

# Auditul Informatic

Ion Ivan  
Alecuc Felician  
Sergiu Capisizu

## 1. Orientarea spre calitate și spre garantarea calității

Societatea informațională a atins acel stadiu căruia îi corespunde orientarea spre cetățean a aplicațiilor și spre diversitatea situațiilor în care acesta se află la un moment dat.

În acest context, aplicațiile informatice trebuie să atingă un nivel calitativ superior. Cetățeanul, principalul beneficiar al efectelor directe pe care le generează societatea informațională, trebuie să obțină un nivel maxim a gradului de satisfacție.

Pentru a dezvolta o aplicație, inclusiv o aplicație web, se parcurge un ciclu format din etape, definite riguros ca succesiune și sarcini.

Calitatea aplicațiilor informatice se planifică. În funcție de experiența acumulată, de exigențele care apar pentru perioade bine determinate, se enumera caracteristicile de calitate iar prin similitudine cu aplicații aflate în exploatare se specifică nivelul de calitate cu care va fi înzestrat produsul informatic sau baza de date, rezultatele interfețelor și fluxurile de selecție ale site-urilor.

Pentru caracteristicile de calitate  $C_1, C_2, \dots, C_m$  se stabilesc nivelurile planificate  $pl_1, pl_2, \dots, pl_m$ . Pe parcursul analizării unui produs informatic, a unei baze de date sau a unei interfețe, se măsoară nivelurile efective pentru caracteristicile de calitate realizate:  $pr_1, pr_2, \dots, pr_m$ .

Prin comparare, rezultă deciziile care trebuie luate:

1. în cazul în care nivelurile planificate sunt identice sau sunt mai mici decât nivelurile efective,  $pl_i \leq pr_i$ , rezultă că procesul de dezvoltare decurge conform cerințelor;
2. în cazul în care  $pl_i \geq pr_i$  înseamnă că trebuie adoptate măsuri pentru creșterea nivelului caracteristicii  $C_i$  a produsului informatic, a bazei de date sau a interfeței web.

În final, pentru certificarea produsului, trebuie ca nivelurile planificate ale caracteristicilor de calitate să fie mai mici sau egale decât nivelurile reale. În activitatea de informatică aplicată trebuie parcurse, pe lângă etapele ciclului de dezvoltare, o serie de activități care au menirea de a stabili că produsul program, baza de date, interfața web sau oricare alte componente, reprezintă exact ceea ce trebuie. Aceste activități au menirea de a stabili că există concordanță între ceea ce s-a planificat și produsul finit existent. Faptul că activitățile sunt derulate de persoane aparținând unor organisme independente, reprezintă garanția că rapoartele consemnează diferențele reale dintre proiect și produs.

Auditarea reprezintă o activitate deosebit de importantă pentru companiile de software întrucât rezultatele procesului de auditare se constituie în bază pentru fundamentarea deciziilor pe termen lung privind politicile de management al calității.

Certificarea echipelor, sistemelor de management al calității, companiilor, persoanelor, produselor informatice, evidențiază capacitatea de a derula procese, tranzacții, capacități care trebuie să se manifeste ori de câte ori se dezvoltă un produs, prin respectarea standardelor, prin utilizarea tehnicilor, modelelor și instrumentelor adecvate. Toate elementele converg spre obținerea de produse și servicii de calitate.

Există o importanță deosebită între ceea ce se dorește și ceea ce se efectuează, se obține și se utilizează în activitatea economică de zi cu zi. Există două probleme și anume:

1. crearea climatului, a educației și a mentalității orientate spre calitatea proceselor, produselor, intrărilor și ieșirilor acestora;
2. stabilirea concordanței dintre imaginea proiectată a produsului sau serviciului și caracteristicile produsului sau serviciului real.

Garantarea calității este un concept de mare importanță, complexitate și dinamică. Existența unei mărci este fundamentată pe existența unei conduite, pe definirea de proceduri restrictive, calificarea continuă a persoanelor pe existența unui management suficient al calității și pe formarea conștiinței orientate spre calitate.

Indiferent de contextul în care o companie de software își desfășoară activitatea, societatea informațională, în stadiul arătat, impune ca dezvoltarea de sisteme informatice prin adăugarea de componente să genereze un efect de antrenare multiplă în direcția pozitivă a evoluției gradului de satisfacție înregistrat de utilizatorul final, cetățeanul.

## **2. Concepte de bază**

Auditul informatic reprezintă un domeniu cuprinzător în care sunt incluse toate activitățile de auditare pentru specificații, proiecte, software, baze de date cât și procesele specifice ciclului de viață asociate unui program, unei aplicații informatice, unui sistem informatic pentru management sau unui portal de maximă complexitate aparținând unei organizații virtuale.

Auditul reprezintă o formă esențială a serviciilor independente prin care se verifică faptul că o aplicație informatică, prin prelucrările pe care le efectuează, își atinge obiectivul pentru care a fost elaborată.

Auditorul din domeniul informatic este acel specialist cu solide cunoștințe de informatică, atestate prin certificate și diplome, care stăpânește cadrul legislativ conexe cu informatica și care are certificări recunoscute în domeniul auditului informatic. Prin înscrisurile pe care le posedă, auditorul informatic dovedește că în fața unor comisii recunoscute a arătat că a studiat documentația specifică, o poate reda, poate face conexiunile necesare pentru a soluționa probleme, a elabora un raport și pentru a parcurge etapele ciclului de audit informatic.

Nu înseamnă, însă, că, automat, auditorul informatic certificat, atunci când trece la auditarea unei aplicații informatice, prestează un serviciu de excelență calitate. Aceste certificări evidențiază faptul că:

- auditorul informatic a parcurs o literatură de specialitate fundamentală minimală și o stăpânește pe o scară de la 1 la 10 obținând puncte de la 6 la 10;
- auditorul informatic a fost supus unor teste de cunoștințe și a dovedit că și-a însușit, dar mai ales că utilizează, atunci când este nevoie, aceste cunoștințe pentru a evidenția fie performanța unui produs informatic, fie carențele produsului;
- auditorul informatic a elaborat un număr de proiecte practice, având prilejul de a valorifica la fața locului, pe un produs software, pe o bază de date sau pe o componentă intermediară, cunoștințele teoretice, să conchidă că gradul de concordanță între specificație și produsul efectiv are un nivel pe care l-a determinat, iar valoarea acestui indicator rezultă prin aplicarea corectă a unei formule;
- auditorul informatic respectă procedurile specifice fiecărei etape a ciclului de auditare. În acest fel asigură derularea procesului de auditare menținând caracterul de imparțialitate pe care însăși meseria de auditor informatic, prin titulatura de independență, o presupune.

Organismele de certificarea auditorilor din informatică sunt numeroase. Este important să fie certificate acele organizații caracterizate prin notorietate, prin rigurozitatea procesului de examinare și mai ales prin standardele și manualele de auditare utilizate. Organizațiile de certificare a auditorilor au o activitate transparentă definită prin:

1. prezentarea clară, fără echivoc, a condițiilor pe care trebuie să le îndeplinească persoanele pentru a participa la o competiție de certificare în vederea obținerii calității de auditor informatic;
2. cursurile pe care le organizează, specificând duratele, conținutul, materialele didactice, forma de finalizare, perioadele, costurile și mai ales drepturile pe care le oferă, în ideea de stabilire a nivelului de certificare la care are acces o persoană după finalizare;
3. costurile de certificare, forma de examen, nivelul de exigență și mai ales recunoașterile diplomelor;
4. enumerarea marilor companii de software, produse software și aplicații complexe care au fost abilitate de specialiști ce au obținut certificarea organizației;
5. organizațiile internaționale de certificare în care organizația este membru și care recunosc respectiva organizație de certificare a auditorilor;
6. procedurile de certificare specifice probelor practice, proiectelor și a celorlalte documentații aferente procesului.

Standardele de auditare existente trebuie să stea la baza formării auditorului informatic. Standardele definesc clar domeniul, activitățile, etapele, conținutul auditării și formele de finalizare. Respectând cerințele standardelor, rezultatul procesului de auditare informatică este eliberat de riscurile contestării. În domeniul informatic există mai multe direcții de dezvoltare a auditului cum ar fi COSO, ITIL, COBIT.

Auditarea software constă în activități prin care se evidențiază gradul de concordanță dintre specificații și programul elaborat. Auditul software măsoară siguranța pe care trebuie să o aibă utilizatorul de programe scrise în limbajul C++ , Pascal, Java atunci când obține rezultate. Siguranța se referă la corectitudinea și completitudinea rezultatelor finale atunci când datele de intrare sunt, de asemenea, corecte și complete.

Auditul bazelor de date este un domeniu de maximă complexitate având în vedere faptul că, de regulă, lucrul cu bazele de date presupune atât datele ca atare însoțite de relațiile create între ele, cât și programele cu care datele se gestionează.

Auditul datelor vizează definirea acelor elemente prin care se stabilește măsura în care datele stocate îndeplinesc cerințele de calitate: corectitudine, completitudine, omogenitate, comprehensibilitate, temporalitate, reproductibilitate. Pentru fiecare caracteristică există o metrică elaborată, iar auditorul de date trebuie să evalueze nivelul atins de caracteristică, pentru setul de date supus auditării. În final, auditorul de date certifică faptul că datele stocate în baze de date constituie intrări valabile pentru a obține rezultate corecte.

Auditorul riscului de gestiune a datelor stocate în baze de date, prin procedurile pe care le aplică, verifică dacă:

1. programele referă corect câmpurile cu date stocate;
2. operațiile de prelucrare sunt cele din specificații;
3. agregările, sortările, evaluările de expresii de extragere a subseturilor de date sunt în concordanță cu specificațiile de obținere a rezultatelor ca structură, dimensiune și conținut.

Dincolo de totalitatea procedurilor, standardelor, fluxurilor și rapoartelor pe care trebuie să le parcurgă specialiștii cu certificare în acest domeniu, auditul organizației presupune și stabilirea gradului de siguranță pe care îl oferă respectiva organizație ca o

măsură a faptului că, odată ce a primit o comandă pentru a proiecta un produs software, o aplicație informatică sau un sistem informatic, sunt lansate, pe măsură ce se desfășoară etapele specifice, componente corespunzătoare calitativ și funcțional. Prin asamblarea acestora, în final, se obține produsul software aplicativ sau sistemul informatic care răspunde obiectivului definit.

Auditarea se caracterizează prin faptul că se bazează pe un număr de principii, cum ar fi: comportamentul etic (încrederea, integritatea, confidentialitatea și discreția sunt esențiale pentru auditare), prezentarea corectă a rezultatelor (care trebuie să reflecte cu fidelitate și cu acuratețe activitățile de audit), responsabilitate profesională. Metodele de colectare a informațiilor necesare pentru audit includ interviul, observarea activităților și analiza documentelor, a înregistrărilor și a altor surse de informare. Auditorii nu sunt nici prieteni, nici dușmani ci profesioniști care trebuie să identifice diferențele față de procedurile implementate sau procesele operationale

În cadrul unui audit, comunicarea este vitală și elemente ale comunicării, cum ar fi cuvintele (7%), timbrul vocii (38%) dar mai ales mimica, ținuta, gesturile, contactul cu privirea (55%) sunt foarte importante. De asemenea, ascultarea activă este o tehnică pe care ar trebui să o învețe și să o aplice în mod constant toți auditorii, aceasta fiind cheia comunicării eficiente.

### **3. Codul auditului informatic**

Analistul informatic are la dispoziție numeroase tehnici și metode pe care le adaptează contextului. Într-un fel este auditat un program de calcul statistic sau de optimizare și altfel este auditată o aplicație care utilizează o bază de date. Pentru un sistem informatic complex există metode adecvate de auditare, iar pentru aplicațiile web accentul auditării cade pe gradul de satisfacție a grupului țintă. Aplicațiile mobile au fundamente de auditare în care accentul este pus pe asigurarea continuității, compatibilității, accesibilității rapide la resurse și mai ales asupra nivelului atins de asigurare a securității fluxurilor din întregul sistem.

De aceea, în cadrul procesului de audit informatic, *planificarea și definirea metodei de audit* este esențială. Alegerea unei metode nepotrivite conduce la utilizarea de instrumente neadecvate, iar rezultatele auditului au caracter speculativ.

Alegerea metodei presupune obținerea unor informații privind contextul în care se derulează procesele legate de obiectul auditării cum ar fi produsul software, aplicația informatică sau sistemul informatic.

Auditul este, prin complexitatea sa, o activitate în care sunt luate în analiză legăturile și implicațiile pe care le generează produsul software, aplicația informatică sau sistemul informatic între dezvoltator (compania de software) și utilizator. Raporturile trebuie privite din punct de vedere tehnic, financiar și juridic. Aspectul tehnic privește date de interior, algoritmi, rezultate, resurse folosite. Aspectul financiar vizează costul estimat al produsului software, aplicației, sistemului informatic și costul efectiv, modul în care s-au efectuat plățile. Caracterul juridic al abordării vizează obligațiile contractuale și legislația din domeniul informatic.

Toate aceste elemente conduc la stabilirea unor proceduri preliminare prin care sunt definite direcțiile de analiză, gradul de semnificație pe care procesul de audit informatic îl oferă și riscurile ca unele concluzii să fie infirmate de practica derulării proceselor de utilizare curentă a produsului software, a aplicației informatice sau a sistemului informatic în integralitatea lui.

Pentru a realiza auditul informatic se definește planul de audit general și programul de audit. Structura planului și definirea programului sunt standard, presupunând parcurgerea unor pași obligatorii. Specificitatea produsului software, a aplicației informatice sau a

sistemului informatic și complexitatea acestora determină efectuarea unor detalieri care diferă de la un plan general la altul, respectiv de la un program de audit informatic la altul. Sarcinile care se includ în plan, eșalonarea etapelor din program au elemente de variabilitate legate strict de structura și de diversitatea produselor informatice analizate.

Standardele de auditare includ suficiente elemente astfel încât planul general și programul de audit să fie riguroase, neambigue și mai ales operaționale. Înainte de a se trece la auditul informatic propriu-zis, datorită eforturilor ridicate de derulare și mai ales datorită riscurilor ca reproductibilitatea procesului de audit să fie afectată chiar de schimbările care au loc în produsele informatice auditate, trebuie efectuate teste asupra mecanismelor de control și a mecanismelor de testare pe codul sursă, pe specificații, pe diagrame, pe documentații, pe structuri de rezultate. Pentru a obține o reducere a nivelului estimat pentru riscul erorilor de analiză și control asociate produsului informatic, se procedează la efectuarea, în manieră iterativă, a unor corecții asupra procedurilor tehnice, metodelor și modelelor de analiză și control asociate procesului de audit. Procesul iterativ se întrerupe atunci când estimarea probabilității ca rezultatele auditării informatice să fie afectate de erori a atins un prag acceptabil.

Auditul propriu-zis include proceduri analitice prin care se evidențiază diferențele dintre ceea ce s-a planificat a se realiza și ceea ce s-a realizat.

Procedurile analitice au la bază contractele încheiate între părți, minutele care detaliază obiective, sarcinile ce revin partenerilor, specificațiile. Auditorul tehnic trebuie să ierarhizeze informațiile astfel încât să identifice punctele cheie care definesc procesul de analiză, proiectare, dezvoltare, testare, implementare a produsului informatic, fie că este vorba de un simplu program, fie că este vorba de o aplicație informatică desktop sau în rețea, fie că este vorba de un sistem informatic care vizează întreaga activitate a unei organizații.

Toate procedurile analitice și textele de detaliere aplicate modulelor, programelor și sistemelor de programe au menirea de a evidenția pas cu pas comportamentul secvențelor de program. În cazul în care auditorul informatic are la bază pregătire de programator, acesta știe să aleagă din multitudinea de proceduri și texte cu caracter analitic pe acelea care oferă informația reprezentativă privind produsul software auditat, fie că este vorba de un modul sau de un sistem complex. Efortul de auditare este ridicat, indiferent de complexitatea produsului auditat.

Auditul se încheie cu un raport care are la bază o serie de verificări ale intercondiționărilor dintre module, dintre programe, respectiv dintre subsistemele sistemului informatic, pentru momentul  $t$ , considerat ca bază.

Se verifică modul de producere a evenimentelor care sunt concretizate prin succesiuni de prelucrări, corespunzătoare momentului  $t + 1$ . În acest fel, produsul informatic, proiectat pentru derularea unor seturi de prelucrări, este analizat ținând cont tocmai de succesiunea prelucrărilor.

Auditul informatic are la bază înregistrări privind structura software, structura bazei de date, înregistrări ale lungimilor, volumului, complexității și înregistrări complete asupra comportamentului în timpul execuției.

În cazul în care există seturi de date cu care au fost testate produse informatice din aceeași clasă cu produsul auditat acum, se colectează serii de date privind comportamentul produsului pentru a fi comparat cu produsele deja existente. Când nu există date, acestea sunt generate și testarea produsului se realizează simultan cu produse din aceeași clasă, tocmai pentru a se efectua analize și pentru a compara produsele informatice. Seriile de date se constituie în baze de redactare a raportului de auditare.

Raportul de auditare este o lucrare de sinteză care are la bază o analiză a rezultatelor obținute din parcurgerea textelor sursă, din lansarea în execuție a programelor și din

interpretarea rezultatelor finale, mai ales prin interpretarea rezultatelor intermediare și a celor de tip depanare.

Raportul de audit este o lucrare cuprinzătoare care oferă o imagine privind siguranța pe care o oferă produsul. Este necesar ca raportul de audit să joace un rol activ în dezvoltarea ulterioară a companiei de software iar de cele mai multe ori acesta conține și propuneri de îmbunătățire.

De cele mai multe ori, auditul informatic este cerut ca soluție finală, imparțială, pentru a justifica ipotezele unei părți contractante, fie că este vorba de cumpărătorul sau beneficiarul de software.

#### **4. Auditul software**

Un produs software este de fapt un program principal care apelează proceduri. A audita un produs software înseamnă a parcurge etapele codului de audit având ca obiect mai întâi un text sursă care, dacă trece de etapele de analiză, după compilare și editare de legături, este lansat în execuție și este auditat ca produs finit, validat pentru a obține a anumită funcție de prelucrare.

Analiza bazată pe revizuirea textului sursă are ca rezultat o serie de comentarii din partea membrilor echipei sau clienților asupra modului în care a fost conceput și elaborat produsul informatic.

Evidențierea comportamentului statistic al programului, privit ca automat nedeterminat, se realizează în cazul în care se caută disfuncționalități prin parcurgerea textului sursă sau prin discuții asupra locului și rolului unor instrucțiuni.

Un rol special îl are revizuirea textului sursă, care constă în parcurgerea textului pas cu pas, evidențind etapele algoritmului și evaluând capacitatea de generare a erorilor.

Inspectarea codului sursă are ca obiectiv creșterea calității software prin aplicarea unor reguli verificate și neluate în considerare de către programatorii care au dezvoltat produsul analizat.

Avantajele inspecției sunt enumerate în continuare:

- îndepărtarea timpurie a defectelor majore;
- evaluare imediată;
- mai economică și eficientă din punct de vedere al costurilor;
- îmbunătățește calitatea produsului;
- contribuie la dezvoltarea organizației spre realizarea unor produse mai performante;
- îmbunătățește procesul;
- instruește personalul implicat la realizarea produsului.

Pornind de la specificațiile de programare în care se prezintă datele de intrare ale aplicației, rezultatele, modelele de calcul, seturile de date de test și rezultatele ce trebuie obținute, prin inspecția textului sursă se pun față în față componentele specificațiilor cu secvențele corespondente din program și se identifică următoarele tipuri de situații:

1. tuturor secvențelor de texte din specificații le corespund secvențe de instrucțiuni în programul de sursă. În acest caz auditul consemnează că programul realizează cerințele definite în specificații;
2. numai anumitor secvențe de text din specificații le corespund părți de program sursă ceea ce înseamnă că programatorii nu au dezvoltat un produs care să realizeze toate funcțiile de prelucrare care sunt descrise prin specificații;
3. există mai multe secvențe de cod decât cele care sunt cerute prin specificații. Un astfel de caz se explică prin existența unor situații în care programatorii intuiesc o

serie de prelucrări necesare programului, neincluse în specificații datorită abordării de suprafață a problematicii în textele specificațiilor.

Aceleași situații se întâlnesc și în cazul proceselor de testare. Dacă în planul de testare sunt descrise seturi de date de test, asta nu înseamnă că testerii chiar urmează planul de testare. Auditul pe textul sursă are menirea de a analiza modul în care au fost introduse datele de test, ce proceduri au activat și mai ales de a stabili caracterul parțial sau caracterul complet al prelucrărilor.

Auditul pe textul sursă punctează care sunt minusurile din textul sursă sau care sunt definițiile în plus, fără a da soluții, adică fără a genera secvențele lipsă pentru a arăta cum trebuie rescris programul. Auditorul care își propune să efectueze o inspecție a textului sursă trebuie să îndeplinească o serie de condiții precum:

1. cunoașterea în detaliu a limbajului de programare în care este scris programul auditat;
2. să se familiarizeze rapid cu stilul de programare adoptat pentru scrierea codului de către echipa de programatori care a realizat programul;
3. să cunoască efectele de antrenare pe care le generează conturarea de secvențe echivalente sau să stabilească efectele unor procese specifice introducerii recursivității sau compunerii de instrucțiuni;
4. să fie gata să accepte o multitudine de atitudini din partea programatorilor pornind de la decizia de a rescrie programul ca produs integral nou și până la reutilizarea la maximum a componentelor existente în bibliotecă.

Auditorul are o listă de erori probabile însoțite de modul direct de reproducere. Prin inspecție se analizează textul sursă pentru a evidenția modelele de program care includ construcții generatoare de erori.

Experiența în activitatea de programare conduce la conturarea listei de evenimente (erori)  $E_1, E_2, \dots, E_m$  împreună cu frecvențele de operare deduse prin analiza unui set de  $N$  produse program. Evenimentele se ordonează după frecvențele  $f_1, f_2, \dots, f_m$  de apariție astfel încât  $f_1 > f_2 > \dots > f_m$ . Pe durata inspecției, auditorul se orientează să identifice secvențele care generează evenimentele  $E_1, E_2, \dots, E_l$  și nu neglijează nici evenimentul  $E_m$  care are probabilitatea minimă de apariție. Inspecția software este asemenea investigațiilor unui medic care pornește în stabilirea diagnosticului de la cazurile cele mai probabile, lăsând spre finalul analizei cauzele neprecizate ale efectelor.

Inspecția software presupune o vastă experiență și o cultură a textelor sursă. Acumulările au menirea de a crea un stil de programare. Activitatea de programare devine o activitate de aplicare a unor rețete. Auditul pe text sursă revine la a verifica dacă la fiecare pas din algoritm a fost aplicată rețeta cea mai potrivită. Auditul pe text sursă reprezintă un pas spre a stabili gradul de siguranță pe care îl oferă programul atunci când se rulează seturile de date.

Inspecția are menirea de a localiza funcțiile de bază, de a construi tabele de dependență și de a identifica disfuncționalitățile care apar dacă în tabel apar linii sau coloane necompletate. Tabelele de dependență stabilesc modul în care instrucțiunile  $I_1, I_2, \dots, I_n$  ale unui program utilizează variabilele  $V_1, V_2, \dots, V_m$  (Tabelul 1).

Elementul  $a_{kj}$  are valoarea 0 dacă instrucțiunea  $I_k$  nu folosește variabila  $V_j$ , respectiv 1 în cazul în care instrucțiunea  $I_k$  referă variabila  $V_j$ . Atunci când o coloană  $r$  conține numai zerouri înseamnă că variabila  $V_r$  este definită și nefolosită. Dacă instrucțiunile nu folosesc variabile, auditorul trebuie să studieze atent semnificația instrucțiunii sau rolul funcției apelate și mai ales să determine oportunitatea poziției în program a unei astfel de linii sursă.

Inspecția pe textul sursă este profesională, are un caracter strict iar observațiile trebuie să fie clare și fără echivoc.

	$V_1$	$V_2$	...	$V_j$	...	$V_n$
$I_1$						
$I_2$						
...						
$I_k$				$a_{kj}$		
...						
$I_n$						

**Tabloul 1 – Tabel de dependență între instrucțiuni și variabile**

Realizarea testului de birou prin simularea trecerii de la o instrucțiune la alta vine să evidențieze mai ales evaluările de expresii condiționale complexe, care, atunci când sunt construite eronat, generează traversări, de asemenea eronate, ale ramurilor structurii arborescente asociată programului. Testul de birou și analiza textului sursă reprezintă numai o componentă a procesului de auditare.

Se consideră un program  $P$  căruia i se asociază o structură arborescentă la baza căreia se află  $K$  module care afișează  $K$  tipuri de rezultate finale. Pentru a testa complet programul  $P$  se construiesc  $K$  seturi de date de test  $DS_1, DS_2, \dots, DS_k$  și se imprimă urma programului împreună cu rezultate scontate. Urmă programului este dată de o serie mesaje care permit reperarea instrucțiunilor care au fost executate.

Testarea acestui program folosind seturile de date de test  $DST_1, DST_2, \dots, DST_k$  conduce la obținerea urmei programului pe baza mesajelor care evidențiază executarea anumitor instrucțiuni. Pentru fiecare set de date de test se verifică corectitudinea programului prin compararea șirului de mesaje obținut în urma execuției cu șirul care a fost generat în specificații. Prin compararea celor două șiruri se stabilește gradul de concordanță al programului cu specificațiile.

Dacă are loc pe măsură ce se elaborează modulele unui program, atunci inspecția pe textul sursă impune efectuarea de corecții pentru eliminarea erorilor iar evaluarea efectelor este imediată. Inspecția este o activitate de grup, cu mai mulți participanți, ce conduce la perfecționarea programatorilor. Eliminarea erorilor sau obligativitatea de a folosi anumite construcții conduce la creșterea calității programului. De exemplu, dacă înaintea efectuării unei împărțiri este testat numitorul, produsul program va genera continuitate în procesul de prelucrare, iar apariția unei astfel de situații este controlată și este precedată de un mesaj. Introducerea de reguli verificate în a defini variabile și în a construi secvențe omogene, are rolul de a îmbunătăți procesul de elaborare a codului sursă.

Aplicarea regulilor conform cărora procedurile de calcul nu pot conține apeluri de funcții pentru citire de fișiere sau pentru scriere în fișiere conduce la ameliorarea rapidă a procesului de scriere cod sursă.

Inspecția pe codul sursă scurtează intervalul dintre momentul în care s-a produs eroarea și momentul corectării. Este cunoscut faptul că, cu cât o eroare este înlăturată mai târziu, cu atât efectele ei sunt mai profunde asupra întregului produs.

Pentru ca inspecția să fie tratată cu maximă seriozitate trebuie ca persoanele care o efectuează să fie specialiști recunoscuți. Este foarte dificil ca un programator slab sau o persoană care nu vine din zona programării să procedeze la inspecție de cod sursă, întrucât carențele de pregătire conduc la interpretări eronate, ce îndepărtează momentul depistării erorilor de cel al efectuării de corecții. Procedurile verificate și perfecționate, care sunt căutate în programele supuse inspecției, formează arsenalul unui programator foarte bun cu bogată experiență.



Și în cazul inspecției există o abordare pe etape ce trebuie parcurse: planificarea inspecției, evaluarea generală, pregătirea inspecției și derularea procesului propriu-zis. Dintr-un studiu Fagan reiese că aplicarea inspecției software asupra unei componente de complexitate medie a unui sistem de operare a dus la creșterea cu 23 % a productivității și o îmbunătățire cu 38 % a calității componentelor. Pentru ca o metodă de inspectare a unui produs software să aibă o productivitate mare, este necesar ca aceasta să îndeplinească următoarele condiții:

- să existe garanția că metoda de inspectare se poate aplica riguros, astfel încât rezultatele să fie specifice fiecărui produs particular și, în același timp, repetabile la nivelul aceluiași produs;
- să fie flexibilă, să poată servi unor scopuri complexe, nu numai unei simple detectări a erorilor înainte de faza de testare;
- să fie concepută astfel încât o mare parte a muncii să poată fi făcută automat, resursele umane să fie utilizate numai atunci când este imperios necesar;
- să fie eficientă, astfel încât, cu un anumit volum de resurse, să se poată obține rezultate maxime.

Pentru a decide ce componente software să fie inspectate și ce tip de metodă să se utilizeze, se consideră următoarele criterii referitoare la risc:

- componente care utilizează tehnologii, tehnici și instrumente noi;
- componentele arhitecturale de bază;
- componente sau cerințe critice (securitate, siguranță) cu mare risc de defectare;
- cod de tratare a excepțiilor care nu poate fi ușor testat;
- componente care se intenționează a fi reutilizate;
- componente care servesc ca model pentru alte componente;
- componente care afectează multiple zone ale aplicației;
- interfețe utilizator complexe;
- componente create de programatori mai puțin experimentați;
- module cu complexitate ciclomatică ridicată;
- module care au avut multe defecte sau modificări de-a lungul timpului.

Aplicațiile sau componentele care pot fi în una sau mai multe dintre categoriile mai sus menționate sunt considerate de risc ridicat. Un produs este considerat de risc scăzut dacă o eroare nedetectată nu va afecta abilitatea proiectului de a respecta graficul, calitatea, bugetul și funcționalitățile planificate.

Revizuirile au diferite forme și nume ca de exemplu revizuire formală, inspecții, audituri și walkthroughs. În fiecare dintre aceste categorii, termenii au diferite înțelesuri și conotații. Cele mai importante caracteristici care diferențiază tipurile de revizuirii sunt scopul utilizării, domeniul de aplicare și metoda. În general, scopul și domeniul de aplicare al revizuirii vor determina metoda folosită și gradul de formalism aplicat, așa cum se prezintă în Tabelul 2.

Diferențele tipuri de revizii utilizate în dezvoltarea de software variază de la simple revizii tehnice până la cele oficiale, formale, cum este cel de configurație fizică a sistemului. Toate aceste tipuri de revizii sunt puternic influențate de către mărimea și complexitatea proiectului. Pentru un proiect intern al companiei, reviziile tind să fie mai simple în comparație cu cele pentru un proiect aflat sub contract. Deși sunt diferite motive pentru a conduce o revizie, principalul scop rămâne acela de a evalua avansul și integritatea unui produs sau proces. Reviziile sunt folosite și pentru a obține date. Colectarea sistematică de date este esențială pentru evaluarea viitoare a întregului proces de dezvoltare.

Implementarea inspecțiilor software într-o firmă de dezvoltare software este o provocare. Această implementare cere nu numai o foarte bună înțelegere a tehnologiilor de dezvoltare software dar și o foarte bună evaluare a comportamentului, motivării și culturii organizaționale a personalului implicat

Tipul	Domeniu de aplicare	Utilizat	Metoda
Review	Destul de vast	Pentru evaluarea îndeplinirii obiectelor în cadrul evoluției proiectului	Ad hoc
Walkthroughs	Destul de vast	Pentru evaluarea proiectelor de dezvoltare	Analiză statistică a produselor
Inspecții	Restrâns	Pentru evaluarea proiectelor de dezvoltare	Neinteractivă, destul de procedurală
Audit	Restrâns către vast	Verifică procesele și produsele în timpul dezvoltării	Formală și procedurală

**Tabelul 2 – Tipuri de revizii**

Deși inspecțiile software sunt metode de gestiune și management al calității orientate către produs, faptul că toți participanții la inspecții își îmbunătățesc, în timp, într-un mod sau altul, abilitățile personale (profesionale, de comunicare, de conducere), efectul major al acestora se va reflecta, pe termen mai lung, asupra calității proceselor.

Prin inspecția software se realizează:

- îmbunătățirea semnificativă a calității produsului;
- micșorarea fazei de dezvoltare a produsului;
- îmbunătățirea productivității;
- reducerea costurilor cu realizarea produsului în diferite etape, cum ar fi cele de analiză, proiectare, generare cod și testare;
- implicarea unor persoane într-un stil de lucru foarte organizat.

Pentru fiecare limbaj de programare utilizat se pot stabili reguli și convenții referitoare la cod, care să includă minim următoarele:

1. reguli și convenții pentru formatarea textului sursă: indentare, spațiere, majuscule, ordinea informațiilor;
2. reguli și convenții pentru comentarii, cum ar fi numele și identificatorul modulului, identificatorul versiunii, istoricul modificărilor, scop, specificații funcționale și decizii de proiectare implementate, note referitoare la programare (algoritmi utilizați, ipoteze, constrângeri, efecte colaterale), note referitoare la date (intrări, ieșiri, variabile, structura datelor);
3. convenții pentru denumirea variabilelor, procedurilor, modulelor și fișierelor;
4. limitări, dacă există, în utilizarea anumitor facilități ale uneltelor de programare utilizate;
5. limitări, dacă există, ale complexității codului.

Inspeția aplicației informatice va urmări dacă aceste reguli și convenții referitoare la cod sunt respectate și dacă cerințele considerate critice (performanțe, securitate, siguranță, fiabilitate, confidențialitate) sunt tratate. Dacă firma are create standarde pentru proiectare, programare și testare, atunci respectarea acestora se va urmări tot în cadrul acestei etape.

După efectuarea inspecției, programatorii procedează la corectarea erorilor. Pe parcursul fazelor de mai sus pot fi necesare mai multe cicluri de verificare, corectare și îmbunătățire până la finalizarea documentelor propuse: specificații funcționale, arhitectura sistemului, proiectarea detaliată. Reinspecția presupune ca volumul erorilor de la o inspecție la alta să fie cu tendință de amortizare adică să scadă devenind aproape nul în final.

## **5. Verificarea și validarea aplicațiilor informatice**

Aplicațiile informatice sunt construcții complexe care includ software și seturi de date organizate în fișiere, baze de date sau alte structuri a căror dinamică și conținut ameliorează duratele tranzacțiilor. Aplicația informatică soluționează probleme ce aparțin unei clase riguros definită.

Există aplicații informatice pentru rezervarea de bilete, rezervarea de locuri la hoteluri, optimizare folosind programarea liniară, pentru implementarea unei metode de analiză statistică pentru accesarea anumitor date de pe internet, pentru efectuarea unor calcule privind salariații, pentru managementul unei activități bine delimitate din companiile orientate pe servicii sau producție.

Elaborarea și implementarea de aplicații informatice creează premisele reale pentru dezvoltarea de sisteme informatice pentru management deoarece:

1. prin agregarea de aplicații informatice ale companiei se obțin subsisteme informatice de management; prin agregarea de subsisteme de management se obține sistemul informatic în integralitatea sa; sistemul este integrat datorită faptului că abordează totalitatea aspectelor ce se manifestă la nivelul companiei;
2. experiența acumulată prin analiza unei probleme, proiectarea soluției și traversarea celorlalte etape ale codului de dezvoltare software fac să se obțină acel nivel de experiență care să deschidă orizonturi noi pentru analiza și mai ales proiectarea unui sistem informatic; echipa care dezvoltă o aplicație informatică are șansa să dovedească unitate, capacitate de analiză și sinteză și în special performanță în a respecta standarde și a utiliza coerent tehnicile și metodele cele mai noi ale ingineriei software.

În cadrul codului de dezvoltare a aplicațiilor informatice apar două aspecte și anume:

1. atitudinea echipei implicate în dezvoltarea de procese care au ca obiectiv asigurarea calității de concepție, de execuție, de conformitate, a capacității de utilizare și de mentenanță; se planifică nivelurile caracteristicilor de calitate pentru a ale atinge pe durata proceselor de realizare a produsului; echipa de specialiști cunoaște modalitățile practice prin care să definească obiective, să asigure atingerea acestora și implicit să satisfacă cerințele beneficiarului; prin respectarea fluxurilor specifice a procesului de dezvoltarea a aplicației informatice, a sarcinilor care decurg din fiecare etapă prin definirea intrărilor și ieșirilor etapelor și prin executarea strictă a activităților, prin respectarea procedurilor definite la nivelul companiei de software, se asigură condiția obținerii unui nivel corespunzător pentru calitatea de execuție; elaborarea specificațiilor impune o serie de restricții care orientează pe cel ce elaborează soluții, texte sursă și de structuri ale seturilor de date, spre direcții clare privind resursele pe care le alocă pentru a obține atât operaționalitatea, cât și mentanabilitatea produsului finit;
2. verificarea mai întâi a respectării de către membrii echipei a tuturor cerințelor strict din punct de vedere tehnic în ceea ce privește latura calitativă a abordării cât și în ceea ce privește latura cantitativă a acesteia; a verifica înseamnă a prelua ceea ce s-a executat sau informația privind modul în care s-a produs execuția și a

compara cu proceduri, standarde sau norme descrise în manualele de implementare a unor tehnici de analiză-proiectare-codificare-testare; rezultatul verificării este fie identificarea de diferențe între ceea ce s-a făcut și cum trebuia făcut, fie confirmarea unei suprapuneri perfecte, ceea ce marchează faptul că echipa de dezvoltare a aplicației informatice a lucrat perfect; validarea aplicației informatice este consecința verificărilor și se produce atunci când diferențele dintre cum s-a lucrat și cum trebuia lucrat precum și diferențele dintre ceea ce s-a obținut, respectiv ceea ce trebuia obținut, nu sunt semnificative în raport cu criteriile de exigență stabilite și cunoscute atât de auditori, cât mai ales de echipa care dezvoltă aplicațiile informatice.

Verificarea are menirea de a analiza module, interacțiuni, seturi de date și rezultate parțiale pentru a putea stabili diferențele care apar în cazul componentelor între nivelurile efective și nivelurile planificate ale unor caracteristici de calitate.

Verificarea stabilește compatibilitatea și precizia componentelor aplicației informatice, a datelor de intrare și a rezultatelor față de standarde, norme și restricții impuse de tehnicile utilizate de-a lungul etapelor proceselor.

Validarea este un proces de evaluare, în care aplicația informatică este tratată ca un produs finit. Dacă sunt asigurate cerințele de conformitate cu condițiile specifice, procesul de validare conduce la acceptarea aplicației informației pentru utilizarea curentă, ea asigurând un bun grad de satisfacere a exigențelor utilizatorilor.

Verificarea aplicației informatice se realizează prin efectuarea de teste, prin analiza structurii, a modulelor, a interacțiunilor dintre module și prin analiza procesului care stă la baza ciclului de dezvoltare. Se verifică nivelul concordanței dintre cerințele formulate de beneficiar și reflectarea acestora la nivelul specificațiilor. Se stabilește concordanța dintre structura proiectului și structura sistemului așa cum rezultă din specificații. Este important să se definească fluxurile care se parcurg la nivelul aplicației informatice astfel încât testele să acopere totalitatea elementelor definite prin specificații.

Verificarea la nivelul aplicației informatice este un proces iterativ convergent, fiecare verificare fiind urmată de corectarea abaterilor înregistrate, dacă aceste abateri reprezintă de fapt minusurile modulelor sau structurilor seturilor de date în raport cu definițiile incluse în specificațiile aplicației.

Testele se efectuează pentru a măsura în care componentele aplicației informatice (date de intrare, module program, date de ieșire) satisfac specificațiile și de a stabili care sunt diferențele între ceea ce se aștepta să se obțină prin utilizarea aplicației și ceea ce s-a obținut în mod concret.

Testele utilizează module program, structuri de date, exigențele de acceptare, seturi de date de test și documentația care alcătuiește ghidul utilizatorului. Procesul de testare are la bază un plan care include: definirea obiectivului testării, etapele prin care se efectuează testarea, intrările și ieșirile testării. Planul include, de asemenea, modul de finalizare a analizei rezultatelor obținute.

Planurile de testare trebuie să conțină următoarele date :

- obiectivele urmărite de fiecare tip de teste (teste unitare, teste de integrare, teste de sistem, teste de acceptare, alte tipuri de teste);
- criteriile prin care se determină când se încheie o fază de testare;
- graficul de lucru;
- responsabilități (cine răspunde de fiecare fază);
- resursele necesare, cum ar fi
  - o programele auxiliare, inclusiv instrumentele de testare;

- configurațiile hardware;
- timpul disponibil;
- personalul necesar;
- strategia de testare, inclusiv procedurile pentru
  - identificarea, generarea și documentarea cazurilor testate;
  - urmărirea rezultatelor;
  - testarea performanței la stress a programelor;
  - testarea regresivă;
  - riscurile asociate procesului de testare;
- producerea documentației;
- procedurile de testare.

Este posibil ca modificările efectuate după testarea unui program să submineze calitatea acestuia. Orice modificare, indiferent de cât de neînsemnată, poate anula testarea componentei respective. Pentru a evita acest pericol, programul testat – aflat în una din stadiile testării (unitarî, integrare, sistem, acceptare) – trebuie considerat ca element supus managementului configurației. În această situație, programul este un proiect identificabil care nu mai poate fi modificat decât în urma unor proceduri clare.

În planul de testare sunt evidențiate distinct, ca teste de acceptare, procesele de la nivelul unităților, de la nivelul întregii aplicații și de la nivelul interacțiunii dintre aplicația informatică și utilizator. Planului de testare pentru aplicația informatică îi corespunde un program de testare în care obiectivele și etapele iau forme concrete prin enumerarea de componente ale aplicației, prin persoane care derulează activitățile de testare, prin precizarea obiectivelor urmărite în fiecare caz în parte.

Testarea aplicației informatice se derulează conform planului. Rezultatul testării, ca formă concretă de verificare a aplicației informatice, este un set de date privind atât comportamentul acesteia în execuție, pe seturi de date impuse, cât și structura și resursele utilizate la construirea sa. Aceste rezultate conduc, prin agregare, la obținerea unor indicatori. Pe baza nivelurilor indicatorilor se decide dacă aplicația informatică este acceptată sau nu, adică este livrabilă clientului, existând garanția că se va obține un grad de satisfacere a cerințelor acestuia suficient de ridicat ca să justifice investiția respectivă. Riscurile de a obține rezultate eronate sunt prea mari, ele afectând grav modul de satisfacere a acestor cerințe.

Auditul aplicației informatice traversează etapele oricărui proces de auditare. Într-un fel este privită o aplicație informatică în care utilizatorul este operatorul instruit printr-un curs de scurtă durată pentru a ști exact cum se utilizează produsul, și în alt fel sunt analizate aplicațiile web care se adresează unui grup țintă neomogen, deoarece criteriile urmărite diferă. Orice proces de audit este derulat dacă și numai dacă este fezabil. Fiecare aplicație informatică este însoțită de o documentație cadru care definește condițiile de elaborare, sarcinile elaboratorului și cele ale beneficiarului. De asemenea echipa de audit analizează și documentele relevante de ordin tehnic prin care se evidențiază modul în care au fost parcurse etapele ciclului de dezvoltare. Este cunoscută reticența echipelor de specialiști care proiectează, programează, testează și implementează aplicații informatice în a da o formă finală documentației tehnice deoarece se consideră că activitățile și produsele finite sau stadiile pe care acestea le-au produs nu sunt suficiente pentru a crea o imagine completă a întregului efort depus.

Sunt necesare texte și reprezentări grafice complete, concise și consistente prin care să se prezinte, pas cu pas, tot ceea ce s-a efectuat, tot ceea ce s-a obținut și mai ales cum s-a procedat și cine a executat fiecare operație.

Auditul aplicației informatice se efectuează la fața locului, pentru a se vedea exact pe ce echipamente s-a lucrat și ce medii de programare au fost folosite. Există premise reale de a

obține informațiile necesare completării tabelor pe baza cărora se elaborează raportul de audit. Efectuarea auditului presupune o transparență totală a echipei de auditori față de echipa care a realizat aplicația informatică. Orice altă abordare conduce la obținerea de rezultate neconcludente, la redefinirea contextului și mai ales la elaborarea unor concluzii false, care nu vor juca un rol activ în dezvoltarea aplicației și în procesul de reinginerie.

Etapelile de derulare a auditului privesc interacțiunile ale echipei de auditori cu cei care au dezvoltat aplicația în așa fel încât, printr-o abordare gradată, să se obțină toate datele necesare stabilirii concordanței dintre ceea ce se solicită prin specificații și ceea ce există în realitate: programe, fișiere, baze de date și structuri de rezultate finale. Raportul de audit trebuie să consemneze toate diferențele. El nu soluționează problemele, dar creează bazele corecte ale derulării proceselor de eliminare a erorilor și de creștere a nivelurilor unor caracteristici de calitate. Raportul de audit se difuzează și constituie document care însoțește procedurile de predare a aplicației informatice spre client.

Auditul aplicației informatice vine ca o apreciere efectuată de un organism independent asupra concordanței dintre calitatea planificată și calitatea efectivă a aplicației. Auditul reprezintă o concluzie asupra siguranței la care trebuie să se aștepte clientul în a obține rezultatele de calitate și cantitatea pe care și-a dorit-o.

Auditul calității software are câte ceva de oferit fiecărui participant: utilizatorul este asigurat că produsul îndeplinește nevoile sale operaționale și de performanță, cumpărătorul este asigurat că produsul va fi livrat la timp și în limitele bugetului alocat, iar producătorul este asigurat că produsul s-a dezvoltat într-o manieră trasabilă, fiind deci mentenabil, și va fi acceptat de utilizator și cumpărător. Costul auditului este prețul pe care utilizatorul, cumpărătorul și producătorul trebuie să-l plătească pentru a-și reduce riscurile.

## **6. Auditul sistemelor informatice de management**

Sistemele informatice pentru management sunt construcții deosebit de complexe care au ca obiectiv ridicarea la cote maxime a procesului de informatizare la nivelul organizațiilor.

Dacă o organizație este caracterizată prin funcțiile  $F_1, F_2, \dots, F_k$ , structura sistemului informatic pentru management include  $K$  subsisteme,  $SSI_1, SSI_2, \dots, SSI_k$ , câte un subsistem pentru fiecare funcție. Întregul sistem informatic este proiectat sub forma unor subsisteme cu legături între ele.

Abordarea sistemelor pentru management presupune un fundament teoretic și practic deosebit de solid și acceptarea unui demers de anvergură pe o perioadă de doi-cinci ani. Tehnicile și metodele de analiză și proiectare au la bază o cunoaștere în detaliu a stadiului actual atins de procesul de informatizare la nivelul organizației. Dacă în unele momente a fost necesară efectuarea unei analize critice, stadiul actual impune o cu totul altă abordare.

Există sisteme puternice de gestiune a bazelor de date, de soluționare a problemelor definite în cadrul fiecărei funcții din organizație. Trecerea la dezvoltarea unui sistem informatic pentru management tehnic trebuie să ia în considerare existența acestor componente. Platformele *SAP* și *ORACLE* oferă soluții *ERP*, iar reluarea unor cicluri complete de creare de module în afara concepției unitare pe care o oferă una din platformele menționate conduce la menținerea procesului de informatizare în aria meșteșugărească, neadecvată stadiului atins acum în zona ingineriei sistemelor informatice.

În același fel se pune problema și în cazul în care se dorește dezvoltarea de sisteme de gestiune a documentelor, din moment ce există deja astfel de sisteme livrate la cheie, care își așteaptă cumpărătorii. A spune acum că o organizație are particularități ce impun definirea unei structuri proprii de sistem informatic înseamnă a considera că echipele care au proiectat sisteme suport de decizie care se aplică în peste 30 de țări nu sunt competente, iar organizația este atât de specială încât nu se încadrează în nici o categorie. Abordarea este nu numai

absurdă prin simplismul ei dar denotă un nivel de ignoranță greu de acceptat din punctul de vedere al informaticii anului 2005.

Problemele de audit pentru un sistem informatic de management au o altă anvergură față de celelalte entități, produsul program sau aplicația informatică. Literatura de specialitate include numeroase lucrări care se adresează celor care elaborează sisteme informatice orientate pe gestiune financiar-contabilă. Auditul acestor sisteme este o problemă extrem de actuală pentru că:

1. sistemele informatice de gestiune contabilă pot conduce la efectuarea de operații neautorizate;
2. sunt situații în care nu există concordanță între teoria contabilității și procedurile care se apelează pentru a efectua prelucrări;
3. în faza de analiză sunt definite incomplet cerințele care corespund laturilor calitative și cantitative definite prin relația între conturi, prin restricții privind efectuarea de operații și prin proporții impuse unor niveluri cu care se efectuează debitările sau creditările;
4. priorităților de efectuare a operațiilor existente în teoria contabilă trebuie să le corespundă secvențe de testare care să asigure concordanța între listele priorităților existente în teoria contabilă și listele generate prin procesele de prelucrare prin programe;
5. validările datelor capătă o altă semnificație, fiind legate nu numai de apartenența la un anumit domeniu de variație, ci fiind dependente de context, întrucât operațiunile contabile sunt definite în cadrul unui anumit context;
6. interfețele acestor sisteme trebuie să fie orientate spre o abordare a proceselor în timp real, întrucât numeroase operații se derulează prin sistemele e-banking și e-commerce; operatorii trebuie să lucreze în regim responsabilizat cu înregistrarea operației într-o structură impusă din care să nu lipsească momentul efectuării operației și elementele de identificare a operatorului;
7. între sistemul de restricții de acces la efectuarea de operații în baza de date și cerințele teoriei și practicii contabile trebuie să existe o concordanță perfectă; trebuie să existe persoane care au acces la consultarea întregii baze de date; trebuie să existe alte persoane care au acces la consultarea numai a unor părți din baza de date, sunt alte persoane din organizație care au drept de a consulta numai operațiile care privesc activitatea lor; lista persoanelor care operează pe baza de date se definește astfel încât, pe măsura creșterii importanței operației în baza de date, numărul și funcția în organizație se definesc cu un nivel de exigență sporită, regulile impuse au menirea de a ține sub control totalitatea operațiilor pe câmpurile bazei de date;
8. sistemul informatic de gestiune financiar-contabilă se proiectează incluzând numeroase chei de control care să evidențieze frecvențele unor operațiuni, apropierea de limitele domeniilor de variație, astfel încât să se ia rapid deciziile adecvate;
9. la proiectare și la realizare se definesc situațiile de blocare pentru a semnaliza tentativele de efectuare a operațiilor interzise;
10. aceste sisteme sunt organizate ca structuri ierarhice, cu intervenții de asemenea ierarhice. Astfel, dacă s-a produs un eveniment la nivelul  $K+1$ , atunci numai un administrator de la nivelul  $K$  al structurii arborescente intervine și produce deblocarea sau efectuează operația care trebuie autorizată pentru a readuce sistemul la nivelul de operare normală; toate procesele de blocare/deblocare sunt înregistrate și se tratează distinct.

Rezultă că un sistem informatic, fie că este un sistem general pentru management sau unul special destinat managementului financiar contabil, trebuie înzestrat, pe lângă funcțiile clasice de prelucrare, de extragere a rezultatelor și de creare-actualizare a bazei de date, cu funcții de management pentru calitatea și protecția sistemului informatic însuși.

Marile probleme rezultate în activitatea curentă a implementării de software pentru contabilitate au impus dezvoltarea auditului spre această categorie de produse program. Există o preocupare specială pentru auditul sistemelor informatice de gestiune financiar-contabilă. Și celelalte sisteme informatice sunt auditate. Principiile auditului sistemelor informatice de gestiune sunt o particularizare a principiilor auditului pentru sistemele informatice pentru management.

Așa cum în modul clasic de operare pe documente există riscul transferurilor de fonduri și de mijloace care generează fraude împotriva companiei, fraude împotriva altor companii, fraude ale managerilor, fraude ale unor membri ai companiei, în cazul implementării unui sistem informatic de gestiune contabilă, toate aceste tipuri de fraude se reproduc dacă și numai dacă sistemul nu conține procedurile care să semnaleze efectuarea de operații neconsistente în raport cu o serie de criterii precis stabilite.

În primul rând, legile definesc foarte clar situațiile în care o persoană nu are drept să ia un credit. Sunt date reguli extrem de precise în a defini un creditor ca fiind rău-platnic. Sistemul informatic dintr-o bancă trebuie să includă proceduri prin care se verifică statutul solicitantului și încadrarea acestuia în una din categoriile următoare:

1. rău platnici;
2. creditorii al căror plafon de creditare a fost atins;
3. creditorii care mai pot solicita un credit dar nu mai mare decât o valoare impusă;
4. creditorii care au dreptul să solicite credit cu valoare care se situează într-un interval definit de garanții, de cifra de afaceri și de istoricul relației lor cu banca;
5. creditorii falșiți.

Evident, rolul auditului unui astfel de sistem este de a testa dacă respectivul sistem bancar include astfel de proceduri. Datele de test sunt date reale cu care se operează în banca unde sistemul informatic va fi operațional.

Sistemul informatic bancar nu trebuie să valideze efectuarea unor operații interzise prin acte normative, dintre care operația de efectuare de plăți ale ratelor unui credit cu banii obținuți dintr-un alt credit. Auditorii sistemelor informatice bancare au deja inclusă această operație în lista operațiilor interzise, listă folosită cu prioritate în testarea comportamentului unui sistem informatic bancar.

Un sistem informatic de management financiar-contabil auditat devine credibil când echipa conchide în raportul de audit că sistemul răspunde tuturor cerințelor din specificații, din legi și regulamente, securitatea operațiilor este asigurată iar condițiile de risc în utilizare sunt minime.

Auditul sistemelor de gestiune financiar-contabilă are menirea de a oferi încredere utilizatorului în produsul informatic. De aceea trebuie supuse auditării toate componentele sistemului, intrările și ieșirile acestora. Numai prin coborârea auditului la nivelul detaliilor se vor obține informațiile necesare fundamentării unei concluzii finalizate printr-o propoziție simplă, fără echivoc, de calificare a sistemului.

Prin specificații este creată o imagine, un sistem informatic virtual. Dacă se adaugă noi cerințe desprinse din legislație, din experiența curentă, dacă se produce o ierarhizare a priorităților privind operații permise, respectiv, operații interzise, se creează proiecția unui sistem informatic virtual și ideal. Toate comparațiile sistemului real sunt efectuate strict față de coordonatele pe care le oferă ca reper sistemul virtual ideal.



Auditul unui sistem informatic de gestiune financiar-contabil nu are rolul de a controla. Esența auditului nu este controlul. Auditorii sunt persoane cu înaltă calificare care nu se substituie controlorilor de calitate, controlorilor care stabilesc existența fizică a unui produs, exprimând-o prin cantitate, după măsurare. Auditul este o activitate superioară de orientare, analiză și de sinteză. Este o necesitate tocmai prin extensiile pe care le determină asupra întregii viziuni de abordare. Planul de audit și programul de audit presupun activități clare, nici una dintre acestea nefiind de control.

Specificațiile reprezintă un text structurat iar sistemul informatic reprezintă o structură. Auditorul are menirea de a stabili existența corespondenței dintre componentele textului structurat și, respectiv, componentele sistemului, identificând o concordanță perfectă, parțială, redusă sau chiar absența totală a concordanței.

Componentele din structura textului care alcătuiesc specificațiile includ:

1. nivelul managementului;
2. ciclul de elaborare a sistemului informatic de gestiune financiar-contabilă;
3. securitatea sistemului: precizarea responsabilităților, separarea funcțiilor incompatibile, ierarhizarea accesului la resursele sistemului;
4. nivelul operațional, prin procedurile pe care operatorii le efectuează în ceea ce privește: introducerea de date, manipularea de documente, manipularea mediilor de stocare pentru datele intermediare, înregistrarea evenimentelor, asistența tehnică;
5. nivelul aplicațiilor presupune parcurgerea de către auditor a tuturor etapelor astfel încât să se dezvolte convingerea că sistemul informatic de gestiune contabilă este chiar construcția în care utilizatorul trebuie să aibă mare încredere; se reia un ciclu complet de prelucrare, de la inițierea procesului, pregătirea datelor, procesarea acestora și obținerea rezultatelor: fișierele și bazele de date suferă o serie de modificări pe care analistul trebuie să le analizeze pentru a vedea dacă există sau nu și alte efecte secundare;
6. nivelul de acces presupune identificarea modului în care au fost soluționate elementele fundamentale ale accesului la resursele sistemului informatic - proceduri, baze de date, modul în care se dezvoltă și alte canale de transfer a informațiilor și cum se asigură robustețea rețelei de calculatoare.

Echipa de audit colectează date proprii dar preia și rezultate oferite de sistemul informatic de gestiune financiar-contabilă. Pe măsură ce se traversează etapele ciclului de dezvoltare, echipa de realizare a sistemului informatic elaborează părți ale documentației care însoțește sistemul. Echipa de audit analizează și această documentație pentru a urmări traseul parcurs de la specificații până la obținerea produsului finit în formă livrabilă către organizații.

Raportul de audit este un document sinteză care efectuează analiza comparată între un sistem virtual ideal și un sistem real. Toate datele înregistrate în procesul de auditare se coroborează cu specificațiile, cu documentația și se calculează o serie de indicatori asupra garanției sau asupra siguranței sau credibilității produsului numit sistem informatic pentru gestiunea financiar-contabilă. În final se spune că sistemul este sau nu credibil, asigură sau nu calitatea prelucrărilor, că există sau nu garanția ca sistemul informatic să dea satisfacție clientului în raport cu un obiectiv stabilit.

Auditul informatic este un demers deosebit de complex, motiv pentru care trebuie așezat pe un fundament solid. Obiectivul fundamental al activității de auditare informatică este stabilirea gradului de credibilitate a sistemului informatic de management. Fluxurile de informații specifice oricărui sistem informatic trebuie să asigure integritatea informațiilor organizației, completitudinea prelucrărilor, corectitudinea rezultatelor și mai ales

accesibilitatea beneficiarului la informația așteptată, obținând în acest fel un nivel maxim al satisfacerii cerințelor proprii.

Auditul sistemelor informatice include:

1. auditul intern, prin care se confirmă respectarea procedurilor de transformare a datelor de intrare în rezultate urmărindu-se modul în care noul sistem care se implementează este mai eficient, este însoțit de o economisire a resurselor;
2. auditul extern, care include proceduri prin care se evidențiază comportamentul sistemului informatic, prin testări cu ajutorul cărora se măsoară cât de stabile, cât de fiabile și mentenabile sunt procedurile de control care intră în componența sistemului și care implementează toate cerințele exprese incluse în specificații, în legi, în regulamente și care restricționează prin blocare orice tentativă de execuție a operațiilor interzise.

Pentru a dezvolta un proces de auditare a sistemului informatic de management, sunt parcurși următorii pași:

1. planificarea proceselor de auditare având la bază o serie de elemente prin care se stabilește anvergura prin cunoașterea unor elemente legate de complexitatea sistemului informatic și mai ales prin stabilirea nivelului de credibilitate pe care trebuie să-l stabilească auditorii sistemului;
2. evaluarea riscurilor legate de influențele negative care se manifestă asupra componentelor sistemului informatic ce vor fi auditate, pe măsură ce se activează procedurile de control;
3. elaborarea programului de audit ce include: definirea scopului, stabilirea obiectivelor, efectuarea planificării, derularea propriu-zisă, întocmirea de rapoarte;
4. culegerea de date ce evidențiază modul cum se execută prelucrările, care sunt neconcordanțele între specificații și produsul real; datele apar sub forme extrem de variate, de la liste, fișiere de tranzacții, liste de erori, chestionare care vizează obținerea unor răspunsuri cu cheie, diagrame, texte sursă, seturi de date de test, documentația care se livrează o dată cu implementarea sistemului informatic, ghidurile de utilizare și de administrare;
5. elaborarea raportului de auditare care preia elemente definite în planul de audit a sistemului informatic la care sunt adăugate detalii asupra modului cum s-a derulat procesul de auditare, gradul de transparență asigurat.

Întrucât auditorii de sisteme informatice pentru management sunt specialiști de înaltă calificare în domeniu, aceștia enumeră în raportul de audit totalitatea diferențelor care au fost întâlnite între sistemul real și sistemul virtual ideal. Raportul nu include și soluții, deși auditorii, prin competența lor deosebită, au capacitatea de a le oferi. Auditul sistemelor informatice pentru management consemnează numai diferențele. Caracterul sistematic al procesului de auditare oferă o grupare ascendentă în raport cu profunzimea efectelor de antrenare, a diferențelor iar în cazul în care sunt identificate erori, toate erorile sunt tratate distinct și contribuția lor în diminuarea nivelului de credibilitate a sistemului informatic pentru management este amplificată prin utilizarea de coeficienți de importanță cunoscuți atât de auditori, cât mai ales de cei care au dezvoltat sistemul informatic.

Activitatea de audit pentru sisteme informatice se bazează pe agregarea unor indicatori și pe obținerea de valori care să fundamenteze apartenența sistemului informatic la clasa de sisteme credibile în care se garantează calitatea soluțiilor așteptate de beneficiar. Pe de altă parte însă, raportul de audit poate concluziona că sistemul nu este credibil și că acesta trebuie să fie supus unor corecții urmate de un nou proces de auditare. Dacă tehnica de auditare și

procedurile de măsurare a diferențelor sunt clar definite, procesul de audit pentru sisteme informatice este reproductibil în proporție de 100%.

Asociațiile care au preocupări în a elabora tehnici și metode de auditare sau de a certifica specificații în auditul sistemelor informatice pentru management și-au concentrat atenția asupra algoritimizării proceselor de auditare, în viitor trebuind să se definească metrici necompensatorii, nediscriminatorii și necatastrofice, pentru a asigura un caracter obiectiv auditului, prin transferul din zona interpretărilor în zona certitudinilor.

## **7. Concluzii**

Auditul informatic reprezintă o ramură distinctă a auditului ce include tehnici și metode de auditare a aplicațiilor informatice, a sistemelor informatice tradiționale, a sistemelor informatice distribuite, a aplicațiilor mobile și a tuturor aplicațiilor informatice care utilizează resurse internet.

Pentru a realiza un audit informatic de calitate este necesar ca următoarele condiții să fie îndeplinite simultan:

1. să existe organisme recunoscute de certificare a auditorilor cu proceduri specializate de auditare pentru fiecare tipologie de componentă informatică;
2. auditorii să reprezinte persoane calificate în domeniul informatic deoarece acestea sunt cele mai potrivite pentru acest domeniu întrucât în procesul auditare sunt necesare elemente de teorie și practică în domeniu;
3. să existe instrumente pentru asistarea auditării cu computerul.

Pe măsura creșterii complexității proceselor din societatea informațională, cerințele sistemelor informatice impun un nivel de credibilitate deosebit de ridicat pe care numai auditul informatic îl poate susține cu succes.