

METODOLOGII PENTRU MANAGEMENTUL PROIECTELOR INFORMATICE

*Constanța Bodea, Vasile Bodea, Academia de Studii Economice
București*

Articolul prezintă rezultatele obținute în cadrul proiectului INFOSOC *Sistem de management adaptabil dinamic*, contract 154/2004 în etapa de documentare referitoare la managementul proiectelor informatice. Sunt prezentate două abordări importante MSF - Microsoft Solutions Framework și ORACLE PJM

1. Microsoft Solutions Framework - MSF

Dezvoltarea sistemelor software de mari dimensiuni, complexe este un demers riscant. Statisticile facute asupra unor proiecte de mare anvergura din domeniul tehnologiei informatiei arata ca o buna parte dintre acestea esueaza. Unele cauze ale esecului acestor proiecte sunt:

- Cerintele beneficiarului se schimba in mod constant
- Specificatiile sunt volatile sau incomplete
- Slaba calitate a codului
- Aria cuprinsa de proiect este prea mare
- Resurse umane participante la proiect inadecvate
- Procese slabe (poor process)
- Teluri neclare

Pentru a adresa aceste probleme ale dezvoltarii proiectelor software, Microsoft Consulting Servces a conceput Microsoft Solutions Framework- MSF pe baza experientei si practicilor pozitive ale grupurilor de dezvoltare pe produse din cadrul firmei Microsoft, ale partenerilor si clientilor lor. Acest cadru poate fi folosit pentru managementul cu succes al proiectelor software in care sunt implicati mai multi dezvoltatori.

MSF este o colecție de modele, principii și practici care ajută organizațiile să devină mai eficiente în crearea și utilizarea tehnologiilor informatice pentru rezolvarea problemelor pe care le au. MSF ajută prin punerea la dispoziție a unor instrumente de măsurare a progresului proiectelor și printr-un proces adaptabil de proiect management care este îndeajuns de flexibil pentru a face față schimbărilor rapide de cerințe ale mediilor de afaceri moderne.

MSF este compus din următoarele componente (figura 1):

- Principiile fundamentale ale MSF.
- Modelele MSF
- Disciplinele MSF.
- Conceptele cheie MSF
- Cele mai bune practici MSF
- Recomandările MSF

Principiile fundamentale ale MSF. Reprezintă valorile și standardele comune tuturor elementelor. Principalele principii MSF sunt:

- Comunicarea deschisă
- Viziunea unică
- Motivarea și responsabilizarea membrilor
- Stabilirea clară a răspunderii
- Accentul pus pe valoarea adăugată în afaceri
- Flexibilitatea. Așteptarea schimbărilor
- Investiția în calitate
- Învățarea din experiență

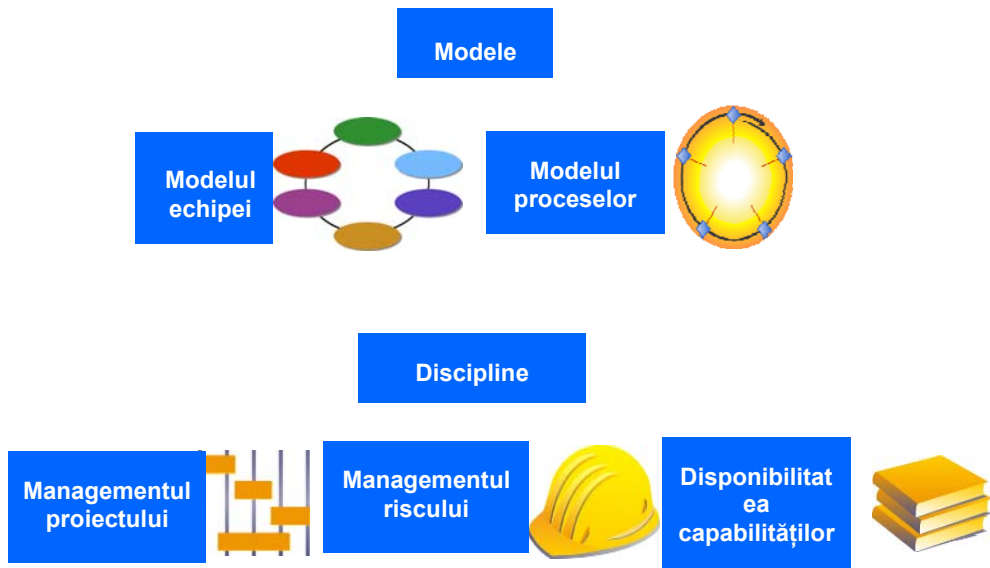


Figura 1 - Componentele MSF

Modelele MSF. Principalele modele sunt:

- modelul echipei
- modelul proceselor

1.1 Modelul echipei

Figura 2 prezintă modelul MSF al echipei de proiect. Modelul asigură o structură flexibilă pentru organizarea echipelor implicate în proiect. Acest model indică cu claritate rolurile, responsabilitățile și telurile ce vor duce la succesul echipei și crește nivelul colaborării membrilor echipei printr-o impunerea unei relații de egalitate între membrii echipei. Flexibilitatea acestui model permite adaptarea la diferite nivele ale ariei de cuprindere a proiectului, ale dimensiunii echipelor și ale pregătirii profesionale ale membrilor echipelor. Folosirea acestui model și a principiilor și practicilor aferente ajută la crearea unor echipe mai eficiente, mai entuziaste, mai rezistente în fața greutăților proiectului și mai de succes.

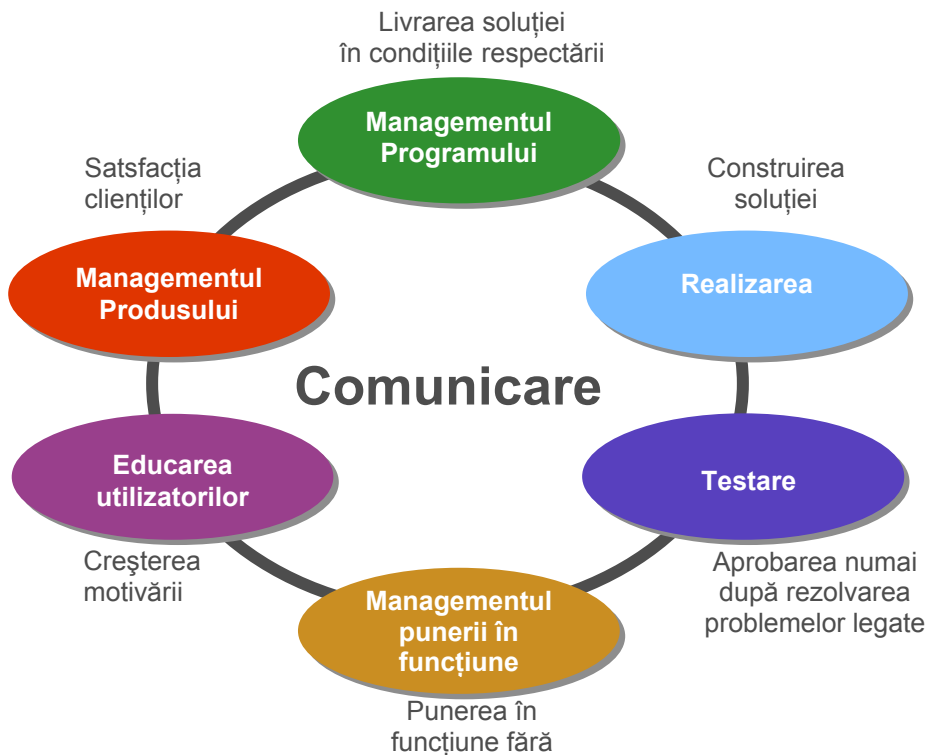


Figura 2 – Modelul echipei proiectului

Modelul descrie cum trebuie să se organizeze echipele și ce principii trebuie să urmeze pentru a avea succes în activitatea de dezvoltare de software. Modelul are o natură specifică, dar ca parte integrantă a cadrului MSF, trebuie privit ca punctul de plecare în realizarea proiectului. Echipele participante la proiect pot implementa în mod diferit aspectele cadrului MSF în funcție de aria de cuprindere a proiectului, dimensiunea echipei și pregătirea profesională a membrilor echipei.

Pentru ca un proiect să aibă succes trebuie ca anumite responsabilități specifice să fie asumate și ca obiectivele propuse să fie atinse. Aceste responsabilități și obiective asigură direcționarea continuă a membrilor echipei. Responsabilitățile și obiectivele cele mai importante sunt:

- **Satisfacția clientului.** Proiectele trebuie să răspundă necesităților beneficiarului și utilizatorilor sistemului pentru a avea succes. Este posibil ca o echipă să se încadreze în constrângerile bugetului de bani și de timp, dar să nu reușească să își atingă obiectivele deoarece necesitățile beneficiarului nu au fost adresate cu succes.
- **Livrarea proiectului cu respectarea constrângerilor impuse.** Cele mai multe proiecte măsoară succesul folosind metrici de tipul “încadrare în bugetul de timp și de bani”.
- **Proiectul să se livreze conform specificațiilor bazate pe cerințele utilizatorilor aplicației.** Specificațiile funcționale descriu în detaliu ceea ce trebuie livrat de echipa beneficiarului. Aceste specificații reprezintă un contract între echipă și beneficiar referitor la ce trebuie construit și baza principiului “Doing what we say we will do”.
- **Livrarea proiectului se va face după ce au fost identificate și tratate toate problemele.** Orice software este livrat cu defecte. Obiectivul echipei este de a se asigura că aceste defecte sunt identificate și adresate înainte de livrarea produsului. Tratarea unui defect poate implica corectarea defectului sau documentarea unor soluții de evitare a defectului. Livrarea unui defect cunoscut, care a fost tratat prin documentarea unei modalități de evitare a lui este preferabilă livrării unui produs care conține defecte neidentificate ce vor “surprinde” echipa și beneficiarul la momentul descoperirii lor.
- **Sporirea performanțelor utilizatorilor aplicației.** Un produs va avea succes numai dacă va spori performanțele și eficiența în munca a utilizatorilor săi. Livrarea unui produs prin deficiențe, dar care nu poate fi utilizat de către utilizatorii săi direcți este considerată un eșec.
- **Instalare și mentenanță usoare.** Eficacitatea instalării afectează în mod direct modul în care gândesc utilizatorii și beneficiarul despre calitatea produsului. De exemplu, un program de instalare defectuos va face ca utilizatorii aplicației să creadă că aplicația în sine este la fel de plină de defecte. Echipa trebuie să facă mai mult decât simpla instalare a produsului: trebuie să instaleze produsul printr-un proces fără erori și apoi să asigure suport și mentenanță pentru produs.

Modelul echipei asigura atingerea obiectivelor mentionate mai sus prin asignarea taskurilor la sase roluri din cadrul echipei: Product Management, Program Management, Development, Testing, User Education si Release Management.

Fiecare obiectiv are nevoie de o disciplina diferita, asa ca fiecare rol din cadrul echipei cuprinde o disciplina diferita. Oamenii care indeplinesc diferitele roluri din cadrul echipei trebuie sa aiba competenta si perspectiva necesare pentru atingerea obiectivului propus. Tabelul 1 face corespondenta intre cele sase roluri ale modelului si cele sase obiective ale unei echipe eficiente.

Deoarece fiecare obiectiv este critic pentru succesul proiectului, rolurile corespondente sunt vazute ca avand aceeasi importanta, cu aceeasi putere in loarea deciziilor. Nu exista un "lider" de proiect, ci numai o echipa care stie ce trebuie facut.

Tabelul 1

Obiectiv	Rol
Satisfacerea clientului	Product Management
Livrarea proiectului cu respectarea constrangerilor impuse	Program Management
Proiectul sa se livreze conform specificatiilor bazate pe cerintele utilizatorilor aplicatiei	Development
Livrarea produsului se va face dupa ce au fost identificate si tratate toate problemele	Testing
Sporirea performantelor utilizatorilor aplicatiei	User Education
Instalare si mentenanta	Release Management

1.2 Modelul proceselor. Este prezentat în figura 3.



Figura 3 – Modelul proceselor

Modelul proceselor asigura structura organizatorica si cadrul necesar de-a lungul ciclului de viata al unui proiect; ciclul de viata este bazat pe termene de livrare, este iterativ si flexibil. Acest model descrie fazele, termenele de livrare, activitatile, documentele si rezultatele aferente unui proiect de dezvoltare a unei aplicatii software, precum si relatiile dintre aceste elemente si rolurile definite la modelul echipei proiectului. Utilizarea acestui model amelioreaza controlul asupra proiectului, minimizeaza riscurile, duce la cresterea calitatii si la reducerea timpului in care proiectul va fi livrat.

Modelul proceselor indeplineste o functie cheie in dezvoltarea proiectului prin specificarea exacta a ce activitati trebuie realizate si cand trebuie realizate aceste activitati. Modelul are o stansa legatură cu:

- modelul echipei
- practicile si principiile constituyente ale MSF

De cele mai multe ori abordările tradiționale ale dezvoltării de software, precum modelele cascada și spirala, nu fac față cerințelor actuale ale dezvoltării de nivel întreprinderii. În cazul modelului cascada, un proiect progresează prin pași secvențiali, începând cu conceptul inițial și terminând cu sistemul de testare. Acest model identifică termenele de predare de-a lungul vieții proiectului și le folosește ca puncte de tranziție în următoarea etapă și ca puncte în care se face evaluarea proiectului.

Acest model funcționează cu succes în cazul proiectelor pentru care cerințele pot fi specificate cu succes la începutul proiectului, dar pot apărea dificultăți în cazul proiectelor complexe la care specificațiile se pot schimba de-a lungul timpului de viață al proiectului. În plus, practicantii acestui model se bazează extensiv pe volume mari de documentație și pe un singur proces de revizuire / ajustare pentru fiecare fază a proiectului. Aceste două practici ale modelului cascada conduc de obicei la paralizii datorate procesului de analiză supradimensionat, precum și la relații de adversitate între dezvoltatori, beneficiari și utilizatori.

În cazul modelului spirala, aplicația evoluează de-a lungul unui număr de iterații. Primele iterații în cadrul acestui model asigură o definiție din ce în ce mai exactă a produsului, următoarele iterații adăugând diferite "features" și funcționalități aplicației. Modelul spirala caută să confrunte riscurile cât mai devreme în ciclul de viață al proiectului software, adresându-le în "releas"-urile timpurii ale produsului.

Datorită naturii iterative, modelul spirala suportă ajustări creative de-a lungul vieții proiectului, ducând astfel la o evoluție a calității produsului (se speră la o ameliorare a calității). Natura iterativă a proceselor spirala necesită o eficiență ridicată a proceselor și automatizărilor referitoare la documentație. În practică însă, beneficiarii și utilizatorii sistemului pot dezvolta un sentiment de instabilitate, deoarece schimbările asupra produsului pot fi mai rapide decât puterea lor de asimilare. Multe proiecte dezvoltate cu ajutorul modelului spirala nu au precizat clar un punct de finalizare, astfel încât continua să itereze nedefinit, fără a avea un termen de finalizare financiar sau din punct de vedere al mediului de afaceri.

Modelul proceselor din MSF combina punctele forte ale celor doua modele, asigurand atat beneficiile planificarii prin milestone-uri specifice modelului cascada, cat si beneficiile procesului creativ iterativ al modelului spirala.

Caracteristicile de bază ale modelului proceselor:

- Model pe bază de faze
 - Model pe bază de milestone-uri (romburile care separa fazele procesului)
 - Model iterativ

Model pe bază de faze. Modelul proceselor este compus din cinci faze interrelationate. Fiecare dintre aceste faze reprezinta, de fapt, ceea ce trebuie livrat si pentru care trebuie stabilit un termen de livrare inainte ca procesul de dezvoltare sa poata trece la urmatoarea faza a proiectului.

Model pe bază de milestone-uri. Modelul proceselor este bazat pe milestone-uri; acestea trebuie privite ca puncte de revizuire si sincronizare si nu ca puncte in care aplicatia sau specificatiile vor fi inghetate. Milestone-urile dau posibilitatea echipei sa evalueze progresul proiectului si sa efectueze corectii pe parcursul proiectului, precum ajustarea ariei de cuprindere pentru a reflecta modificarile cerintelor beneficiarului sau reactionarea la riscurile ce se pot materializa de-a lungul vietii proiectului.

Model bazat pe versiuni. Modelul proceselor este un model bazat pe versiuni, in sensul ca este conceput astfel incat sa se repete de-a lungul ciclului de viata al unui produs. Fiecare completare cu succes a modelului permite adaugarea de noi caracteristici si functionalitati care sa satisfaca cerintele schimbatoare ale mediului de afaceri.

1.3 Disciplinele MSF.

1.3.1 Managementul proiectului

Disciplina de management al proiectului este aliniată la corpurile de cunoștințe de referință în managementul proiectelor, respectiv: PMBOK al PMI, ICB al IPMA, Prince2. Disciplina de management al proiectelor în MSF prezintă o serie de caracteristici:

- Managementul proiectelor reprezintă o disciplină structurată în arii de cunoștințe și activități, nu pe roluri sau titluri.
- Cele mai multe dintre responsabilitățile rolului cunoscut ca *Manager de proiect* sunt incluse în clusterul MSF Managementul Programului.
- În proiectele de mari dimensiuni, activitățile de management al proiectului apar la mai multe niveluri.
- Proiectele foarte mari sau proiectele complexe necesită un manager de proiect dedicat sau o echipă de management
- În cadrul MSF accentul se deplasează pe colaborarea dintre roluri, de exemplu luarea deciziilor prin consens. Abordările tradiționale accentuează pe calitatea managerului de proiect ca decident cheie, care deține controlul asupra membrilor echipei.

Figura 4 prezintă responsabilitățile pentru echipa de proiect.

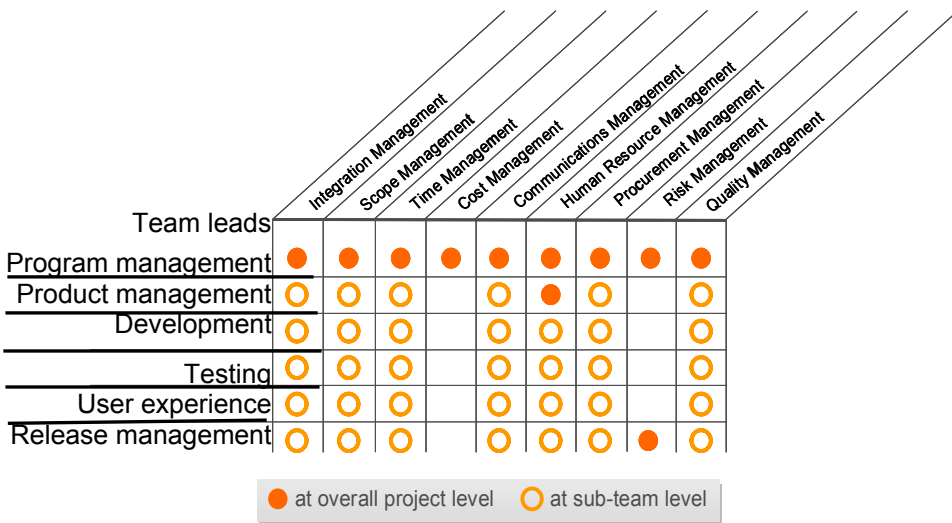


Figura 4 – Responsabilitățile de realizare a activităților de management al proiectului

1.3.2 Managementul riscurilor. Asigura o modalitate structurata si activa de management al riscurilor proiectului. Aceasta disciplina stabileste cadrul necesar luarii unor decizii si realizarii unor activitati “proactive” cu scopul de a evalua permanent problemele potientiale, de a determina care sunt riscurile ce trebuie adresate si de a implementa strategii de tratare a acestor riscuri (figura 5). Folosirea acestui model si a principiilor si practicilor aferente ajuta echipa sa se focalizeze asupra aspectelor cu adevarat importante, sa ia decizii corecte si sa fie pregatita pentru viitor.

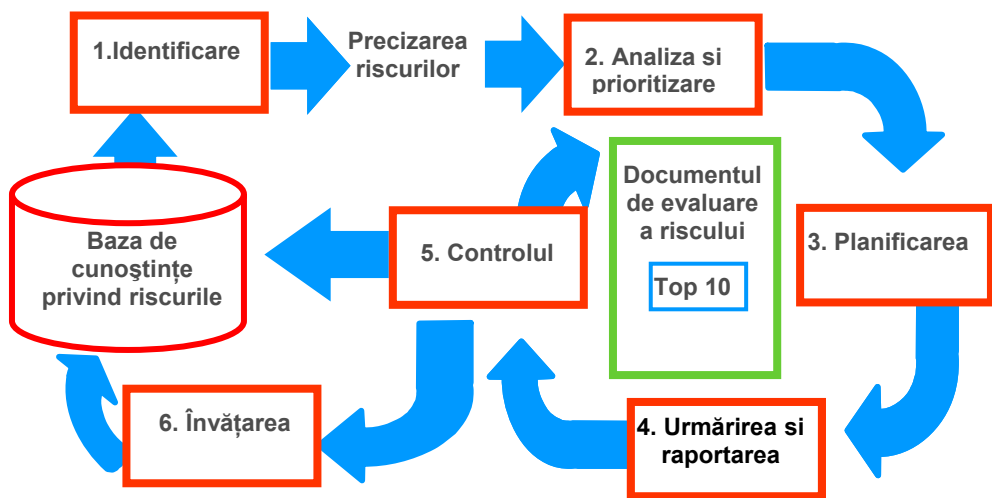


Figura 5 – Procesele de management al riscurilor

1.3.3 Disponibilitatea capabilităților. Figura 6 prezintă managementul disponibilității capabilităților.

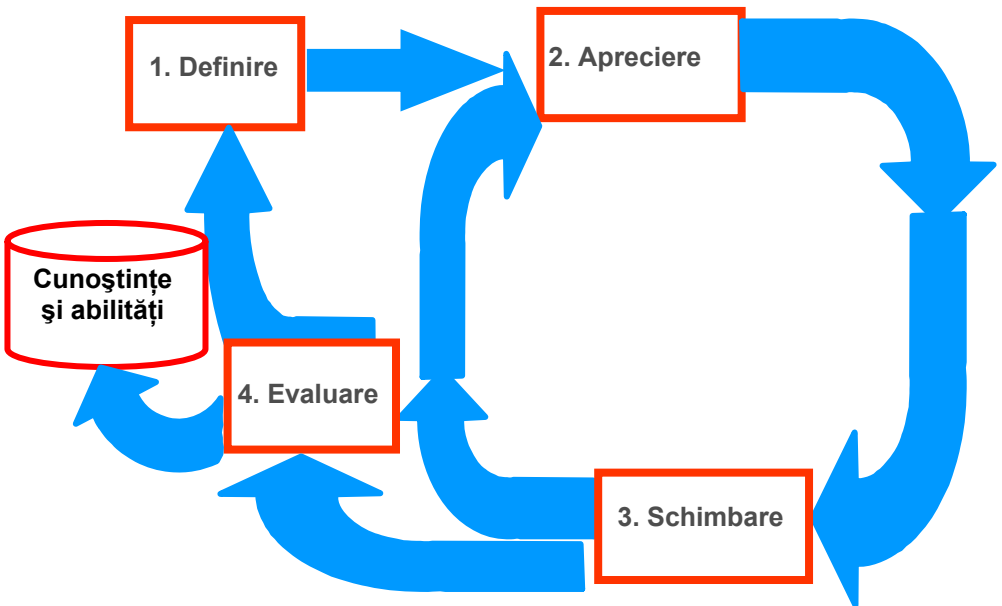


Figura 6 – Managementul disponibilității capabilităților

Proiectarea unui sistem software în cadrul MSF are în componența trei niveluri distincte de proiectare și anume:

- nivel conceptual
- nivel logic
- nivel fizic.

Fiecare parte a procesului de proiectare abordeaza activitatea de proiectare dintr-o perspectiva diferita si defineste solutii diferite, dupa cum se poate vedea in tabelul 2.

Tabelul 2

Tipul muncii de proiectare	Perspectiva	Actiune
Conceptual	Vede problema din perspectiva utilizatorului si a mediului de afaceri.	Defineste problema si solutia prin scenarii.
Logical	Vede solutia din perspectiva echipei care implementeaza proiectul.	Defineste solutia ca un set de servicii ce coopereaza intre ele.
Phisical	Priveste solutia din perspectiva dezvoltatorilor.	Defineste serviciile si tehnologiile solutiei.

Rezultatul obținut la fiecare nivel este utilizat ca intrare pentru nivelul urmator (figura 7):

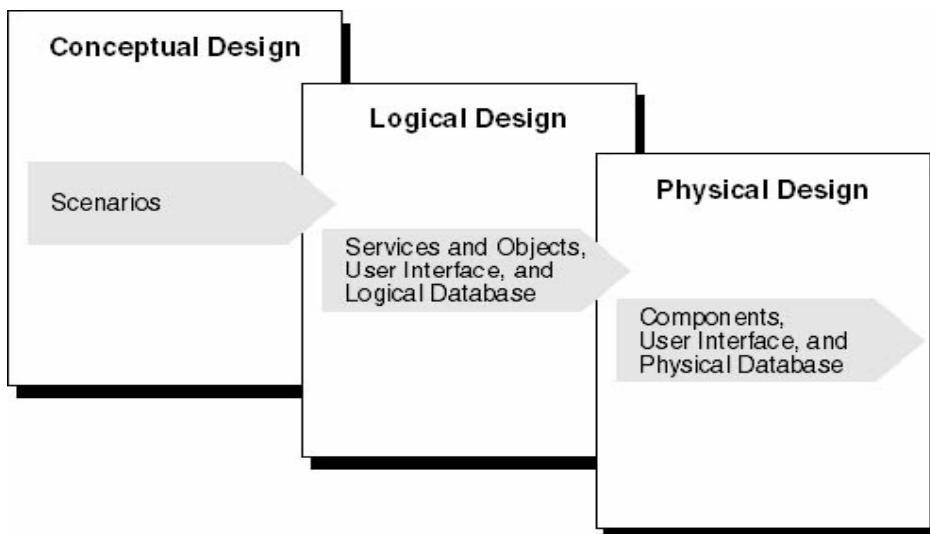


Figura 7 – Nivelurile activității de proiectare

Obiectivele celor trei niveluri de proiectare sunt:

- **Proiectarea conceptuala.** Identifica nevoile afacerii si intelege ce fac utilizatorii si de ce au nevoie. Proiectarea conceptuala genereaza scenarii ce vor reflecta cu completitudine si acuratete cerintele afacerii prin implicarea beneficiarului, utilizatorilor si a tuturor celor interesati de proiect.
- **Proiectarea Logica.** Organizeaza solutia si comunicarea intre elementele sale componente. Proiectarea logica preia problema identificata in faza de proiectare conceptuala si formuleaza un model abstract al solutiei. Conform desenului prezentat mai sus, proiectarea logica preia scenariile generate de proiectarea conceptuala si produce obiecte si servicii, prototipuri de interfete cu utilizatorul si proiectarea logica a bazei de date.
- **Proiectarea fizica.** Outputului generat de proiectarea logica i se aplica constrangeri tehnologice, inclusiv cele ce decurg din tipul de implementare si performantele cerute, specificand astfel detaliile solutiei. Output-ul proiectarii logice este folosit la producerea de componente, specificatii ale interfetei cu utilizatorul si proiectarea fizica a bazei de date.

Proiectarea conceptuala nu ia in considerare tehnologiile necesare construirii solutiei. Ea corespunde schitelor si desenelor initiale folosite la construirea unei case. Aceste elemente sunt modele usor de inteles create impreuna de catre beneficiar, utilizator si arhitect. Proiectarea conceptuala reprezinta traducerea cerintelor utilizatorilor si afacerii din limbajul utilizatorilor in limbajul dezvoltatorilor.

Proiectarea logica incepe sa intre in detaliile aplicatiei pe care echipa o va construi pentru a satisface cerintele afacerii si ale utilizatorilor. Astfel, proiectarea logica corespunde planurilor fundatiei si de evaluare ale unei case, in care sunt organizate elemente de tipul relatiilor spatiale. In aceasta faza a procesului echipa arhitectului va stabili relatiile si caile de comunicare intre elementele proiectului.

Proiectarea fizica are de-a face cu tehnologiile ce vor fi folosite. Acest proces de proiectare adauga detalii arhitecturii stabilite si va reflecta constrangerile din mediul in care va fi implementata aplicatia.

Activitatea de proiectare incepe adesea inainte ca faza de planificare sa inceapa in mod oficial si continua in anumite privinte pana cand nu se mai fac modificari in cod (catre sfarsitul fazei de dezvoltare). Termenele pentru fiecare din cele trei parti ale procesului de proiectare sunt stabilite in timpul fazei de planificare. Asa cum arata figura urmatoare, cele trei faze apar intr-o maniera decalata si paralela, primul output al unei activitati semnaland ca urmatoarea activitate ar trebui inceputa.

In figura 8 este prezentata interdependenta dintre activitatea de proiectare si cea de planificare a proiectului.

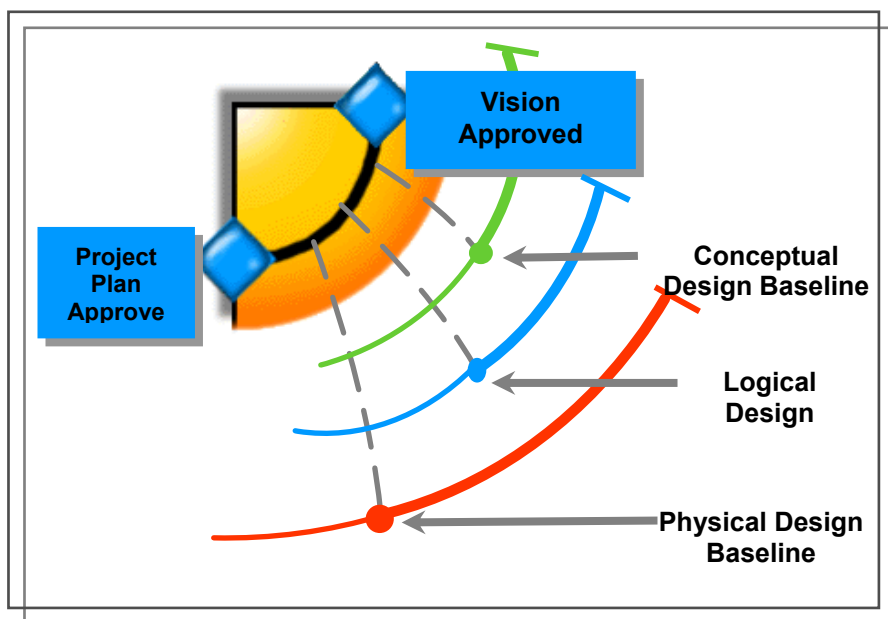


Figura 8 - Interdependenta dintre activitatea de proiectare si cea de planificare a proiectului.

2. Metoda ORACLE pentru managementul proiectelor software - ORACLE PJM

Metoda ORACLE PJM oferă un cadru consistent de planificare, estimare, control și terminare a proiectelor informatice de orice tip. Metoda pune accentul pe defnirea clară a cerințelor

clientului care să rămână vizibile de-a lungul întregului ciclu de viață al proiectului. Metoda ORACLE PJM formalizează mecanismele de control pentru a ajuta partajarea informațiilor critice despre proiect în cadrul echipei proiectului și coordonarea cu stakeholderii externi.

Metoda este aplicabilă la o gamă variată de proiecte. Deși inițial metoda a fost definită pentru proiectele de dimensiuni medii-mari, în prezent este aplicabilă la toate tipurile de proiecte, inclusiv la cele de mici dimensiuni. Metoda este dimensionată la nivelul activităților realizate în cadrul unor pachete de activități, subproiecte și chiar programe informatice.

Metoda ORACLE PJM este o metodologie orientată pe procese, care poate fi adaptată la nevoile specifice ale unui proiect informatic (figura 9).

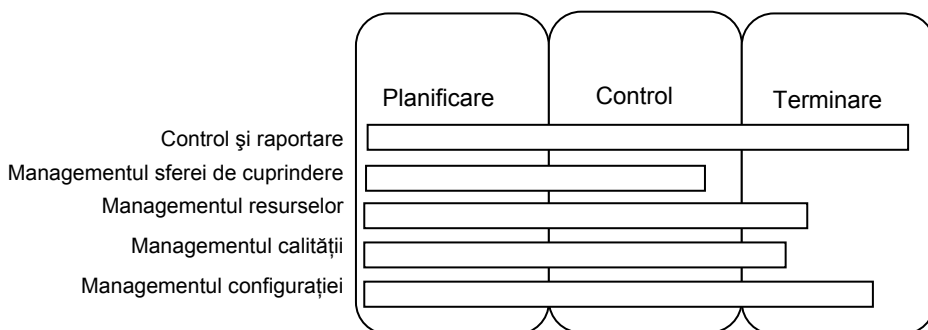


Figura 9 – Procesele de management în metoda ORACLE PJM

Principalele procese sunt:

- Control și raportare
- Managementul sferei de cuprindere
- Managementul resurselor
- Managementul calității
- Managementul configurației

Control și raportare. Asigură confirmarea sferei de cuprindere a proiectului, managementul schimbării și controlul riscului. Aceste procese exprimă modul în care se realizează elaborarea planurilor de proiect și a rapoartelor cu privire la starea proiectului.

Managementul sferei de cuprindere. Reprezintă task-urile prin care se definește, se monitorizează și se coordonează activitățile din cadrul proiectului. Aceste task-uri asigură perceperea din punct de vedere financiar a proiectului.

Managementul resurselor. Reprezintă task-urile prin care se asigură personalul necesar în cadrul proiectului și implementarea unei infrastructuri corespunzătoare.

Managementul calității. Permite implementarea măsurilor de calitate pentru a se asigura satisfacerea cerințelor clientului de-a lungul ciclului de viață al proiectului.

Managementul configurației. Conține task-urile care servesc la memorarea, organizarea, urmărirea și controlul tuturor elementelor produse și livrate în cadrul proiectului.

Impărțirea proiectului pe faze oferă un mai bun control asupra proiectului, reducând incertitudinea. La sfârșitul fiecărei faze se obțin o serie de livrabile care sunt analizate și recepționate / acceptate de către client. Ansamblul fazelor formează ciclul de viață al proiectului.

Taskurile de management al proiectului în cadrul metodei ORACLE PJM sunt organizate în cinci faze (figura 10):

- Planificarea proiectului
- Planificarea fazei
- Controlul fazei
- Terminarea fazei
- Terminarea proiectului

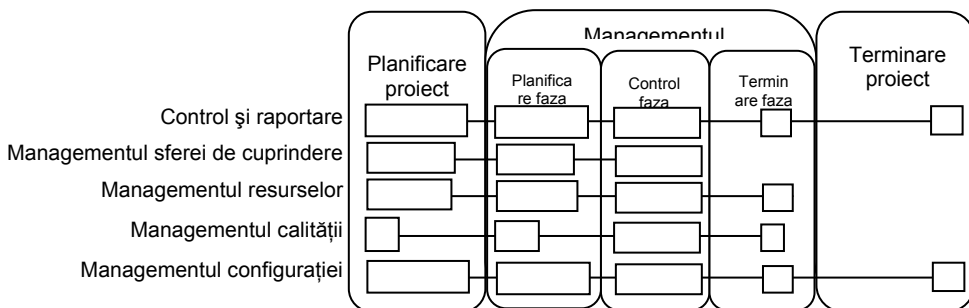


Figura 10 – Procesele de management și ciclul de viață al proiectului

Bibliografie

1. *** - Microsoft Solutions Framework, version 3.0, Overview, White paper, 2003.
2. *** - Oracle Method SM – Project Management, Method Handbook, 1999.