible for the project. (6) Organizational financial problems force reductions in the project budget. (7)
Tools	The code generated by software code generation tools is inefficient. (8) Software tools cannot work together in an integrated way. (9)
Requirements	Changes to requirements that require major design rework are proposed. (10) Customers fail to understand the impact of requirements changes. (11)
Estimation	The time required to develop the software is underestimated. (12) The rate of defect repair is underestimated. (13) The size of the software is underestimated. (14)

Când ați terminat procesul de identificare a riscurilor, ar trebui să aveți o listă lungă de riscurile care ar putea apărea și care ar putea afecta produsul, procesul și afacerea.

Apoi, trebuie să tăiați această listă la o dimensiune gestionabilă. **Dacă aveți prea multe riscuri, este practic imposibil să le urmăriți pe toate.**

b) Analiza riscurilor

În timpul procesului de analiză a riscului, trebuie să luați în considerare fiecare risc identificat și faceți.

Nu există o cale ușoară de a judeca cu privire la probabilitatea și gravitatea acestui risc. faceți acest lucru. Trebuie să vă bazați pe propria judecată și experiența proiectelor anterioare și problemele care au apărut în ele. **Nu este posibil să se facă precise, numerice evaluarea probabilității și gravității fiecărui risc.** Mai degrabă **ar trebui să atribuiți riscul pentru una dintre numeroasele situații:**

1. Probabilitatea riscului poate fi evaluată ca fiind **foarte scăzută (10%)**, scăzută (10-25%), moderat (25-50%), ridicat (50-75%) sau foarte mare (75%).

2. Efectele riscului pot fi evaluate ca fiind **a) catastrofale** (amenință supraviețuirea proiectul), **b) grav** (ar provoca întârzieri majore), **c) tolerabil** (întârzierile sunt în interior contingență permisă) sau d) **ne semnificativă**.

Apoi ar trebui să intabulați rezultatele acestui proces de analiză folosind un tabel ordonat în funcție de gravitatea riscului. Figura 22.4 ilustrează acest lucru pentru riscurile pe care I au identificat în Figura 22.3 Evident, evaluarea probabilității și a gravității este arbitrar aici. Pentru a face această evaluare, aveți nevoie de informații detaliate despre proiectul, procesul, echipa de dezvoltare și organizația.

Desigur, atât probabilitatea, cât și evaluarea efectelor unui risc pot schimbați pe măsură ce mai multe informații despre risc devin disponibile și ca gestionare a riscurilor planurile sunt implementate. Prin urmare, ar trebui să actualizați acest tabel în timpul fiecăruia iterația procesului de risc.

Odată ce riscurile au fost analizate și clasificate, ar trebui să evaluați care dintre acestea riscurile sunt cele mai semnificative. Judecata dvs. trebuie să depindă de o combinație a probabilitatea apariției riscului și efectele acestui risc. În general, catastrofale riscurile trebuie întotdeauna luate în considerare, la fel ca toate riscurile grave care au mai mult de un probabilitate moderată de apariție.

Risk	Probability	Effects
Organizational financial problems force reductions in the project budget (7).	Low	Catastrophic
It is impossible to recruit staff with the skills required for the project (3).	High	Catastrophic
Key staff are ill at critical times in the project (4).	Moderate	Serious
Faults in reusable software components have to be repaired before these components are reused. (2).	Moderate	Serious
Changes to requirements that require major design rework are proposed (10).	Moderate	Serious
The organization is restructured so that different management are responsible for the project (6).	High	Serious
The database used in the system cannot process as many transactions per second as expected (1).	Moderate	Serious
The time required to develop the software is underestimated (12).	High	Serious
Software tools cannot be integrated (9).	High	Tolerable
Customers fail to understand the impact of requirements changes (11).	Moderate	Tolerable
Required training for staff is not available (5).	Moderate	Tolerable
The rate of defect repair is underestimated (13).	Moderate	Tolerable
The size of the software is underestimated (14).	High	Tolerable
Code generated by code generation tools is inefficient (8).	Moderate	Insignificant

Boehm (1988) recomandă identificarea și monitorizarea primelor 10 riscuri, dar eu credeți că această cifră este destul de arbitrară. Numărul adecvat de riscuri de monitorizat trebuie depinde de proiect. Ar putea fi 5 sau ar putea fi 15. Totuși, numărul de riscuri alese pentru monitorizare ar trebui să fie gestionabile. Un număr foarte mare de riscuri ar Pur și simplu cereți prea multe informații pentru a fi colectate. Din riscurile identificate în În figura 22.4, este adecvat să se ia în considerare cele 8 riscuri care prezintă catastrofe sau grave consecințe (Figura 22.5).

c) Planificarea riscurilor

Procesul de planificare a riscurilor ia în considerare fiecare dintre riscurile cheie care au fost identificate, și dezvoltă strategii pentru gestionarea acestor riscuri. Pentru fiecare dintre riscuri, trebuie să vă gândiți de acțiuni pe care le-ați putea întreprinde pentru a minimiza întreruperea proiectului dacă problema identificat în acest risc apare. De asemenea, ar trebui să vă gândiți la informații pe care le-ați putea trebuie să colecteze în timp ce monitorizați proiectul, astfel încât problemele să poată fi anticipate.

Din nou, nu există un proces simplu care să poată fi urmărit pentru planificarea de urgență se bazează pe judecata și experiența managerului de proiect. Figura de mai jos prezintă posibile strategii de gestionare a riscurilor care au fost identificate pentru riscurile cheie (de exemplu, cele grave sau intolerabile) prezentate în Figura de mai sus .

Risk	Strategy
Organizational financial problems	Prepare a briefing document for senior management showing how the project is making a very important contribution to the goals of the business and presenting reasons why cuts to the project budget would not be cost-effective.
Recruitment problems	Alert customer to potential difficulties and the possibility of delays; investigate buying-in components.
Staff illness	Reorganize team so that there is more overlap of work and people therefore understand each other's jobs.
Defective components	Replace potentially defective components with bought-in components of known reliability.
Requirements changes	Derive traceability information to assess requirements change impact; maximize information hiding in the design.
Organizational restructuring	Prepare a briefing document for senior management showing how the project is making a very important contribution to the goals of the business.
Database performance	Investigate the possibility of buying a higher-performance database.
Underestimated development time	Investigate buying-in components; investigate use of a program generator.

Figure 22.5 Strategies to help manage risk

Again, there is no simple process that can be followed for contingency planning. It relies on the judgment and experience of the project manager.

Figure 22.5 shows possible risk management strategies that have been identified

Aceste strategii se împart în trei categorii:

1. **Strategii de evitare** Urmarea acestor strategii înseamnă că probabilitatea ca riscul va apărea va fi redus. Un exemplu de strategie de evitare a riscului este strategie pentru tratarea componentelor defecte prezentate în Figura 22.5.
2. **Strategii de minimizare** Urmărirea acestor strategii înseamnă că impactul riscului va fi redus. Un exemplu de strategie de minimizare a riscului este strategia pentru boala personalului prezentată în Figura 22.5.
3. **Planuri de urgență** Urmarea acestor strategii înseamnă că sunteți pregătit pentru cel mai rău și au o strategie pusă la punct pentru a face față acestuia. Un exemplu de contingență strategia este strategia pentru problemele financiare organizaționale pe care le am prezentat în Figura de mai **sus**

Aici puteți vedea o analogie clară cu strategiile utilizate în sistemele critice pentru asigurați fiabilitatea, securitatea și siguranța, acolo unde trebuie să evitați, să tolerați sau să vă recuperați Evident, cel mai bine este să folosiți o strategie care să evite riscul. În caz contrar posibil, ar trebui să utilizați o strategie care reduce șansele pe care le va avea riscul efecte grave. În cele din urmă, ar trebui să aveți strategii în vigoare pentru a face față riscului dacă acesta

Acestea **ar trebui să reducă impactul general al unui risc asupra proiectului sau produsului.**

d) Monitorizarea riscurilor

Monitorizarea riscurilor este procesul de verificare a ipotezelor dumneavoastră despre produs, procesului, iar riscurile de afaceri nu s-au schimbat. Ar trebui să evaluați în mod regulat fiecare dintre acestea riscurile identificate pentru a decide dacă acest risc devine sau nu mai probabil.

De asemenea, ar trebui să vă gândiți dacă s-au schimbat sau nu efectele riscului. Pentru a face acest lucru, trebuie să vă uitați la alți factori, cum ar fi numărul de cerințe solicitări de modificare, care vă oferă indicii despre probabilitatea riscului și efectele acestuia. Acești factori sunt în mod evident dependenți de tipurile de risc. Figura 22.6 prezintă unele exemple de factori care pot fi de ajutor în evaluarea acestor tipuri de risc.

Risk type	Potential indicators
Technology	Late delivery of hardware or support software; many reported technology problems.
People	Poor staff morale; poor relationships amongst team members; high staff turnover.
Organizational	Organizational gossip; lack of action by senior management.
Tools	Reluctance by team members to use tools; complaints about CASE tools; demands for higher-powered workstations.
Requirements	Many requirements change requests; customer complaints.
Estimation	Failure to meet agreed schedule; failure to deal reported defects.

Figure 22.6 Risk indicators

Ar trebui să monitorizați periodic riscurile în toate etapele unui proiect. La fiecare management revizuiți, ar trebui să luați în considerare și să discutați fiecare dintre riscurile cheie separat. Ar trebui să luați în considerare și să discutați fiecare dintre riscurile cheie separat. decideți dacă riscul este mai mult sau mai puțin probabil să apară și dacă gravitatea și consecințele a riscului s-au schimbat.

22.2 Gestionarea oamenilor

Persoanele care lucrează într-o organizație software sunt cele mai importante aturi ale sale recrutează și reține oameni buni și este responsabilitatea managerilor de software să se asigure că organizația obține cel mai bun randament posibil al investiției sale. În companii de succes și economii, acest lucru se realizează atunci când oamenii sunt respectați de organizație și li se atribuie responsabilități care reflectă abilitățile și experiența lor.

Este important ca managerii de proiect software să înțeleagă problemele tehnice care influențează munca de dezvoltare software. Din păcate, totuși, bine inginerii software nu sunt neapărat buni manageri de oameni au deseori abilități tehnice puternice, dar pot lipsi abilitățile mai ușoare care le permit motivează și conduce o echipă de dezvoltare a proiectului.

Ca manager de proiect, ar trebui să fii conștient de problemele potențiale ale managementului oamenilor și ar trebui să încerce să se dezvolte abilități de gestionare a oamenilor.

De regulă, există patru factori critici în managementul persoanelor:

1. **Coerență** (согласованность) Oamenii dintr-o echipă de proiect ar trebui tratați într-un mod comparabil.
Nimeni nu se așteaptă ca toate recompensele să fie identice, dar oamenii nu ar trebui să simtă că au contribuția la organizație este subevaluată.
2. **Respect** Diferite persoane au abilități diferite, iar managerii ar trebui să le respecte diferențelor. Toți membrii echipei ar trebui să aibă posibilitatea de a face un
În unele cazuri, desigur, veți constata că oamenii pur și simplu nu se potrivesc într-o echipă și nu pot continua, dar este important să nu treci la concluzii despre acest lucru într-un stadiu incipient al proiectului.
3. **Incluziune** Oamenii contribuie eficient atunci când simt că ceilalți îi ascultă și să țină cont de propunerile lor. Este important să se dezvolte un mediu de lucru unde sunt luate în considerare toate punctele de vedere, chiar și cele ale celor mai juniori personal.
4. **Onestitate** Ca manager, ar trebui să fii întotdeauna sincer cu privire la ceea ce merge bine și ce se întâmplă prost în echipă. De asemenea, ar trebui să fii sincer cu privire la nivelul tău de cunoștințe tehnice și dispus să amâne personalul cu mai multe cunoștințe atunci când

Dacă încerci să ascunzi ignoranța sau problemele, vei fi în cele din urmă aflat și va pierde respectul grupului. Managementul oamenilor, în opinia mea, este ceva care trebuie să se bazeze pe experiență, mai degrabă decât învățat dintr-o carte. Scopul meu în această secțiune și în următoarea secțiune despre munca în echipă este pur și simplu pentru a introduce unii dintre cei mai importanți oameni și managementul echipei probleme care afectează managementul proiectelor software. Sper că materialul de aici va fi vă sensibilizează la unele dintre problemele cu care managerii se pot confrunta atunci când se ocupă echipe de indivizi talentați tehnic.

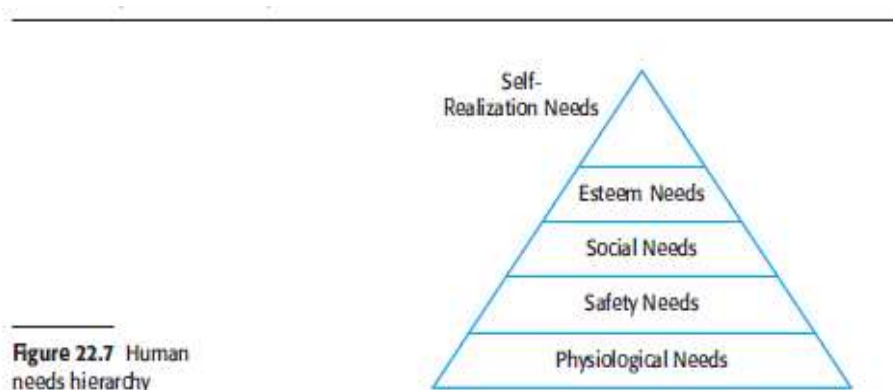
22.2.1 Motivarea oamenilor

Ca manager de proiect, trebuie să motivați oamenii care lucrează cu dvs., astfel încât Acestea contribuie la cele mai bune abilități. Motivarea înseamnă organizarea muncii și mediul de lucru pentru a încuraja oamenii să lucreze cât mai eficient posibil.

Dacă oamenii nu sunt motivați, nu vor fi interesați de munca pe care o fac. Vor lucra încet, vor fi mai predispuși să greșească și nu vor contribui la asta obiectivele mai largi ale echipei sau ale organizației.

Pentru a oferi această încurajare, ar trebui să înțelegeți puțin despre ceea ce motivează sugerează că oamenii sunt motivați prin satisfacerea lor

Aceste nevoi sunt aranjate într-o serie de niveluri, după cum se arată în Figura 22.7 nivelurile inferioare ale acestei ierarhii reprezintă nevoi fundamentale de hrană, somn și așa mai departe, și nevoia de a se simți în siguranță într-un mediu. Nevoile sociale sunt preocupate de nevoia de a se simți parte a unui grup social. Nevoile de stimă reprezintă nevoia de a simți



respectate de ceilalți, iar nevoile de auto-realizare se referă la dezvoltarea personală.

Oamenii trebuie să satisfacă nevoile de nivel inferior, cum ar fi foamea, înaintea celor mai abstracte, nevoi de nivel superior.

Oamenii care lucrează în organizații de dezvoltare software nu sunt de obicei flămânzi sau

sete sau amenințate fizic de mediul lor. Prin urmare, asigurându-vă că nevoile sociale, de stimă și de realizare de sine ale oamenilor sunt satisfăcute este cel mai important din punct de vedere al managementului.

1. Pentru a satisface nevoile sociale, trebuie să le acordați oamenilor timp pentru a-și întâlni colegii de muncă și le oferiți locuri de întâlnire. Acest lucru este relativ ușor atunci când toate membrii unei echipe de dezvoltare lucrează în același loc, dar, treptat, echipă membrii nu se află în aceeași clădire sau chiar în același oraș sau stat.

Pot lucra pentru diferite organizații sau de acasă, de cele mai multe ori.

Sistemele de rețele sociale și teleconferința pot fi utilizate pentru a facilita comunicațiile dar experiența mea cu sistemele electronice este că acestea sunt cele mai multe

eficientă odată ce oamenii se cunosc. Prin urmare, trebuie să le aranjați întâlniri față în față la începutul proiectului, astfel încât oamenii să poată interacționa direct cu alți membri ai echipei. Prin această interacțiune directă, oamenii devin parte dintr-un grup social și acceptă obiectivele și prioritățile acelui grup.

2. Pentru a satisface nevoile de stimă, trebuie să le arătați oamenilor că sunt apreciați de Recunoașterea publică a realizărilor este un mod simplu, dar eficient

Evident, oamenii trebuie să simtă, de asemenea, că sunt plătiți la un nivel care reflectă abilitățile și experiența lor.

3. În cele din urmă, pentru a satisface nevoile de auto-realizare, trebuie să le oferi oamenilor responsabilitate pentru munca lor, atribuiți-le sarcini solicitante (dar nu imposibile) și furnizați o program de formare în care oamenii își pot dezvolta abilitățile. Formarea este un lucru important influență motivantă, deoarece oamenii le place să dobândească noi cunoștințe și să învețe noi

aptitudini.

A fi membru al unui grup coeziv este extrem de motivant pentru majoritatea oamenilor. Cu locuri de muncă îndeplinite, de multe ori le place să meargă la serviciu, deoarece sunt motivați de oameni lucrează și munca pe care o fac. Acolo, precum și gândirea la individ motivație trebuie să te gândești și la modul în care un grup în ansamblu poate fi motivat pentru a atinge obiectivele organizației. Discut probleme de management al grupului în următoarea secțiune.

Tipul de personalitate influențează și motivația în trei tipuri:

1. Oameni orientați spre sarcini, **care sunt motivați de munca pe care o fac. În ingineria software**, aceștia sunt oameni care sunt motivați de provocarea intelectuală a dezvoltare de software.

2. **Oameni orientați spre sine, care sunt motivați în principal de succesul personal și** Sunt interesați de dezvoltarea de software ca mijloc de realizare. Acest lucru nu înseamnă că acești oameni sunt egoiști și gândesc mai degrabă la obiective pe termen lung, cum ar fi progresul în carieră, care îi motivează și doresc să aibă succes în lucrări pentru a vă ajuta să realizați aceste obiective.

3. **Oameni orientați spre interacțiune, care sunt motivați de prezența și acțiuni**

Pe măsură ce dezvoltarea software devine mai centrată pe utilizator, orientată spre interacțiune indivizii sunt din ce în ce mai implicați în ingineria software.

Personalităților orientate spre interacțiune le place, de obicei, să lucreze ca parte a unui grup, oamenii orientați spre sarcini și auto-orientați preferă de obicei să acționeze ca indivizi. Femeile sunt mai susceptibile de a fi orientate spre interacțiune decât bărbații. De multe ori sunt mai eficiente ca comunicatori.

Motivația fiecărui individ este alcătuită din elemente ale fiecărei clase, dar un singur tip motivația este de obicei dominantă în orice moment. Cu toate acestea, indivizii pot

.De exemplu, oamenii tehnici care **simt că nu sunt corect recompensat** poate deveni auto-orientat și pune interesele personale înaintea preocupărilor tehnice.

Dacă un grup funcționează deosebit de bine, oamenii auto-orientați pot deveni mai mulți orientată spre interacțiune.

22.3 Munca în echipă

Majoritatea software-urilor profesionale sunt dezvoltate de echipe de proiect care variază în mărime de la două pentru câteva sute de oameni. Cu toate acestea, deoarece este în mod clar imposibil pentru toată lumea dintr-un grup mare pentru a lucra împreună la o singură problemă, echipele mari sunt de obicei împărțite în un număr de grupuri. Fiecare grup este responsabil pentru dezvoltarea unei părți din ansamblu

Ca regulă generală, grupurile de proiecte de inginerie software nu ar trebui să aibă mai mult decât 10 membri. Când se utilizează grupuri mici, problemele de comunicare sunt reduse. Toată lumea îi cunoaște pe toți ceilalți și întregul grup se poate înconjura de o masă pentru o întâlnire pentru a discuta proiectul și software-ul pe care îl dezvoltă.

Alcătuirea unui grup care are echilibrul adecvat de abilități tehnice, experiență, și personalitățile este o sarcină critică de gestionare. Cu toate acestea, grupurile de succes sunt mai mult decât o simplă colecție de indivizi cu echilibrul adecvat al abilităților grupul este coeziv și are un spirit de echipă. Oamenii implicați sunt motivați de succesul grupului precum și prin propriile obiective personale.

Într-un grup coeziv, membrii cred că grupul este mai important decât indivizii care sunt membri ai grupului. Membrii unui grup bine condus și coeziv sunt loiali grupului. Se identifică cu obiectivele grupului și cu alți membri ai grupului. Încearcă să protejezi grupul, ca entitate, de interferențe din exterior. Acest lucru face ca grupul să fie robust și capabil să facă față problemelor și situațiilor neașteptate.

Avantajele creării unui grup coeziv sunt:

1. Grupul își poate stabili propriile standarde de calitate, deoarece aceste standarde sunt stabilite prin consens, acestea sunt mai susceptibile de a fi respectate decât standardele externe impuse grupului.

2. Persoanele învață și se sprijină reciproc. Oamenii din grup învață Inhibițiile cauzate de ignoranță sunt reduse la minimum, așa cum este învățarea reciprocă încurajată.

3. Cunoașterea este partajată Continuitatea poate fi menținută dacă un membru al grupului pleacă.

Alții din grup pot prelua sarcini critice și se pot asigura că proiectul nu este tulburat nejustificat.

4. Refactorizarea și îmbunătățirea continuă sunt încurajate să lucreze membrii grupului Colectiv pentru a oferi rezultate de înaltă calitate și a rezolva problemele, indiferent de persoane care au creat inițial designul sau programul.

Managerii de proiect buni ar trebui să încerce întotdeauna să încurajeze coeziunea grupului poate organiza evenimente sociale pentru membrii grupului și familiile acestora, încercați să stabiliți un

simț de identitate de grup prin denumirea grupului și stabilirea unei identități de grup și sau se pot implica în activități explicite de construire a grupurilor, cum ar fi sportul și jocuri.

Una dintre cele mai eficiente modalități de promovare a coeziunii este de a fi incluziv înseamnă că ar trebui să tratați membrii grupului ca fiind responsabili și de încredere și Pune la dispoziție informațiile în mod liber. Uneori, managerii simt că nu pot dezvălui anumite informații pentru toată lumea din grup. Acest lucru creează invariabil un climat de neîncredere. Schimbul simplu de informații este un mod eficient de a face oamenii să se simtă apreciați

și că fac parte dintr-un grup.

Puteți vedea un exemplu în acest caz în studiul de caz din Figura 22.9

întâlniri informale periodice în care le spune celorlalți membri ai grupului ce se întâmplă.

Ea își propune să implice oamenii în dezvoltarea produsului, cerându-le

să vină cu idei noi derivate din propriile experiențe de familie. „Zilele departe” sunt, de asemenea, modalități bune de a promova coeziunea - oamenii se relaxează împreună în timp ce ajută

reciproc învață despre noile tehnologii.

Dacă un grup este sau nu eficient depinde, într-o oarecare măsură, de natura proiectul și organizația care lucrează. Dacă o organizație se află într-o stare de frământare cu reorganizări constante și nesiguranța locului de muncă, este foarte dificil pentru membrii echipei

să se concentreze pe dezvoltarea de software. Totuși, în afară de proiect și organizațional probleme, există trei factori generici care afectează munca în echipă:

1. Persoanele din grup Aveți nevoie de un amestec de oameni dintr-un grup de proiecte ca software

dezvoltarea implică diverse activități, cum ar fi negocierea cu clienții, programarea, testare și documentare.

2. Organizarea grupului Un grup ar trebui să fie organizat astfel încât indivizii să poată contribuie la cele mai bune abilități și sarcinile lor pot fi îndeplinite conform așteptărilor.

3. Comunicări tehnice și manageriale Comunicări bune între membrii grupului și între echipa de inginerie software și alte proiecte vehicule, este esențială.

La fel ca în toate problemele de management, obținerea echipei potrivite nu poate garanta proiectul

succes. Prea multe alte lucruri pot merge prost, inclusiv modificări ale afacerii și mediul de afaceri. Cu toate acestea, dacă nu acordați atenție compoziției grupului, organizare și comunicații, creșteți probabilitatea ca proiectul dvs. să crească întâmpina dificultăți.

22.3.1 Selectarea membrilor grupului

Sarcina unui manager sau a unui lider de echipă este de a crea un grup coeziv și de a le organiza grup, astfel încât să poată lucra împreună eficient. Aceasta implică crearea unui grup cu echilibrul adecvat dintre abilitățile și personalitățile tehnice și organizarea grupului respectiv că membrii lucrează împreună eficient. Uneori, oamenii sunt angajați din exterior organizația; totuși, de cele mai multe ori, se pun grupuri de inginerie software împreună de la angajații actuali care au experiență în alte proiecte. Cu toate acestea, managerii au rareori o mână complet liberă în selecția echipei. De multe ori trebuie folosiți persoanele disponibile în companie, chiar și atunci când este posibil să nu fie oameni ideali pentru slujbă.

După cum am discutat în secțiunea 22.2.1, mulți ingineri software sunt motivați în primul rând Prin munca lor, grupurile de dezvoltare software, prin urmare, sunt adesea compuse din oameni care au propriile idei despre modul în care ar trebui rezolvate problemele tehnice. Acest lucru se reflectă în problemele raportate în mod regulat de ignorare a standardelor de interfață,

sistemele fiind re-proiectate pe măsură ce sunt codificate, înfrumusețări inutile ale sistemului, și așa mai departe.

Un grup care are personalități complementare poate funcționa mai bine decât un grup care

este selectat numai pe baza abilităților tehnice. Oamenii care sunt motivați de muncă sunt Persoanele care se auto-orientează vor fi probabil cel mai bine în a împinge munca înainte pentru a termina treaba. Oamenii orientați spre interacțiune ajută la facilitarea comunicării în cadrul grupului. Cred că este în mod special important pentru a avea oameni orientați spre interacțiune într-un grup. Le place să vorbească oameni și pot detecta tensiunile și dezacordurile într-un stadiu incipient, înainte ca acestea să se întâmple un impact serios asupra grupului.

În studiul de caz din Figura 22.10, am sugerat modul în care Alice, managerul de proiect, a încercat să creeze un grup cu personalități complementare grupul are un amestec bun de oameni orientați spre interacțiune și sarcini, dar am discutat deja, în Figura 22.8, modul în care personalitatea auto-orientată a lui Dorothy a provocat probleme pentru că nu a făcut munca pe care o aștepta. Rolul lui Fred cu jumătate de normă grupul ca expert în domeniu ar putea fi, de asemenea, o problemă. El este cel mai mult interesat de

provocări tehnice, deci este posibil să nu interacționeze bine cu alți membri ai grupului faptul că el nu face întotdeauna parte din echipă înseamnă că este posibil să nu se raporteze bine la obiectivele echipei.

Uneori este imposibil să alegeți un grup cu personalități complementare.

Dacă acesta este cazul, managerul de proiect trebuie să controleze grupul astfel încât persoana respectivă

obiectivele nu primează asupra obiectivelor organizaționale și de grup. Acest control este mai ușor de realizat dacă toți membrii grupului participă la fiecare etapă a proiectului.

Inițiativa individuală este cel mai probabil atunci când membrii grupului primesc instrucțiuni fără

fiind conștienți de rolul pe care îl joacă sarcina lor în proiectul general.

De exemplu, să spunem că unui inginer software i se oferă un design de program pentru codificare și

observă ceea ce pare a fi posibilele îmbunătățiri care ar putea fi aduse proiectării.

Dacă el sau ea pune în aplicare aceste îmbunătățiri fără a înțelege rațiunea designul original, orice modificare, deși bine intenționată, ar putea avea implicații adverse pentru alte părți ale sistemului. Dacă toți membrii grupului sunt implicați în de la început, vor înțelege de ce au fost luate decizii de proiectare.

Apoi, se pot identifica cu aceste decizii, mai degrabă decât să le opună.

22.3.2 Organizarea grupului

Modul în care este organizat un grup afectează deciziile luate de acel grup, modalitățile prin care se schimbă informațiile și interacțiunile dintre dezvoltare grup și operatori externi de proiect. Întrebări organizaționale importante pentru managerii de proiect includ:

1. Ar trebui ca managerul de proiect să fie liderul tehnic al grupului?

liderul sau arhitectul de sistem este responsabil pentru deciziile tehnice critice luate

în timpul dezvoltării software-ului. Uneori, managerul de proiect are abilitatea și experiență pentru a prelua acest rol. Cu toate acestea, pentru proiecte mari, cel mai bine este să numiți

un inginer superior care să fie arhitectul proiectului, care își va asuma responsabilitatea conducere tehnică.

2. Cine va fi implicat în luarea deciziilor tehnice critice și cum vor fi acestea? Deciziile vor fi luate de către arhitectul de sistem, managerul de proiect, sau ajungând la un consens între o gamă mai largă de membri ai echipei?

3. Cum vor avea loc interacțiunile cu părțile interesate externe și conducerea superioară a companiei?

În multe cazuri, managerul de proiect va fi responsabil pentru aceste interacțiuni, asistate de arhitectul de sistem dacă există. Cu toate acestea, un modelul organizațional alternativ este de a crea un rol dedicat preocupat de legătură externă și desemnează pe cineva cu abilități de interacțiune adecvate acel rol.

4. Cum pot grupurile să integreze oameni care nu sunt colocați? Acum este obișnuit pentru grupuri pentru a include membri din diferite organizații și oameni pentru a lucra de acasă, precum și într-un birou comun. Acest lucru trebuie luat în considerare în procesele decizionale de grup.

5. Cum pot fi împărtășite cunoștințele între grupuri? Organizația grupului afectează partajarea informațiilor, deoarece anumite metode de organizare sunt mai bune pentru partajare decât alții. Cu toate acestea, ar trebui să evitați prea multe schimburi de informații ca oameni devin supraîncărcați și informațiile excesive îi distrag de la munca lor.

Grupurile mici de programare sunt de obicei organizate într-un mod destul de informal liderul grupului se implică în dezvoltarea de software împreună cu ceilalți membri ai grupului.

Într-un grup informal, munca care trebuie efectuată este discutată de grup ca o întreg, iar sarcinile sunt alocate în funcție de abilitate și experiență

membrii grupului pot fi responsabili pentru proiectarea arhitecturală. Totuși, detaliat proiectarea și implementarea sunt responsabilitatea membrilor echipei care este alocată la o anumită sarcină.

Grupurile de programare extremă (Beck, 2000) sunt întotdeauna grupuri informale entuziaștii susțin că structura formală inhibă schimbul de informații. În XP, mulți

decizii care sunt de obicei privite ca decizii de conducere (cum ar fi decizii privind) program) sunt transferate membrilor grupului. Programatorii lucrează împreună în perechi la elaborează codul și își asumă responsabilitatea comună pentru programele dezvoltate.

Grupurile informale pot avea mare succes, în special atunci când majoritatea membrilor grupului sunt experimentați și competenți. Un astfel de grup ia decizii prin consens, care îmbunătățește coeziunea și performanța. Cu toate acestea, dacă un grup este format în principal din

membrii neexperimentați sau incompetenți, informalitatea poate fi o piedică pentru că nu există o autoritate clară pentru a direcționa munca, provocând o lipsă de coordonare între membrii grupului și, eventual, eventualul eșec al proiectului.

Grupurile ierarhice sunt grupuri care au o structură ierarhică cu grupul lider în vârful ierarhiei. El sau ea are o autoritate mai formală decât grupul membri și astfel își pot direcționa munca. Există o structură organizațională clară și

deciziile sunt luate spre vârful ierarhiei și implementate de oameni
comunicațiile sunt în primul rând instrucțiuni de la senior
personal și există relativ puțină comunicare „ascendentă” de la nivelurile inferioare la
nivelurile superioare din ierarhie.

Această abordare poate funcționa bine atunci când o problemă bine înțeleasă poate fi ușor divizată
în subprobleme cu soluții de subprobleme dezvoltate în diferite părți ale În aceste situații, o
comunicare relativ mică între ierarhie

Cu toate acestea, astfel de situații sunt relativ rare în ingineria software
din următoarele motive:

1. Schimbările aduse software-ului necesită adesea modificări ale mai multor părți ale sistemului
și acest lucru necesită discuții și negocieri la toate nivelurile ierarhiei.

2. Tehnologiile software se schimbă atât de repede încât mai mulți angajați juniori știu adesea
mai multe

despre tehnologie decât personalul cu experiență. Comunicațiile de sus în jos pot
înseamnă că managerul de proiect nu află despre oportunitățile de utilizare
noi tehnologii. Mai mulți angajați juniori pot deveni frustrați din cauza a ceea ce
ei văd că tehnologiile de modă veche sunt folosite pentru dezvoltare.

Organizațiile grupului democratic și ierarhic nu recunosc oficial acest lucru
pot exista diferențe foarte mari în ceea ce privește abilitatea tehnică între membrii grupului
cei mai buni programatori pot fi de până la 25 de ori mai productivi decât cei mai răi
programatori.

Este logic să folosești cei mai buni oameni în cel mai eficient mod și să le oferi
cu cât mai mult sprijin posibil. Un model organizațional timpuriu care a fost destinat
echipa principală de programatori a oferit acest sprijin.

Pentru a face cea mai eficientă utilizare a programatorilor cu înaltă calificare, Baker (1972)
și alții (Aron, 1974; Brooks, 1975) au sugerat că ar trebui construite echipe
în jurul unui programator șef individual, extrem de calificat
echipa șefă de programatori este că personalul calificat și cu experiență ar trebui să fie
responsabil

pentru toate dezvoltările de software. Nu ar trebui să fie preocupați de rutină
contează și ar trebui să aibă un bun sprijin tehnic și administrativ pentru activitatea lor.

Ar trebui să se concentreze asupra software-ului care urmează să fie dezvoltat și să nu petreacă
mult timp

întâlniri externe.

Cu toate acestea, organizația șefă a echipei de programatori este, în opinia mea, supradependentă

Pe programatorul șef și asistentul lor. Alți membri ai echipei care nu sunt dați

o responsabilitate suficientă poate deveni demotivată deoarece consideră că sunt abilitățile lor

Nu au informațiile necesare pentru a face față dacă lucrurile merg prost și nu sunt

dat posibilitatea de a participa la luarea deciziilor. Există proiecte semnificative

riscurile asociate acestei organizații de grup și acestea pot depăși orice beneficii

că acest tip de organizație ar putea aduce.

22.3.3 Comunicații de grup

Este absolut esențial ca membrii grupului să comunice eficient și eficient între ei și cu alți actori ai proiectului. Membrii grupului trebuie să facă schimb informații cu privire la stadiul muncii lor, deciziile de proiectare care au fost luate și modificări ale deciziilor anterioare de proiectare. Trebuie să rezolve problemele care apar cu furnizorilor și să informeze acești operatori cu privire la modificările aduse sistemului, grupului, O bună comunicare ajută, de asemenea, la consolidarea coeziunii grupului.

Membrii grupului ajung să înțeleagă motivațiile, punctele forte și punctele slabe ale alte persoane din grup.

Eficacitatea și eficiența comunicațiilor este influențată de:

1. **Dimensiunea grupului** Pe măsură ce un grup devine mai mare, membrii comunică mai greu

Numărul de legături de comunicare unidirecțional este $n * (n - 1)$, unde n este dimensiunea grupului, deci, cu un grup de opt membri, sunt 56 posibile căi de comunicare. Aceasta înseamnă că este foarte posibil ca unii oameni vor comunica rareori între ei. Diferențe de stare între grup membrii înseamnă că comunicațiile sunt adesea unidirecționale. Managerii și cu experiență inginerii tind să domine comunicațiile cu personal mai puțin experimentat, care poate fi reticent în a începe o conversație sau a face observații critice.

2. **Structura grupului** Oamenii din grupurile structurate informal comunică mai mult Efectiv decât persoanele din grupuri cu o structură formală, ierarhică grupuri, comunicațiile tind să curgă în sus și în jos ierarhia la același nivel s-ar putea să nu vorbească între ei. Aceasta este o problemă specială într-un

proiect mare cu mai multe grupuri de dezvoltare. Dacă oamenii lucrează la diferite subsistemele comunică numai prin intermediul managerilor lor, apoi există mai multe probabil să fie întârzieri și neînțelegeri.

3. **Compoziția grupului** Persoane cu aceleași tipuri de personalitate (discutate în Secțiunea 22.2) se poate ciocni și, ca urmare, comunicările pot fi inhibate. Comunicarea este de obicei mai bună și în grupurile mixte (Marshall și Heslin, 1975) decât în grupurile de sex unic. Femeile sunt adesea mai orientate spre interacțiune decât bărbații și pot acționa ca controlori de interacțiune și facilitatori pentru grup.

4. **Mediul fizic de lucru** Organizarea locului de muncă este una majoră factor în facilitarea sau inhibarea comunicărilor. Consultați pagina web a cărții pentru mai multe informații.

5. **Canalele de comunicare disponibile** Există multe forme diferite de comunicare - față în față, mesaje e-mail, documente oficiale, telefon, și tehnologii Web 2.0 precum rețelele sociale și wiki-urile echipele de proiect se distribuie treptat, cu membrii echipei care lucrează la distanță, trebuie să utilizați o gamă largă de tehnologii pentru a facilita comunicații.

Managerii de proiect lucrează de obicei la termene limitate și, prin urmare, pot încerca să o facă folosiți canale de comunicare care nu-și ocupă prea mult timp.

Prin urmare, bazați-vă pe întâlniri și documente oficiale pentru a transmite informații către proiect personal și părțile interesate. Deși aceasta poate fi o abordare eficientă a comunicării din perspectiva managerului de proiect, nu este de obicei foarte eficientă. Există adesea motive întemeiate pentru care oamenii nu pot participa la întâlniri și, prin urmare, nu aud prezentarea.

Documentele lungi nu sunt adesea citite niciodată, deoarece cititorii nu știu dacă documentele sunt relevante. Când sunt produse mai multe versiuni ale aceluiași document, cititorii îl găsesc dificil de urmărit modificările.

Comunicarea eficientă se realizează atunci când comunicațiile sunt bidirecționale și persoanele implicate pot discuta probleme și informații și pot stabili o înțelegere comună de propuneri și probleme. Acest lucru se poate face prin întâlniri, deși acestea sunt adesea dominate de personalități puternice. Este uneori imposibil să organizați întâlniri cu preaviz scurt. Tot mai multe echipe de proiect includ membri la distanță, ceea ce face și întâlnirile mai dificile.

Pentru a contracara aceste probleme, puteți utiliza tehnologii web precum wikis și bloguri pentru a sprijini schimbul de informații. Wikii susțin colaborarea crearea și editarea de documente și bloguri susțin discuții pe subiecte întrebări și comentarii făcute de membrii grupului. Wikis-urile și blogurile permit proiectul membrii și reactivii externi pentru a face schimb de informații, indiferent de locație. Acestea ajută la gestionarea informațiilor și la urmărirea firelor de discuții, care devin adesea confuze atunci când sunt efectuate prin e-mail. Puteți utiliza și instant mesaje și teleconferințe, care pot fi ușor aranjate, pentru a rezolva problemele care au nevoie de discuții.

PUNCTE CHEIE

1. bună gestionare a proiectelor software este esențială dacă vor fi proiectele de inginerie software dezvoltat în termen și în limita bugetului.
2. Gestionarea software-ului este diferită de alte managementuri ingineresti. Software-ul este intangibil. Proiectele pot fi noi sau inovatoare, astfel încât nu există un corp de experiență care să le ghideze procesele software nu sunt la fel de mature ca procesele de inginerie tradiționale.
3. Gestionarea riscurilor este acum recunoscută drept una dintre cele mai importante sarcini de gestionare a proiectelor.
4. Gestionarea riscurilor implică identificarea și evaluarea riscurilor majore ale proiectului pentru stabilirea probabilității ca acestea să apară și consecințele pentru proiect dacă acest risc apare. Ar trebui să faceți planuri pentru a evita, gestiona sau face față riscurilor probabile dacă sau când apar.

5. Oamenii sunt motivați de interacțiunea cu alte persoane, de recunoașterea managementului șicolegii lor și prin faptul că li se oferă oportunități de dezvoltare personală.
6. Grupurile de dezvoltare software ar trebui să fie destul de mici și coezive Factorii cheie care influențează eficiența unui grup sunt oamenii din acel grup, așa cum este organizat și comunicarea între membrii grupului.
7. Comunicațiile în cadrul unui grup sunt influențate de factori precum statutul grupului membrii, dimensiunea grupului, compoziția de gen a grupului, personalități și canalele de comunicare disponibile.