

5. Managementul calității

5. 1. Generalități

Calitate

= totalitatea caracteristicilor unei entități ce asigură abilitățile acesteia de a satisface nevoile **impuse** și **implicite** [PMBOK]

= totalitatea caracteristicilor ce asigură respectarea cerințelor **funcționale** și de **performanță**, a cerințelor **implicite**, așteptate + respectarea standardelor de dezvoltare/construcție corectă [Software]

= nivelul de excelență pentru ceva [Webster Dictionary]

≠

Categorie („grade”)

= rang asociat unor entități cu același rol funcțional, dar caracteristici tehnice diferite

ex: hârtie pentru fotografii, copiator, scris

Dificultăți:

- PM trebuie să asigure calitatea **activității de management** și obținerea unor **livrabile de calitate**
- pentru evaluarea nivelului de calitate nu există mereu **attribute & metrici** unanim acceptate

Calitatea managementului de proiect rezultă din

Calitatea rezultatelor, obținerea rezultatelor care **erau așteptate**

Gradul de mulțumire al echipei

Experiența organizațională acumulată prin proiect

>> Cheia: comunicare, implicare stakeholders

Calitatea unui produs software

Nu există baze teoretice clare, complete → există o doză de subiectivism: există acord în privința unor factori importanți, dar nu există definiții precise, complete pentru factorii de calitate

Varianta propusă de Mc Call:

Factori de calitate (high level)

1. operare produs
 - corectitudine (face corect ce veau?)
 - fiabilitate (face mereu bine?)
 - eficiență (pe hardware-ul existent lucrează cât de bine se poate?)
 - utilizabilitate (e ușor de folosit, răspunde nevoilor utilizatorului?)
 - integritate (e sigur?)
2. revizuire produs
 - mentenanță (pot să repar?)
 - testabilitate (pot să-l testez ușor?)
 - flexibilitate (pot să-l schimb ușor?)
3. tranziție produs
 - portabilitate (pot să-l folosesc pe alte calculatoare?)
 - reutilizare (pot reutiliza?)
 - interoperabilitate (se poate interfața cu alte sisteme?)

Criterii (low level): acces audit, acces control, acuratețe, comunicativitate, consistență, conciziune, expandabilitate, completitudine, toleranță la erori, eficiență în execuție, independență de hardware, modularitate, simplitate, trasabilitate, documentare completă, etc

Calitatea&categoria unui produs sunt legate de

Valoare produs + costuri pentru realizarea produsului

Valoarea de utilizare

Fiabilitate

Longevitate (în IT: longevitate redusă!!!)

Calitatea&categoria unui serviciu sunt legate de

Cost de implementare

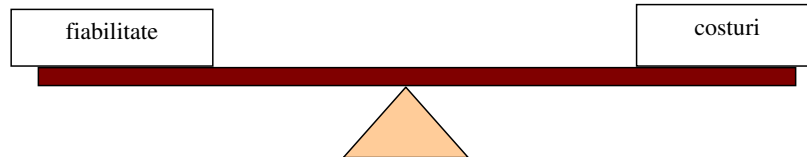
Valoarea obținută prin acel serviciu (tarif unitar * nr. unități accesate)

Gradul de satisfacție al utilizatorului

Fiabilitate

Longevitate

- trebuie realizat un compromis între cerințe divergente



Sugestii generale pentru managementul eficient al calității:

- **clientul** trebuie să fie mulțumit de produs/serviciu
 - >> proiectul trebuie să realizeze ceea ce s-a cerut
 - >> atenție la utilizabilitate
 - !!! calitatea scăzută e periculoasă, categoria scăzută nu trebuie obligatoriu evitată
 - !!! atenție la cerințele implicite
- **acțiunile preventive** sunt preferate celor de inspecție + corecție
- folosește **cicli de dezvoltare** care permit folosirea de modele conceptuale și verificarea începând cu primele faze
- **cadrul organizatoric** este foarte important: cultura + experiență + infrastructură

- **responsabilitatea revine în special PM**, echipa are doar rolul de participant!!

PM este responsabil de asigurarea nivelului de calitate + categorie dorite **pentru livrabile** și pentru **managementul de proiect**

>> uneori aceste obiective sunt divergente:

- ex: respectarea termenelor de predare prin reducerea perioadei de testare; respectarea cerințelor clientului prin creșterea volumului de muncă al echipei

>> **fii un bun exemplu!!!**

- **calitatea ridicată asigură succesul pe termen lung**

Standard de calitate

- dacă producătorul respectă un standard de calitate, cumpărătorul are **garanția** că produsul a fost realizat la un nivel acceptabil de calitate
- respectarea unui standard de calitate poate fi **greoaie și costisitoare**
Ex: ISO 9000 (ISO 9000-3 pentru software) pune accent pe documentare, pe formularea și folosirea consecventă a unor proceduri de lucru
 >> nu asigură direct calitatea dorită
 << revalidare la 3 ani, raportări/audit la 6 luni

5. 2. Procesele managementului calității

= procesele care asigură că proiectul realizează un produs/serviciu la nivelul de calitate dorit

planificarea calitate (PA)

identifică standardele de calitate potrivite și stabilește cum pot fi satisfăcute

asigurare calitate (E)

evaluează performanțele proiectului **în mod regulat** și indică dacă proiectul satisface standardele de calitate impuse

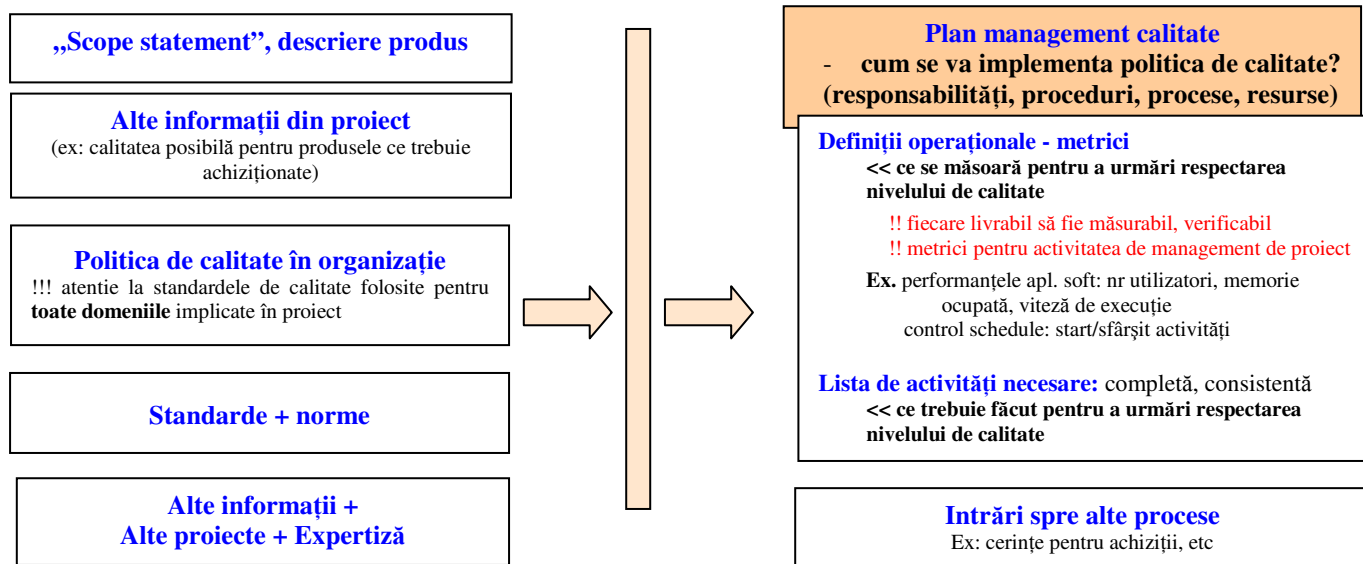
control calitate (C)

monitorizează **anumite rezultate** ale proiectului și determină dacă acestea respectă standardele de calitate +
identifică schimbările necesare pentru a reveni la standardele de calitate impuse

5. 2. 1. Planificarea calității (PA)

= identificarea standardelor de calitate relevante și a procedurilor necesare pentru ca standardele de calitate să fie îndeplinite

CUM OBȚII CALITATE?



Plan management calitate, listă activități pentru asigurarea calității

Listă livrabile majore

Cerințe: stabilite de clienți, interne, standard

Standarde + criterii de calitate pentru fiecare livrabil

Lista documente corelate: responsabili pentru realizare/verificare/execuție
(plan testare/integrare, specificații testare, raport testare)

Standuri/frameworkuri folosite pentru testare (mediu de testare)

Listă activități pentru asigurarea calității:

obiectiv, responsabil (execuție, verificare), estimare efort, metoda de verificare a calității folosită, borne de control, modalitatea de raportare a performanțelor (vezi listă recomandată la 5.3)

Definiții operaționale – metrici pentru verificarea asigurării calității, valori pentru stare normală (verde), alertă (galben), problemă (roșu)

Verificare și validare

Alte documente corelate

Terminologie, definiții, abrevieri

Recomandări:

- Este necesară o **analiza a costurilor** și o **analiză cost/beneficiu**

costurile necesare pentru a asigura calitatea cerută =

= costurile activităților care asigură că sunt respectate cerințele/calitatea
(prevenție + verificare)

+ costurile activităților de corecție, necesare în cazurile în care se
depistează că nu a fost respectat nivelul de calitate/categorie dorit

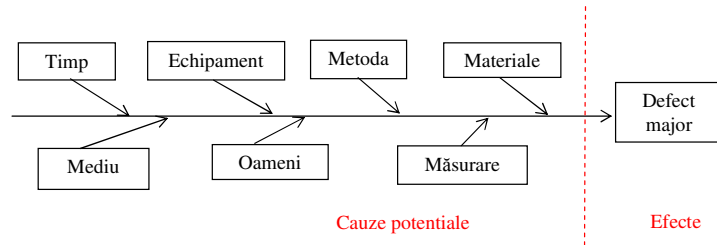
veniturile rezultă din gradul de satisfacție al beneficiarilor, productivitate, etc

!!!! să obții **venituri > costuri**

- **construiește diagrame care să ilustreze principalele cauze** ce pot conduce la nesatisfacerea nivelului de calitate dorit

Ex: **diagrame de flux („flowchart”);**

diagrame de tip cauză-efect Ishikawa („fishbone”) = diagrame care ilustrează cum anumite cauze pot genera anumite efecte:

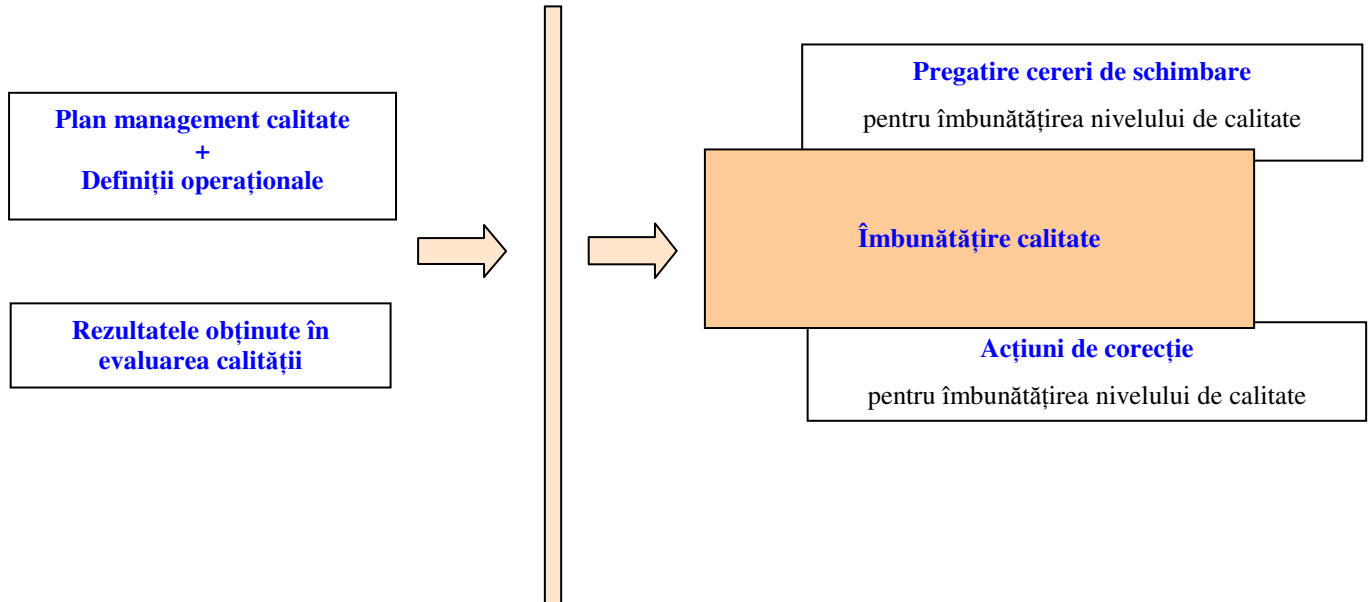


- stabilește dacă există **standuri de testare ("benchmark")** - pentru a evidenția mai bine calitatea produsului creat
- stabilește **setul de experimente** necesare pentru verificarea calității produsului (pot fi și seturi de date stochastice generate)
 - metoda poate fi folosită și pentru verificarea planului realizat în diverse situații ipotetice (riscuri care intervin, diverse combinații de strategii de management, etc)

5. 2. 2. Asigurare calitate (E)

= activitățile sistematice, planificate (să se desfășoare cu regularitate), care asigură că proiectul va respecta nivelul de calitate dorit

PROIECTUL RESPECTĂ NIVELUL DE CALITATE DORIT?



Recomandări + observații:

- organizația poate opta pentru **auditări suplimentare** realizate de către un auditor extern sau intern, la momente de timp prestabilite sau neplanificate
auditul ilustrează cum se desfășoară managementul calității pe proiect, ce lecții învățate pot fi utile pentru organizație
- în unele organizații există un **departament pentru asigurarea calității** care indică o parte din acțiunile necesare
- **acțiunile preventive** sunt preferate celor de corecție

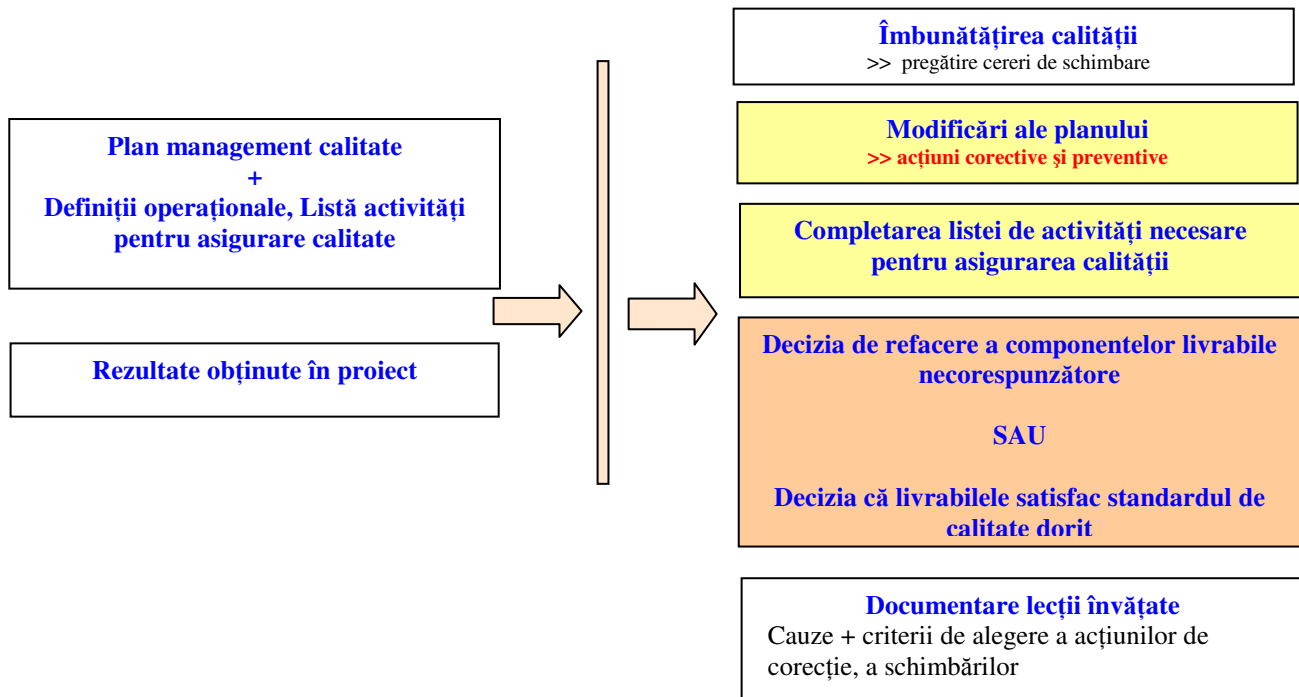
5. 2. 3. Control calitate (C)

- = monitorizează anumite rezultate ale proiectului pentru a determina dacă respectă standardele de calitate impuse
- + identifică variantele de corecție
- + gestionează schimbările necesare

CEEAA CE AI OBȚINUT RESPECTĂ NIVELUL DORIT DE CALITATE?

CE GREȘELI FACI?

CUM REPARI?



Recomandări:

- Sunt necesare evaluări atente ale rezultatelor:

Inspecție = măsurare + examinare + testare pentru a vedea dacă sunt îndeplinite cerințele

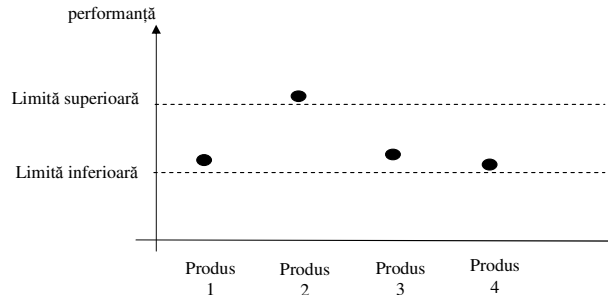
Inspecție	≠	Prevenție
(eroarea nu este văzută de client)		(eroarea este evitată în proiect)

Variante de inspecție:

- „**Peer review**” (= membrii echipei se verifică între ei), posibilă dacă există încredere și expertiză;
avantaje: membrii echipei comunică, învață unii de la alții, există o rezervă dacă un membru al echipei părăsește proiectul
- **Externă** - dacă nu există expertiză internă
avantaje: obiectivitate, acuratețe
- „**Walk around**” (= realizată de PM)
- **Statistică**

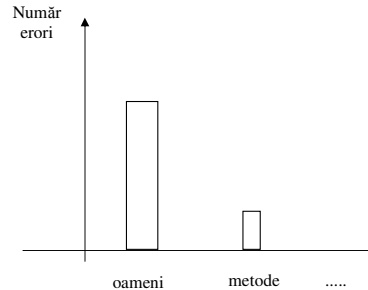
Dacă e posibil, **determină gradul de conformitate („variable sampling”)** nu stabili doar un **verdict acceptat/respins („attribute sampling”)**

La **acțiunile repetitive** construiește un grafic cu gradele de conformitate obținute pentru produsele similare realizate



Regula 7S: dacă 7 produse consecutive au calitate peste nivelul minim impus, dar apropiată de limita inferioară, atunci există o cauză asignabilă (sunt necesare măsuri de corecție)

- Orice modificare/corecție trebuie să se bazeze pe o analiză atentă:
 - **Reanalizează diagramele** de flux + diagramele cauză/efect
 - Realizează o diagrama care să ilustreze pentru fiecare tip de cauză posibilă numărul erorilor/defectelor generate în proiect



Legea Pareto: 80% din probleme sunt generate de 20% din elemente (lege 80/20)

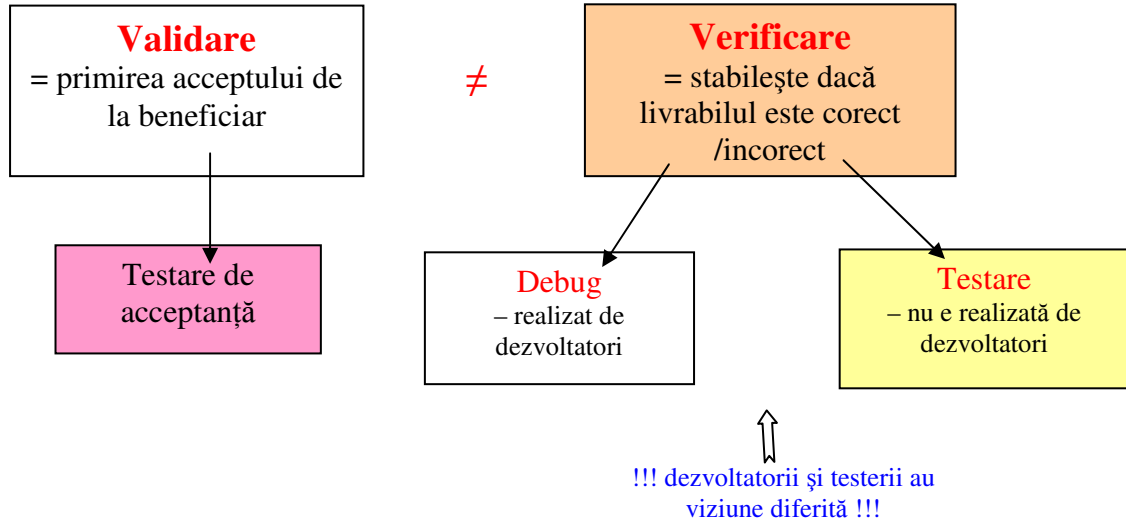
- **Analizează tendințele** manifestate pe proiect și predictează ce se va întâmpla în perioada următoare
 - analiza performanțelor tehnice: cunoscând erorile produse până atunci predictează numărul total al erorilor
 - analiza performanțelor managementului de proiectului (timp, cost, etc): cunoscând depășirile înregistrate până atunci predictează depășirile totale

- **Raportările să fie realizate corect și complet: schimbări/corecții, ce este bine realizat**

- + raportari cumulative: de la începutul proiectului până la momentul curent
- + raportări rezumative – pentru manageri
- + raportări de variație: ce variații au intervenit față de planul inițial

- Folosește **doar** procedurile convenite pentru a accepta schimbările
- **Notifică** toate schimbările către cei implicați
 - >> + integrarea schimbărilor

5. 3. Testarea produselor software



Observații:

- Rată costuri pentru corectare defecte, dacă eroarea a fost depistată la faze diferite ale proiectului **dezvoltare: testare: întreținere = 1: 10:100**
- Testarea solicită de regulă aproximativ **30 % din costul total al proiectului**
- Cauzele principale ale erorilor: **cerințe 60 %, design 30 %, cod 10 %**

Stil de testare:

- top - down (poate detecta repede erori majore, dar necesită stubs)
- bottom - up (se generează mai ușor cazurile de test)

- white-box – rar folosit la testare, folosit mai ales la debug
- black box – partiționarea spațiului de intrare în clustere, testarea la limitele intervalelor variabilelor de intrare, etc.

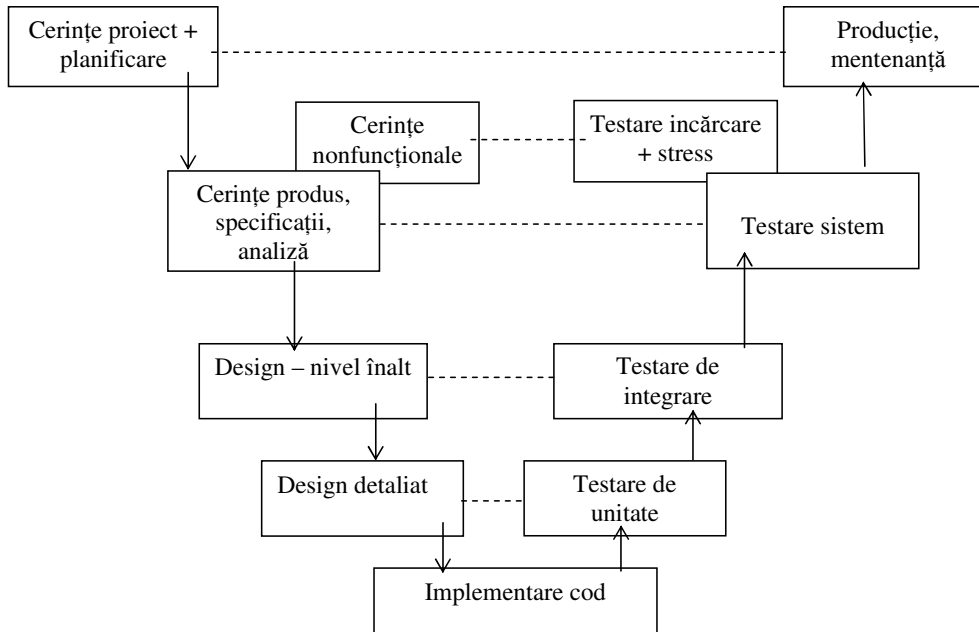
Ce se testează?

Testare

- **de unitate** – testare funcțională (black box)
- **de integrare** – testează interfațarea componentelor (black box)
- **de sistem** - testare de performanță run-time, stress și încărcare, securitate, mod de lucru în cazul unor erori („recovery”)
- **testare beta pentru validare** (pe baza criteriilor de acceptare ale beneficiarului)

!!!! Testarea automată este preferată: după orice corecție a codului, setul de teste trebuie reluat (testare regresivă)

Model V & V (verificare și validare)



Este software-ul suficient testat?

- Testarea nu poate garanta că nu există defecte, nu este exhaustivă!!!

În funcție de eficiență, testarea este

- „Toy test” = nesistematică, pe puține date
- **Testare profesională**
 - = cu selecția sistematică a cazurilor, set larg de date
 - >> grad bun de acoperire
 - + automatizare (scripturi + tools)

Indicatori pentru evaluarea eficienței testării

- **Grad de acoperire („test coverage”)** = % program care a fost testat (ramuri parcurse, instrucțiuni executate măcar o dată în cadrul testelor) [$<100\%$]
 - calculat automat de utilitarul folosit pentru testare
- **Eficiența unei secvențe de teste („fault seed”)**

Pas 1: se introduc NS greșeli în program

Pas 2: se rulează secvența de testare și se detectează NF erori, din care NFS introduse la pasul 1

Pas 3:

NF-NFS	?
NFS	NS

 $\ggg q = \frac{NF - NFS}{NFS} \cdot NS$ - să fie cât mai mic

Indicații pentru elaborarea planului – activități de testare

Activitățile specifice testării

Stabilirea obiectivelor pentru faza de testare (în faza de cerințe)

Design cazuri de testare

Elaborare scripturi de testare

Testare scripturi

Execuție scripturi + înregistrare erori

Evaluare rezultate

Corectarea erorilor – de către dezvoltatori

Reluarea testelor pe codul corectat

Structură recomandată pentru documentele „Specificații de testare”

Identificator proiect, PM

Nume, funcție pentru cel care propune secvența de test

Obiectivul testării, modul/componenta testată

Mediu de testare necesar - descriere

Descrierea detaliată a cazurilor de test manuale și automate

La fiecare caz de test se indică

Identificator,

Prioritate

Precondiții: dacă urmează după alte teste, starea inițială

Secvența de pași exactă precizând valori parametri, etc,

Descrierea seriei de teste – eventual cu o diagramă de secvențe

Ce poate fi detectat: la fiecare caz de test indică cerințele/caracteristicile testate (tabel separat)

Echipe: design teste, testare, elaborare și verificare raport testare

Istoricul schimbărilor

Alte documente corelate – specificații, etc

Terminologie, abrevieri, definiții

Structura recomandată pentru documentul „Plan de acceptanță”

Lista livrabile majore

Teste de acceptanță pentru fiecare livrabil inclus in lista

Indicatori de performanță conveniți pentru acceptanță

Termene de livrare/orar de livrare are rezultatelor

Structură recomandată pentru documentul „Raport de testare”

Identificator proiect, PM

Nume, funcție pentru cel care a coordonat testarea

Tip testare, modul/componenta testată - identificator secvența

Mediu de testare utilizat

Cazurile de test efectuate și rezultate – rezumat

Nr. total teste efectuate/prevăzute

Nr. total teste eșuate/reușite

Procentaj îndeplinire test

Probleme severe nerezolvate: descriere, prioritatea, etc

Cazurile de test efectuate și rezultate – detaliat

Observații: ex - timp de execuție test mai mare decât se aștepta

Echipa de testare, verificare raport

Timp necesar pentru testare

Alte documente corelate – specificații testare, etc

Terminologie, abrevieri, definiții

Recapitulare

Definiții, terminologie:

calitate/categorie, standard de calitate

testare (categorii), verificare/validare, model V&V

diagrame de flux/Ishikawa, regula 7S, 80/20, etc

Procese management calitate: planificarea calitate (PA), asigurare calitate (E), control calitate (C)

Documente

Plan management calitate, listă activități pentru asigurarea calității, indicatori

Specificații de testare

Plan de acceptanță

Raport de testare