

Tema Inginerie Software

Managementul si Problemele Productiei Software

student: Abibula E. Erhan
grupa: 442A

1. Managementul productiei software.....	3
1.1. Initierea si definirea domeniului de aplicare.....	4
1.2. Planificarea proiectului de soft.....	5
1.3. Punerea in aplicare a proiectului software.....	7
1.4. Monitorizarea si evaluarea.....	10
1.5. Finalizarea proiectului.....	10
2. Probleme in productia software.....	11
2.1. Satisfacerea cerintelor utilizatorilor.....	11
2.2. Costul.....	11
2.3. Performanta.....	13
2.4. Portabilitatea.....	13
2.5. Fiabilitatea.....	14
Bibliografie.....	14

1. Managementul productiei software

Managementul productiei software reprezinta procesul de gestionare al software-ului, care este realizat si pus in aplicare ca un produs, tinandu-se cont de durata ciclului de viata si de cerintele consumatorilor.

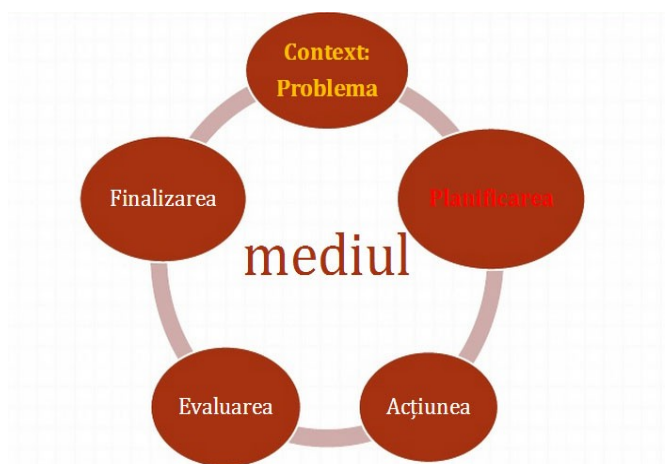
Pentru a face din dezvoltarea si vinderea unui produs software o afacere de success trebuie intelese cerintele pietei si identificate oportunitatile in realizarea unui produs cat mai apropiat de nevoile clientilor.

Aria potentialilor cumparatori nu este reprezentata doar de persoane fizice, ci si de companiile de produse hardware. Este stiut faptul ca fara un soft adecvat un produs hardware nu poate functiona.

Principalele etape de gestionare a productiei unui produs software sunt:

1. Initierea si definirea domeniului de aplicare
2. Planificarea proiectului de soft
3. Punerea in aplicare a proiectului software
4. Monitorizarea si evaluarea.
5. Finalizarea proiectului

Ciclul de viata al unui proces:



Imagine luata de pe:

http://www.academia.edu/1990981/Introducere_in_managementul_proiectelor

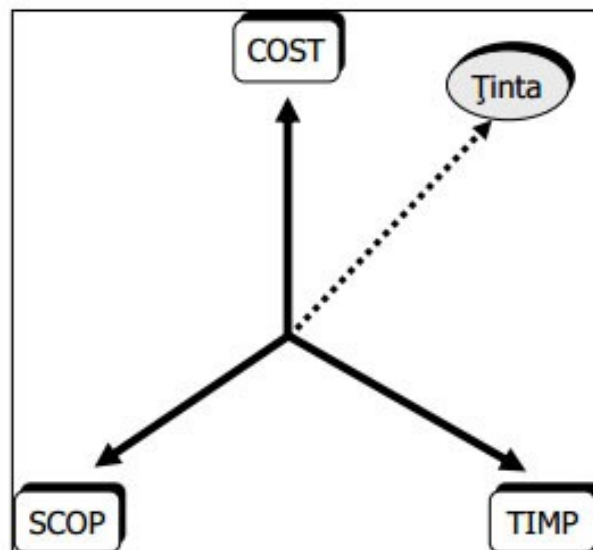
1.1 Initierea si definirea domeniului de aplicare

In faza de initiere se autorizeaza dezvoltarea proiectului, se stabileste managerul de proiect si sunt definite obiectivele si scopurile proiectului. Totodata, se elaboreaza planul de management al proiectului. Dupa ce este aleasa solutia optima se urmareste formularea obiectivelor, si anume:

- Performantele tehnice ale produsului software
- Termenul limita de finalizare a proiectului
- Costul
- Criterii de acceptare

Complexitatea realizarii unui produs software creste odata cu:

- Cresterea dimensiunii programului
- Cresterea calitatii produsului software
- Antrenarea unui volum mare de resurse umane si resurse tehnice.



Imagine luata de pe:

http://ccimn.ulbsibiu.ro/documente/carti/introducere_in_managementul_proiectelor.pdf

La realizarea unui produs software se urmareste atat cresterea calitatii produsului cat si reducerea costului si a termenului de realizare.

1.2. Planificarea proiectului de soft

Planificarea proiectului de soft consta in stabilirea obiectivelor si a sarcinilor proiectului, alocarea resurselor si stabilirea termenului de realizare.

Determinarea efortului de realizare a proiectului software se reduce la evaluarea urmatoarelor elemente de baza:

- Dimensiunea programului
- Complexitatea programului
- Gradul de noutate al problemei
- Gradul de reutilizare a produselor program deja existente

Planificarea realizarii unui produs software se desfasoara pe 3 niveluri:

- Strategic
- Tactic
- Operativ

La nivel strategic se urmareste stabilirea tipurilor de produse program ce trebuie elaborate in raport cu resursele disponibile si fondul de programe deja existent, stabilirea nivelului tehnic si calitativ al produselor software care depinde atat de competenta profesionala a membrilor cat si de dotarea cu tehnica de calcul, stabilirea costurilor si a termenelor de realizare a produselor program. De asemenea, tot aici se stabileste pretul de livrare al produsului.

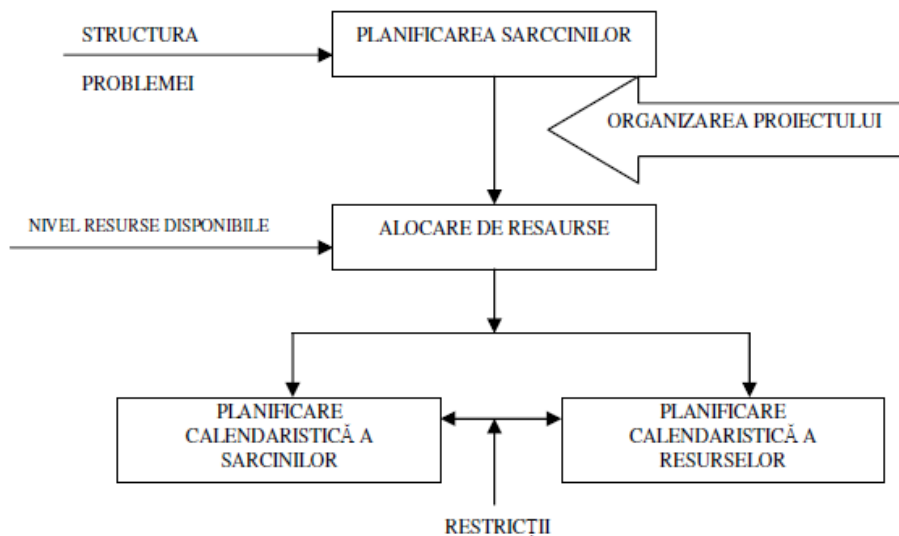
Planificarea tactica este derivata din cea strategica si consta in determinarea pasilor individuali ai planului de lucru. Dezvoltarea acestui proces depinde de urmatoarele surse:

- Experienta autorului planului
- Experienta consultantilor
- Planurile din trecut aflate la dispozitia autorului
- Machete de la diverse planuri
- Planuri de lucru ale altor metodologii de dezvoltare

Etapele planificarii proiectului sunt:

- Planificarea sarcinilor
- Alocarea resurselor
- Planificarea calendaristica

Schema pasilor parcursi este reprezentata in figura:



Imagine luata de pe: <http://ti.utm.md/upload/profs/files/403.pdf>

Planificarea operativa are un rol principal in corectarea planificarii tactice, care pot fi:

- Modificarea sarcinilor pe parcursul desfasurarii proiectului
- Imposibilitatea evaluarii precise a datei de finalizare sau a resurselor
- Alte indisponibilitati neprevazute.

Un alt rol al planificarii operative il reprezinta stabilirea ordinii de executie a diferitelor activitati efectuate la nivelul intregului proiect.

1.3. Punerea in aplicare a proiectului software

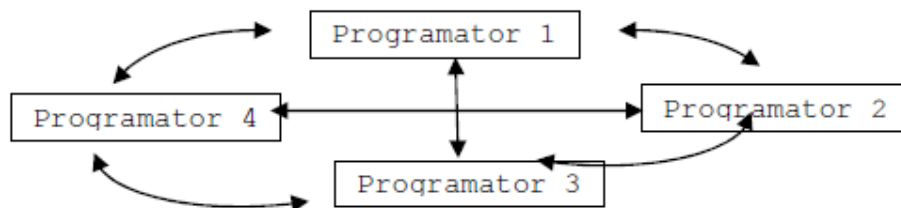
Este faza de realizare a programului prin punerea in aplicare a activitatilor stabilite in etapele precedente. In functie de dimensiunile si complexitatea proiectului se definesc mai multe tehnici de organizare , si anume:

- “Programarea deschisa
- Echipa programatorului sef
- Echipa chirurgicala
- Echipa programatorului sef revizuita”

Programarea deschisa este cea mai simpla forma de organizare si conduce la un mod de lucru democratic. Intreaga echipa are un singur scop, reusita programului.

Sistemul de interactiune in echipa este schitat in figura urmatoare:

Imagine luata de pe:



<http://ti.utm.md/upload/profs/files/403.pdf>

Acest mod de lucru asigura spiritul de echipa, integrarea usoara si rapida a componentelor, identificarea mai rapida a erorilor ce trebuiesc corectate dar contribuie si la transferul de experienta intre membrii echipei.

Echipa programatorului sef are un caracter precis si responsabilitati foarte bine stabilite.

Modul de organizare se bazeaza pe urmatoarele principii:

- Procesul de realizare al programului este structurat pe sarcini precise
- Asigurarea vizibilitatii produsului
- Asigurarea posibilitatii de invatare in cadrul echipei

- Asigurarea conditiei ca cel putin 2 membrii sa cunoasca intreg programul

Schema relatiilor intre membri echipei este data de figura:

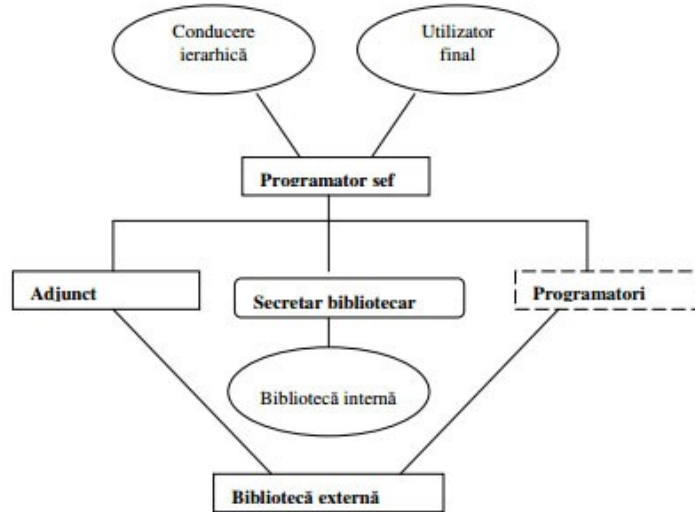


Fig.8.4. Echipa programatorului șef

Imagine luata de pe: <http://ti.utm.md/upload/profs/files/403.pdf>

“Echipa chirurgicala” este compusa dintr-un șef de echipa insotit de un administrator, care, la randul sau, poate fi ajutat de un secretar in cazul problemelor mai complexe. Membrii echipei sunt mai specializati decat in cazul “echipei programatorului șef”.

Schema bloc a organizarii echipei este ilustrata in figura urmatoare:

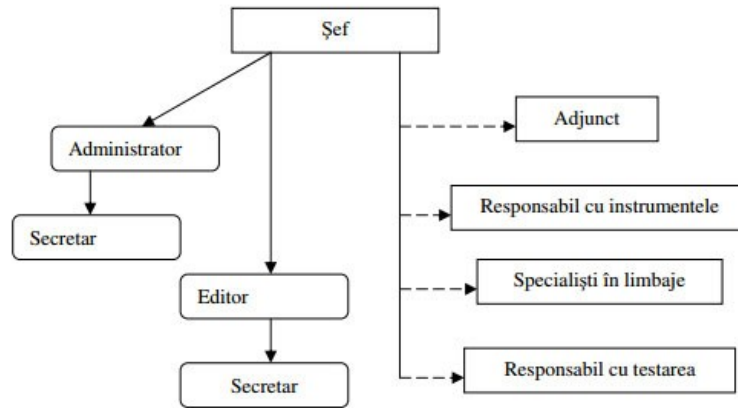


Fig.8.5 Echipa chirurgicală

Imagine luata de pe: <http://ti.utm.md/upload/profs/files/403.pdf>

Nucleul **Echipei programatorului șef revizuita** este format dintr-un conducator de proiect, un adjunct al acestuia, un administrator si un responsabil pentru asigurarea legaturii cu utilizatorul.

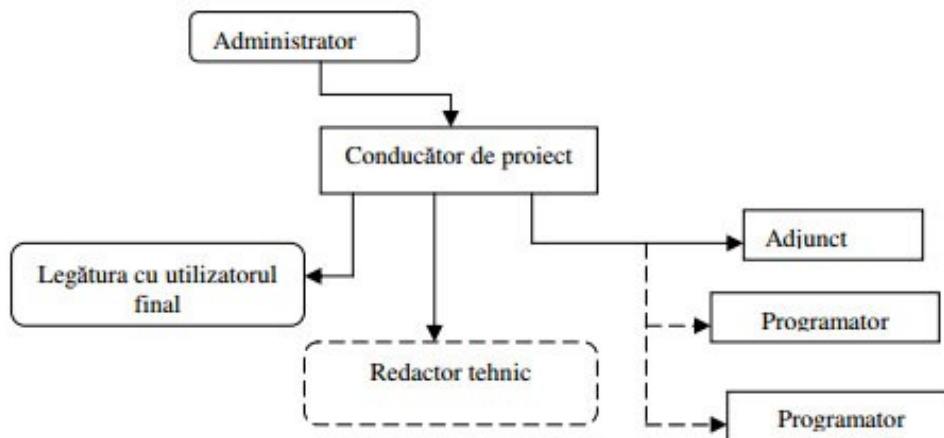


Fig. 8.6. Echipa programatorului șef revizuită

1.4. Monitorizarea si evaluarea

Aceasta etapa are rolul de a asigura atingerea obiectivelor stabilite si se realizeaza in timpul desfasurarii proiectului prin monitorizarea resurselor, a costurilor si a calitatii proiectului. Se urmareste combaterea abaterilor dintre planificarea initiala si cea actuala, pentru a ajusta planul proiectului.

In acest scop sunt utilizate tehnici ca cele descrise mai jos:

- Verificarile tehnice
- Raportarea stadiului proiectului
- Controlul modificarilor

1.5. Finalizarea proiectului

Finalizarea proiectului consta in acceptarea rezultatelor proiectului si terminarea ordonata a acestuia. Etapa de finalizare include:

- Inchiderea administrativa care implica colectarea de informatii pentru a formaliza inchiderea proiectului
- Inchiderea contractului: terminarea si decontarea contractului.

2. Probleme in productia software

Principalele probleme care apar in productia software sunt:

1. Satisfacerea cerintelor utilizatorilor
2. Costul
3. Performanta
4. Portabilitatea
5. Fiabilitatea

2.1. Satisfacerea cerintelor utilizatorilor

[“Nici un produs, deci nici produsele software, nu vor fi solicitate și utilizate dacă ele nu răspund unor nevoi ale utilizatorilor. Cu cât sunt mai bine acoperite cerințele celor care beneficiază de facilitățile produsului software respectiv și cu cât produsul va răspunde mai bine acestor solicitări, cu atât cererea pentru sistemul (produsul) respectiv va fi mai mare.”]

2.2. Costul

S-a observat ca peste 33% dintre organizatiile analizate nu deruleaza un proces de estimare al costului de dezvoltare pentru un produs software. In cele mai multe situatii, costul produsului este stabilit de catre directorul companiei pe baza unor criterii care nu tin de justificare economica. De aici se ajunge ca la mai mult de 66% dintre proiectele de productie software sunt intarziate sau depasesc bugetul alocat.

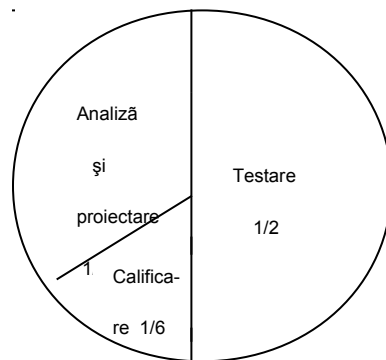
Dificultatea estimarii costului este determinata de mai multe motive:

- In multe situatii nu se colecteaza intr-un mod organizat informatii referitoare la performanta estimarilor din proiectele anterioare, comparativ cu realitatea de la finalizarea proiectului. Lipsa acestor informatii este un impediment important in realizarea estimarilor de efort si de cost pentru proiectele curente

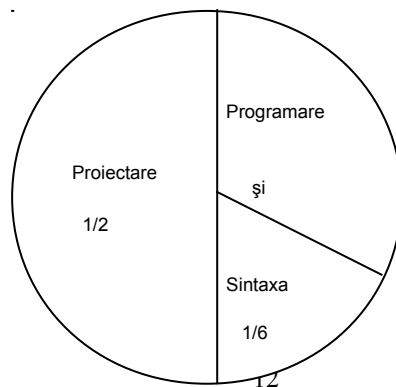
- Sunt situatii in care estimarile au fost realizate, dar sub presiunea timpului si fara a permite o evaluare reala a efortului necesar realizarii produsului
- Modificarile la functionalitatea programului pe parcursul dezvoltarii
- Lipsa de experienta a persoanei care face estimarea acestor costuri
- In estimarea efortului si a timpului de finalizare nu se iau in considerare activitatile de management care insotesc procesul de dezvoltare

Costul software-ului este reprezentat in mare parte de salariile celor implicati in dezvoltarea produsului. Productivitatea medie a unui programator este de 10-20 instructiuni pe zi. Aceasta analiza include documentarea, proiectarea, codificarea si testarea programului.

Figura urmatoare ilustreaza costul testarii care este enorm comparativ cu cel al codificarii programului. Solutia reducerii acestui cost ar fi corectitudinea proiectarii de la inceput, insa erori umane vor aparea tot timpul.



Erorile fiind o problema majora, vom determina momentul cand apar ele. Figura urmatoare ilustreaza acest lucru.



O eroare nedeterminata la timp implica cresterea costului mai mult decat in situatia in care ar fi fost descoperita si inlaturata la timp.

In prezent, nu se mai pune problema erorilor de sintaxa, ele fiind descoperite automat de catre compilatoare si corectate cu usurinta.

Problema majora sunt erorile de proiectare care sunt determinate abia in faza de testare, necesitand reproiectarea sistemului.

2.3. Performanta

Performanta se refera la obtinerea unui semnal de control la iesire intr-un timp cat mai scurt.

Se disting 2 tipuri de programe:

- Lente, dar mici din punctul de vedere al memoriei ocupate
- Rapide, dar de dimensiuni mari

Utilizarea unei memorii mici si cu un timp de executie scurt sunt 2 cerinte contradictorii.

2.4. Portabilitatea

Portabilitatea se refera la capacitatea produsului de a putea fi transferat de pe un tip de calculator pe altul, cu un minim de efort. Odata cu aparitia limbajelor de nivel inalt si folosirea standardelor internationale, in majoritatea aplicatiilor software s-a ajuns la o portabilitate maxima.

Portabilitatea se poate referi si la:

- Adaptabilitatea: capacitatea softului de adaptare la diferite medii
- Usurinta de instalare
- Co-existenta cu alte produse independente
- Efortul necesar pentru a folosi produsul in locul altuia intr-un mediu particular
- Conformitatea

2.5. Fiabilitatea

Fiabilitatea reprezinta capacitatea cu care produsul isi pastreaza nivelul de performanta, pentru o perioada definita de timp.

In acest sens, se deduc urmatoarele atribute ale produsului software:

- Toleranta la defecte: capacitatea produsului de a-si mentine nivelul de performanta in cazul situatiilor neprevazute
- Restabilirea dupa caderi: capacitatea si efortul necesar pentru restabilirea nivelului de performanta al produsului, dupa eventuale caderi
- Maturitatea: atribut bazat pe nr. de caderi raportate la timp, datorate erorilor de soft
- Conformitatea

Bibliografie:

<http://www.techrepublic.com/article/understanding-the-pros-and-cons-of-the-waterfall-model-of-software-development/6118423>

<http://www.linchpinseo.com/the-agile-method>

http://www.dixite.com/docs/spposs/dixite-spposs-en-1_0.pdf

<http://web.eecs.utk.edu/~swain/presentation/DANSESQA.pdf>

https://www.google.ro/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&ved=0CCwQFjAA&url=http%3A%2F%2Fwww.econ.ubbcluj.ro%2Fprograme_analitice%2F2012%2FMASTER%2520programe%2520analitice%25202012%25202013%2FE-business%2FAN%2520I%2FIngineria%2520sistemelor%2520informatice.doc&ei=e9YXUdaaHKWP4gTDw4HgBQ&usg=AFQjCNFipjJGMvOKLxkND4j66GFt7Nlu9g&sig2=EGxTwyC6OYgVelXwPHW7vA&bvm=bv.42080656,d.Yms&cad=rja

http://ro.wikipedia.org/wiki/Managementul_proiectelor

<http://ti.utm.md/upload/profs/files/403.pdf>

http://www.marketwatch.ro/articol/6811/Costul_dezvoltarii_unui_produ_software_ghiceala_sau_estimare/

<http://blogu.lu/kassak/sistemele-software-probleme-si-perspective/>