

## Descrierea efectivă și eficientă a defectelor

Rapoartele defectelor sunt printre cele mai importante rezultate obținute din teste. Ele sunt la fel de importante ca și planul de test și au un impact **mai mare** asupra calității produsului decât celelalte rezultate ale testării. Merită efortul de a învăța cum se scriu rapoarte efective ale defectelor.

Rapoartele *efective* ale defectelor vor:

- Reduce numărul defectelor întoarse de la faza de dezvoltare;
- Mări viteza corectării erorilor;
- Ridica credibilitatea testului;
- Îmbunătăți legătura dintre testare și dezvoltare;

De ce unii testeri obțin un răspuns mult mai bun de la dezvoltatori decât alții? O parte din răspunsul la această întrebare o reprezintă raportul defectelor. Urmând câteva reguli simple putem să deschidem calea spre un mediu mult mai productiv. Obiectivul nu este să scriem un raport al defectelor perfect, dar unul efectiv, care redă un mesaj corect, face lucrul plăcut și ușurează procesul pentru fiecare.

Accentul trebuie pus pe două aspecte ale raportului defectelor:

- 1) Descriere
- 2) Rezumat

Mai întâi, să aruncăm o privire la punctele esențiale pentru scrierea unor rapoarte efective.

### Descrierea defectelor

Iată câteva puncte-cheie pentru a fi siguri că raportul defectelor scris va fi unul efectiv.

1. **Consistență** – Spune clar, dar pe scurt.
2. **Acuratețe** – Este acesta un defect sau o greșală a utilizatorului, neînțelegere, etc?
3. **Neutralitate** – Doar fapte. Fără înfrumusețări. Fără umor. Fără emoții.
4. **Precizie** – Concret, care este problema?
5. **Izolare** – Ce a fost făcut pentru a izola problema?
6. **Generalizare** – Ce a fost făcut pentru a înțelege cât de generală este problema?
7. **Reproducere** – Cum poate fi reprodusă problema (pașii concreți)?
8. **Impact** – Care este impactul asupra clientului? Care este impactul pentru testare?
9. **Depanare** – De ce este nevoie pentru a ușura depanarea? (urmărire, descărcare, log-uri, acces imediat, etc.)
10. **Evidență** – Ce documentație demonstrează existența erorii?

Nu doar capacitățile tehnice bune de scriere duc la rapoarte efective ale defectelor. Este mai important să te asiguri că ai pus întrebările corecte și ai răspuns corect la întrebări. Aceasta este cheia care asigură că au fost acoperite punctele esențiale care vor aduce un beneficiu maxim.

### Consistență

Spune clar, dar pe scurt. Mai întâi de toate, elimină excesul de cuvinte. În al doilea rând, nu adăuga informație redundantă. Este important să fie inclusă toată informația relevantă, dar asigurați-vă că informația este relevantă. În situații când nu este clar cum de reprodus problema sau înțelegerea problemei este vagă din oarecare cauză posibil veți avea nevoie să captați mai multă informație. Țineți minte că informația irelevantă poate fi la fel de problematică ca și informația relevantă insuficientă.

### Acuratețe

Asigurați-vă că ceea ce raportați este într-adevăr o eroare. Puteți să pierdeți din credibilitate foarte rapid dacă obțineți o reputație de raportare a problemelor care se adevăresc a fi probleme de configurare, greșeli ale utilizatorilor sau inconsistențe ale produsului. Înainte de a descrie problema, asigurați-vă că ați îndeplinit toate condițiile pentru rularea testului.

**Înainte** de a descrie problema luați în considerare:

- Este ceva în setări ce ar putea cauza aceasta? De exemplu, sunt instalate versiunile corecte și se respectă toate dependențele? Ați realizat corect logarea, securitatea, secvența de comenzi ș.a.m.d.?
- S-ar putea ca curățarea incompletă a rezultatelor anterioare, rezultate incomplete sau intervenții de la testarea precedentă să cauzeze aceasta?
- S-ar putea să fie rezultatul unei probleme de rețea sau de mediu?
- Într-adevăr înțelegeți cum aceasta ar trebui să lucreze?

Întotdeauna sunt influențe numeroase ce pot afecta rezultatul testului. Asigurați-vă că înțelegeți ce sunt aceste influențe și considerați rolul lor în eroarea percepută pe care o raportați. Este un domeniu care repede separă un tester experimentat de un începător. **Dacă nu sunteți sigur despre validitatea problemei ar fi înțelept să vă consultați cu un tester experimentat sau un dezvoltator înainte** de a descrie problema.

În esență, dacă aveți dubii, întrebați. Dacă nu întrebați nimeni nu va ști ca nu cunoașteți. Fiți proactivi în autoeducare.

De regulă ajută zicala: „este un păcat să supra-raportezi, dar este o crimă să sub-raportezi”. Nu vă fie frică să scrieți problemele. Depuneți maxim efort să vă asigurați că acestea sunt probleme valide. Când ați descoperit că ați depistat o problemă ce a fost raportată incorect, învățați din aceasta.

### **Neutralitate**

Prezentați problema obiectiv. Nu încercați să utilizați umor sau emoții. Ceea ce credeți că e amuzant când descrieți defectul poate să nu fie interpretat la fel de către dezvoltator care lucrează supratimp și este stresat de linii de cod pe care trebuie să le scrie în plus. Utilizând fraze încărcate cu emoții nu ajută cu nimic la rezolvarea problemei. Frazele emoționale doar creează bariere de comunicare și periclitează lucrul în echipă. Chiar dacă dezvoltatorii au dubii și v-au returnat defectul precedent, dar sunteți convins că aveți dreptate, iar ei au greșit, doar prezentați problema și informația adițională ce va fi de folos dezvoltatorilor. Cu timpul aceasta vă va adăuga profesionalism, respect și credibilitate. Citiți descrierea problemei încă o dată înainte de a o trimite și eliminați sau revizuiți acele comentarii ce pot fi interpretate ca negative de către destinatar.

*Exemplu de neutralitate:*

Acest exemplu este un răspuns al unui programator care a returnat un defect pentru mai multă informație și mai multe detalii despre valorile ce au cauzat problema.

*Raportul defectului:*

**Nu scrieți:**

„Cum putea fi determinat din defectul original cu efort foarte mic, funcția ABC într-adevăr eșuează cu orice valori negative la intrare.”

Aceasta va fi probabil interpretată ca o ofensă de către dezvoltator și nu va adăuga informație utilă.

**Scrieți:** Funcția ABC eșuează cu orice valori negative.

Exemple de câteva valori testate: -1, -36, -32767.

### **Precizie**

Persoana care citește descrierea problemei nu trebuie să fie un detectiv pentru a determina care este problema. Descrieți exact problema percepută. Unele descrieri detaliază o serie de acțiuni și rezultate.

*Exemplu:* „Am apăsat tasta Enter și a avut loc acțiunea A. Apoi am apăsat tasta săgeata înapoi și a avut loc acțiunea B. Apoi am tastat comanda „xyz” și a avut loc acțiunea C”.

*Cititorul poate să nu știe dacă credeți că toate trei acțiuni rezultante au fost incorecte, sau care din ele, dacă vreo una este incorectă.*

În toate cazurile, dar mai ales dacă descrierea este lungă, trebuie să faceți un rezumat al problemelor la începutul descrierii. Nu contează dacă el este prezent și într-un câmp diferit a raportului defectului. Să nu credeți că alții vor face aceleași concluzii pe care le-ați făcut. **Scopul vostru nu este să faceți o descriere ce poate fi înțeleasă, ci o descriere care nu poate fi înțeleasă greșit.** Unica cale să faceți aceasta este să descrieți explicit și precis problema în loc să descrieți doar ce s-a întâmplat.

### Izolare

Fiecare organizație are propria filozofie și așteptări pe cât de mult testerul trebuie să izoleze problema. Deși se înaintează anumite cerințe, testerul trebuie să investească o cantitate rezonabilă de efort pentru izolarea problemei. Considerați următoarele când izolați problemele.

- Încercați să găsiți cel mai scurt, simplu set de pași necesari pentru reproducerea problemei.
- Întrebați-vă dacă ceva din exteriorul codului care e testat a contribuit la apariția problemei.

Spre exemplu, dacă ați avut un blocaj sau o pauză, ar putea să fie din cauza unei probleme de rețea? Sunt careva lucruri ce le-ați putea face pentru a ajuta la îngustarea spectrului componentelor ce au produs eroarea?

- Dacă testul are condiții multiple la intrare, combinați și variați valorile de intrare până o puteți găsi pe cea care cauzează problema.

Abilitatea de a izola problemele, în mare parte, definește valoarea voastră ca tester. Izolarea efectivă salvează pe fiecare implicat de mult timp pierdut. La fel aceasta vă salvează mult timp când trebuie să verificați o eroare corectată.

### Generalizare

Deseori, dezvoltatorii vor corecta exact ce ați raportat, fără chiar să înțeleagă că problema este una mai generală care necesită o atenție mai mare. Spre exemplu, putem raporta că funcția procesorului de texte „salvare fișier” a eșuat și procesorul de texte s-a blocat când am încercat să salvăm fișierul „myfile”. O investigație mai profundă poate scoate în evidență că aceeași eșuare are loc când salvăm un fișier de lungime nulă.

Posibil în această versiune acesta se blochează la fiecare salvare pe un disk îndepărtat, un disk doar citibil ș.a.m.d. Dacă cunoașteți deja aceasta când scrieți raportul, veți salva mult timp pentru dezvoltator și veți facilita posibilitatea corectării pentru prelucrarea cazului general. Când detectați o problemă, parcurgeți pașii rezonabili pentru determinarea dacă ea este mai generală decât cazul evident.

*Exemplu de Generalizare:*

**Nu scrieți:** Mesajul de eroare pentru „fișierul nu a fost găsit” are caractere inadmisibile pentru numele fișierului.

**Scrieți:** Mesajul de eroare pentru „fișierul nu a fost găsit” are caractere inadmisibile pentru numele fișierului. Fiecare mesaj încercat care aștepta date pentru a fi inserate în mesaj avea aceeași problemă. Mesajele fără inserări erau valide.

### Reproducere

Unele erori sunt ușor de reprodus, iar unele – nu. Dacă puteți recrea eroarea trebuie să explicați exact ce trebuie făcut pentru re-creare. Trebuie să descrieți toți pașii, inclusiv sintaxa exactă, numele fișierelor, secvențe ce au fost utilizate pentru a reproduce sau recrea problema. Dacă credeți că problema va avea loc cu orice fișier, orice secvență, etc. menționați aceasta, dar totuși oferiți un exemplu explicit ce poate fi utilizat pentru recreare. Dacă în efortul vostru de a verifica dacă eroarea este re-creabilă găsiți o cale mai scurtă și sigură, documentați cea mai scurtă, ușoară cale de re-creare.

Dacă nu puteți reproduce problema sau dacă credeți că nu puteți reproduce problema, adunați toată informația relevantă, care ar putea oferi informație utilă persoanei care trebuie să încerce să elimine

problema. Nu considerați că aceasta poate fi re-creată dacă nu ați verificat acest fapt. Dacă nu puteți sau nu ați realizat re-crearea problemei, este important să menționați aceasta în raportul defectului.

## Impact

Care este impactul dacă eroarea se produce în mediul clientului? Impactul unor erori este evidentă. Spre exemplu, „întregul sistem cade când apăș tasta Enter”

Unele erori nu sunt atât de evidente. Spre exemplu, puteți descoperi un „typo” în fereastră. Aceasta poate părea foarte minor, chiar trivial dacă nu arătați că de fiecare dată când cineva utilizează produsul acesta este primul lucru pe care îl văd și „typo” rezultă într-un cuvânt inofensiv. În acest caz, chiar dacă este doar un „typo”, aceasta poate fi ceva ce trebuie fixat neapărat înainte de a livra produsul. Dacă credeți că este posibil ca acest defect să nu aibă prioritate înaltă, atunci specificați impactul potențial și reclamați defectul.

Nu supra-reclamați, dar asigurați-vă că cititorii defectului au înțeles clar impactul probabil asupra clientului.

## Depanare

De ce va avea nevoie dezvoltatorul pentru depanarea problemei? Există trace-ing, dump-uri, log-uri ș.a.m.d. ce ar trebui captate și furnizate cu acest raport al defectului? Documentați ce a fost capturat și cum poate fi accesat.

Spre exemplu, când este esențial, oferiți: fișiere, link-uri, manuale etc.

## Evidența

Ce poate demonstra existența erorii? Ați furnizat atât rezultatele așteptate cât și cele obținute? Există documentație care suportă rezultatele așteptate? Odată ce scrieți un raport al problemelor este evident că credeți că într-adevăr există o problemă. Oferiți orice pentru a convinge și pe alții că aceasta este într-adevăr o problemă validă. Evidența poate lua forma documentației de la utilizator, ghid-uri, specificații, cerințe și proiectări. La fel pot fi comentarii precedente ale clienților, standarde de-facto de la produse competitive, sau rezultate de la versiunile precedente ale produsului.

**Nu** credeți că toți văd lucrurile la fel ca dvs. **Nu** așteptați ca oamenii să citească printre rânduri și să facă aceleași concluzii. **Nu** credeți că peste 3 săptămâni vă veți aminti de ce ați crezut că este o eroare. Gândiți-vă ce v-a convins că este o eroare și includeți aceasta în raport. Va trebui să oferiți o evidență chiar mai amplă dacă credeți că există șansa ca situația să nu fie acceptată îndată ca o eroare validă.

## Listă de verificare

Nu veți fi în stare să vă întoarceți și să studiați acest document de fiecare dată când scrieți un raport. Este important să dezvoltați o listă de verificare ușor accesibilă pe care o treceți prin minte de fiecare dată când scrieți un raport al defectelor. Inspecțiile s-au dovedit a fi cele mai puțin costisitoare și cele mai efective metode de îmbunătățire a calității soft-ului. Este important ca să memorați această listă de verificare utilizând orice tehnici de memorie preferate. În majoritatea cazurilor rapoarte neadecvate a defectelor nu sunt datorită inabilității de a scrie un raport bun. De obicei pur și simplu nu ne gândim să răspundem la întrebările corecte.

Puteți găsi util utilizarea mnemonicii pentru lista de verificare. Dacă vă uitați la prima literă a fiecărui item din listă, aceasta devine **CAN PIG RIDE**? Aceasta este destul de relevant pentru a se memora ușor. Dacă zece itemi sunt prea mulți pentru a memora, concentrați-vă la **PIG**. Dacă faceți un lucru bun pe acești trei itemi: Precizare, Izolare și Generalizare, acestea vă vor îndruma spre rapoarte adecvate și mai efective a defectelor în majoritatea cazurilor.

### 3. Model (Template) de raport

Un model de raport al descrierii defectului poate fi destul de util în asigurarea faptului că descrierile oferă informația corectă și răspund la întrebările corecte. Unele unelte de urmărire a defectelor pot permite afișarea unui model.

Un model de raport al descrierii defectului (BugReport):

<b>Data:</b>	11.20.03	<b>OS/Browser</b>	Win XP, SP1
<b>Produs:</b>		<b>Versiunea</b>	4.5.36_1
<b>Exploratoriu?</b>	Y	<b>Script de test?</b>	N
<b>Care</b>			
<b>Numele</b>	T. Popescu		

**Rezumat:**

**Descriere detaliată:**

**Pași prin care se reproduce:**

**Note:**

Informație de depanare suplimentară: (log-uri, referințe, screenshot-uri, etc.)

În raportarea efectivă a defectelor, ca și în multe situații, nu contează atât dacă ai răspuns corect, cât dacă ai răspuns la întrebarea corectă. Aceste zece puncte:

- Condensare
- Acuratețe
- Neutralizare
- Precizie
- Izolare
- Generalizare
- Re-creare
- Impact
- Depanare
- Evidență

oferă o listă de verificare rapidă pentru asigurarea faptului că rapoartele defectelor răspund la întrebările corecte care vor aduce un beneficiu maxim organizației.

### Rezumatul raportului

O linie scurtă de rezumat care e asociată cu majoritatea defectelor este un instrument puternic de comunicare. Deseori, rezumatul este unica porțiune a descrierii defectului care este citită de către cei ce iau decizii. Rezumatul, și nu întreaga descriere, este inclus în rapoarte de testare. Rezumatul este cel la care privesc managerii proiectelor, team liderii și alți manageri când încearcă să înțeleagă defectele asociate cu produsul.

Rezumatul trebuie să fie concis, descriptiv și să transmită un mesaj clar. Rezumatul este de obicei foarte limitat în lungime. Din cauza lipsei de spațiu, abrevierile sunt acceptate și mesajele scurte și clare au prioritate față de o gramatică bună. O bună utilizare a cuvintelor-cheie este esențială odată ce multe

căutări sunt bazate pe abstract. Cuvinte-cheie ca eşuare, suspendare, typo ș.a.m.d. sunt descriptive și convenabile pentru căutare. Dacă spațiul permite este de ajutor să fie indicat mediul, impactul, și una din **cine, ce, când, unde, de ce** întrebări ce pot fi adresate într-un spațiu scurt

## Rezumatul conform listei de verificare

### Obligatoriu:

1. Concis, explicit prezentați problema. (Nu doar că există o problemă).

### Recomandat:

1. Utilizați cuvinte-cheie relevante
2. Prezentați mediul și impactul
3. Răspundeți la întrebările: cine, ce, când, unde, de ce și cum
4. E acceptabil să utilizați abrevieri
5. Gramatică e secundară față de transmiterea mesajului

## Descrierea Detaliată:

Testerii pierd o cantitate semnificativă de timp pentru căutarea și descoperirea problemelor de software. Odată detectate, productivitatea crește dacă defectul este raportat astfel ca problema să fie fixată depunând un efort minim. Asigurarea că este oferită informația necesară este mai importantă ca abilitățile superioare în scris.

Cele 10 puncte descrise în acest document

- Condensare
- Acuratețe
- Neutralizare
- Precizie
- Izolare
- Generalizare
- Re-creare
- Impact
- Depanare
- Evidență

vă vor ajuta mult în oferirea informației corecte în fiecare raport al defectelor.

Nu fiecare citește întregul raport al defectelor. Mulți din cei care iau decizii se bazează pe o linie de Sumar al defectului. Este important de scris abstracte care relevă cu acuratețe mesajul corect.

Cu cât mai bun sunteți în scrierea rapoartelor defectelor și abstractelor, mai mare probabilitatea că problemele vor fi fixate într-un timp mai scurt. Credibilitatea și valoarea voastră pentru business va crește odată ce developerii, managerii și alți testeri vor putea face mai bine lucrul lor datorită rapoartelor defectelor bine scrise și de încredere.

## Anexa A

Condensare (Spune clar, dar pe scurt)

Acuratețe (Este acesta un defect sau o eroare a utilizatorului, neînțelegere, etc. ?)

Neutralizare (Doar fapte. Fără înfrumusețări. Fără umor. Fără emoții.)

**Precizie** (Explicit, care este problema?)

**Izolare** (Ce a fost făcut pentru a izola problema?)

**Generalizare** (Ce a fost făcut pentru a înțelege cât de generală este problema?)

**Re-creare** (Care este esența inițierii/re-creării acestei probleme?)

**Impact** (Care este impactul asupra clientului?)

**Depanare** (De ce are nevoie developer-ul pentru a depana aceasta?)

**Evidență** (Ce documentație va demonstra existența erorii?)

## **Referințe:**

Writing Effective Defect Reports de Kelly Whitmill

IBM Printing Systems Division

6300 Diagonal Highway

003G

Boulder, Colorado 80301

(303) 924-9145