

The
Information
Management
Company

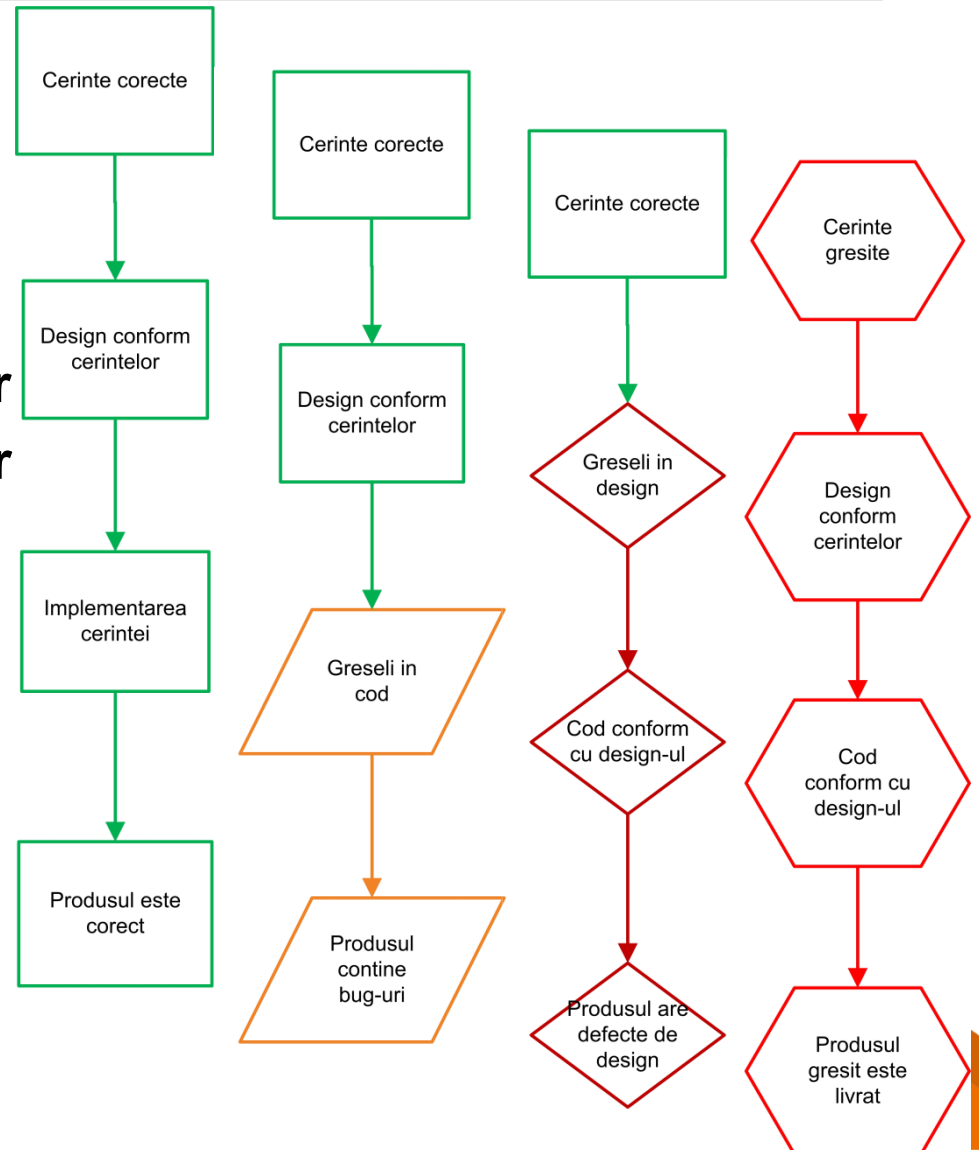


Open4Tech – Concepte de QA



De ce e necesara testarea?

- Este necesara pentru ca:
- Asa se gasesc defectele;
- Dovedeste calitatea softului;
- Previne aparitia defectelor
- Reduce riscul incidentelor operationale;
- A gresi este uman;
- Se creste increderea in produs;



Ce este calitatea?

Calitatea → Gradul in care o componenta, sistem sau proces indeplineste cerintele specificate si/sau nevoile si asteptarile utilizatorului sau clientului;

Calitatea Software → Totalitatea caracteristicilor si functionalitatilor unui produs software care satisfac nevoile implicite sau definite [ISO 9126];

“Quality is value to some person who matters” - Jerry Weinberg;

Ce este calitatea?



**Testarea nu inseamna numai a verifica (lucrul este facut bine)
ci si a valida (lucrul corect este facut)!**

Care sunt cauzele defectelor softului?

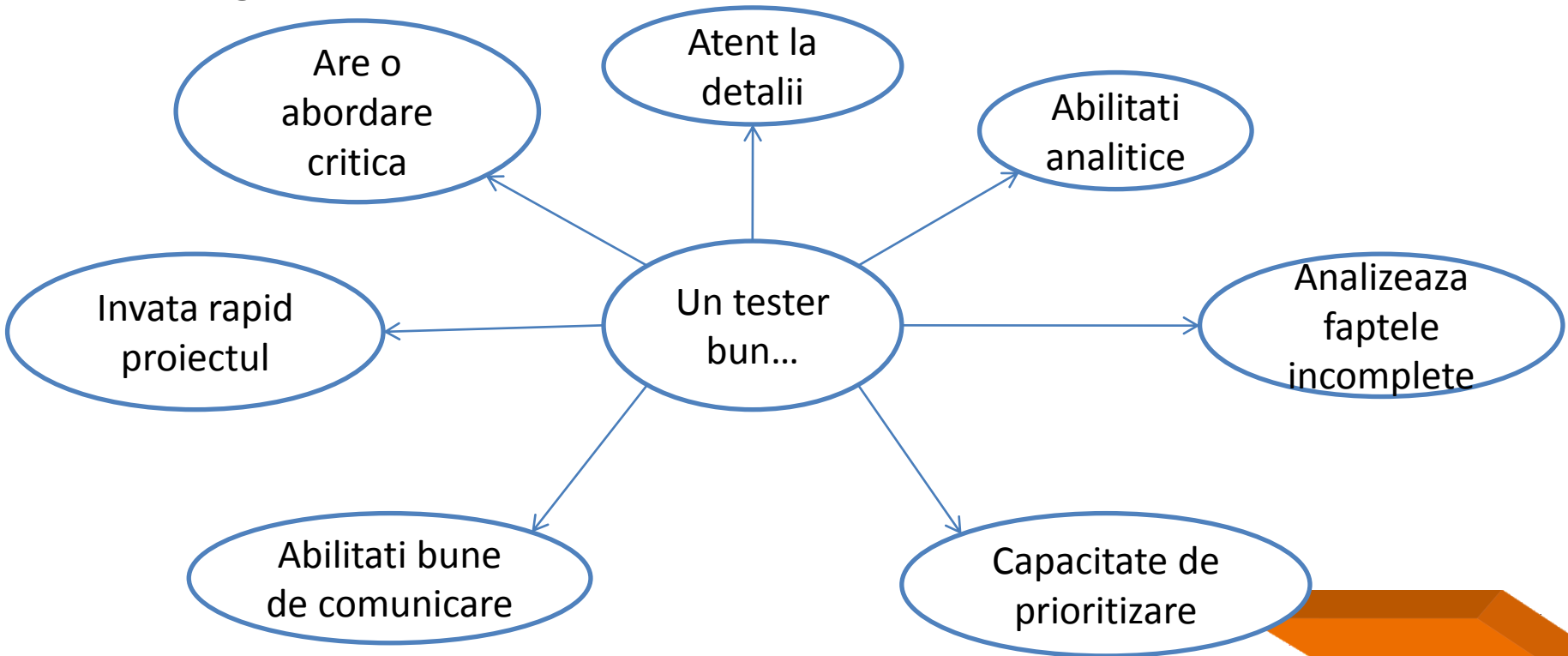
- **Cerinte (corecte, logice, complete, verificabile, documentate)**
- **Caracteristicile si functionalitatile (corectitudine, cazuri lipsa, domeniu si limite, mesaje, exceptii nedescoperite)**
- **Structura aplicatiei (fluxul de lucru, secventialitatea, procesarea datelor)**
- **Documentele si attributele acestora (definirea, structura, accesarea, manipularea)**
- **Implementarea si codul**
- **Integrarea (interfatari interne si externe)**
- **Sistemul (arhitectura, performanta, recuperarea datelor)**

Care sunt cauzele defectelor softului?

- **Definirea testelor si executia acestora (design-ul, executia, documentarea si raportarea testelor);**
- **Interfata (functionalitate, comunicare, elemente lipsa, performanta);**
- **Manipularea erorilor (prevenirea, detectia, recuperarea);**
- **Limitari (numerice, bucle);**
- **Initializarea (data item, controlul buclelor);**
- **Controlul fluxului de lucru (stop, crash, bucle, if-then-else,...);**
- **Manipularea datelor (tipuri de date, lista de parametrii, valori);**
- **Sursele si controlul vesiunilor (apar bug-uri mai vechi)**
- **Testarea (nu s-a observat, nu s-a testat, nu s-a raportat)**

Psihologia testarii

- **Testarea este uneori privita ca o activitate distructiva deoarece rulam teste pentru a face software-ul sa “crape”, dar in realitate testarea este o activitate foarte constructiva avand multe beneficii in gestionarea riscurilor de produs.**



Psihologia testarii

- **“Cel mai bun tester nu este cel care gaseste cele mai multe bug-uri, cel mai bun tester este cel care primeste cele mai multe bug-uri fixate”**
- **Gradul de independenta este direct corelat cu eficienta in gasirea defectelor si esecurilor.**
- **Nivele de independenta pot fi definite astfel:**
 - Nivel scazut de independenta – dezvoltatorii scriu si executa propriile lor teste;
 - Nivel mediu de independenta – testele sunt scrise si executate de un alt dezvoltator;
 - Nivel inalt de independenta – testele sunt scrise si executate de o echipa de testare independenta (din interiorul sau din afara companiei);

Principiile testarii

- **Testarea arata prezenta defectelor, dar nu poate demonstra ca nu exista defecte**
- **Testarea exhaustiva este imposibila**
- **Testarea timpurie (Early testing)**
- **Gruparea defectelor (Defect clustering)**

Principiile testarii

- **Paradoxul pesticidului (pesticide paradox) –**
 - In cazul in care aceleasi teste sunt repetate iar si iar, nu vor mai fi gasite noi defecte.
- **Testarea este dependenta de context**
- **Neadevarul sau iluzia absentei defectelor (Absence-of-errors fallacy)**
 - gasirea si fixarea defectelor nu ajuta in cazul in care sistemul construit este inutilizabil si nu indeplineste asteptarile si nevoile utilizatorilor.

Fundamentele procesului de testare

- **Procesul de testare consta in urmatoarele activitati principale:**
 - Planificarea si controlul testarii
 - Proiectarea si analiza testarii
 - Executia si implementarea testarii
 - Evaluarea criteriilor de iesire si raportarea
 - Activitatile de inchidere a testarii

Nivelele testarii

- **Testarea pe componente/ component testing**
 - defectele sunt cautate in componenta testata;
 - se face verificand si codul si avand in permanenta suportul unui dezvoltator;
 - nu se inregistreaza incidentele;
 - o variatie este “Unit testing”;
- **Testarea integrarii componentelor/ integration testing**
 - se testeaza interdependentele dintre componente, interactiunile dintre diferite parti ale sistemului;
 - poate sa includa si testare non-functionala;
- **Testarea sistemului/ system testing**
 - comportamentul intregului sistem/produs definit ca scop al proiectului sau programului dezvoltat;
 - mediul de test trebuie sa corespunda cu mediul de productie;
 - se vor testa atat cerintele functionale, cat si cele non-functionale;

Nivelele testarii

- **Testarea de acceptanta/ acceptance testing**
 - realizata adesea de consultanti sau de clienti, insa pot fi implicate si alte parti;
 - are scopul de a stabili increderea in sistem si de a raspunde la intrebarea “Este produsul destul de bun incat sa fie livrat catre client?”;
 - nu se pune accent pe gasirea de defecte, insa in testarea de acceptanta se poate evalua instalarea si utilizarea sistemului;

Tipuri de testare

Testare

- Functionala
- Non-functionala
- Structurala
- Confirmare si Regresie
- Mentenanta

Tehnici de testare

- **Tehnici black-box**
 - a) Equivalence partitioning
 - b) Boundary value analysis
 - c) Testarea folosind tabele de decizie (decision table testing)
 - d) Tranzitii de stare (State transition testing)

Cerinte: Numar – natural – cuprins intre 1 si 50



Care din urmatoarele seturi de valori de intrare este corect in cazul analizei *boundary value*?

- a) 0, 1, 2, 50, 51
- b) 1, 2, 49, 50
- c) -1, 0, 1, 2, 49, 50, 51
- d) -32767, -1, 0, 1, 49, 50, 51, +32768

EP: Pentru exemplul dat, care din seturile de date de mai jos acopera fiecare clasa de echivalenta a unei partitii de echivalenta?

- a) 0, 10, 50;
- b) 1, 2, 40;
- c) 0, 1, 50;
- d) -1, 1, 99

Instrumente ce ajuta in activitatea de testare

- **Tool-uri pentru managementul procesului de testare**
 - Tool-uri de management al cerintelor
 - Tool-uri de management al incidentelor (issue tracker tools)
 - Tool-uri de management al configurarii (CM)
- **Tool-uri pentru proiectarea testelor si pregatire a datelor**
- **Tool-uri folosite pentru executia testelor si logare**
 - Tool-uri de executare a testelor
 - Test harness/unit test framework tools
 - Comparatoare de test
 - Tooluri pentru masurarea acoperirii testarii
 - Tool-uri de securitate
- **Tool-uri pentru testarea performantei si pentru monitorizare**
 - Tool-uri de analiza dinamica
 - Tool-uri de testare a performantei/ de testare a incarcarii/de testare de stres
 - Tool-uri de monitorizare

Beneficii si riscuri ale utilizarii instrumentelor de testare

➤ **Beneficii:**

- Munca repetitiva este redusa (teste de regresie, implementarea de standarde, conventii, etc...);
- O mai mare coerenta si repetabilitate;
- Evaluarea obiectivului;
- Acces usor la informatiile despre teste sau testare.

➤ **Riscuri:**

- Asteptari nerealiste despre tool;
- Subestimarea timpului, costului si efortului necesar introducerii initiale a tool-ului;
- Subestimarea timpului si efortului necesar de a obtine beneficii semnificative si continue de pe urma tool-ului;
- Subestimarea efortului necesar pentru mentenanta produselor (teste, loguri, etc.) generate de tool;
- Increderea nerealista acordata toolului (folosirea lui pentru situatii in care nu este eficient si unde testarea manuala ar fi mai indicata).

Elemente esentiale unui soft de calitate

➤ **Funcționalitate**

- Adecvata;
- Precizie;
- Interoperabilitate;
- Respectarea cerintelor;
- Securitate;
- Trasabilitate;

➤ **Uzabilitate**

- Inteligibila si invatabila;
- Operabila;
- Explicita;
- Customizabila;
- Atractiva si clara;
- Ajutatoare;
- Prietenoasa;

➤ **Robustete**

- Maturitate;
- Tratarea erorilor;
- Recuperabilitatea datelor;
- Disponibilitate;

➤ **Eficienta**

- Dpdv al duratei unei actiuni;
- Dpdv al comportamentului resurselor;
- Scalabilitate;

➤ **Portabilitate**

- Adaptabilitate;
- Instalare usoara;

➤ **Mentenabilitate**

Q&A pentru QA



Multumesc