

Tesztelő

- Fő feladata: ügyfél elégedettsége a használt szoftverrel kapcsolatban
 - Az elégedettség természetes, nem vesszük észre
 - Az elégedetlenség azonnal tudatosan jelentkezik

A tesztelés szükségessége

- Szoftver meghibásodása, nem megfelelő működése
- Okai lehetnek:
 - Hardveres hiba
 - Hálózati hiba
 - Konfigurációs hiba
 - Kezelői hiba
 - Nem megfelelő környezet
 - Adatsérülés

A tesztelés szükségessége

- Programhiba
 - Emberi tévedés
 - Ismerethiány
 - Segédprogramok hibái
 - Rohammunka
 - Szervezési hiba

Emberi hibák kiváltó okai

- Szűk határidők
- Fáradtság
- Túlterheltség
- Zavaró környezeti körülmények (zaj, hőmérséklet, stb.)
- Gyakorlatlanság
- Túl bonyolult feladat
- Ellentmondásos célok

Tesztelés fő feladatai

- A fellépő hibák csökkentése
- Megbízhatóság növelése
- Szabványoknak való megfelelés
- Ügyfél elégedettség növelése a cél

Minőség

- Quality
 - Az IEEE szerint a minőség:
az a szint, amikor a komponens, a rendszer vagy folyamat megfelel a meghatározott követelményeknek és/vagy a felhasználó/ügyfél igényeinek, elvárásainak.
- Befolyásoló szoftverjellemzők
 - Funkcionális – funkcionalitás
 - Nem funkcionális – megbízhatóság, használhatóság, hatékonyság, karbantarthatóság, hordozhatóság

Minősegbiztosítás részei

- Szabványok, szabályzatok
- Képzés
- Tesztelés
- Hibaelemzés

Tesztelés mennyisége

- Milyen mértékben kell tesztelni?
 - Fő feladat az optimális érték megtalálása
 - Túl kevés teszt sok hibát hagy benne
 - Túl sok teszt már nem éri meg
- Kockázat (Risk)
 - A jövőbeni negatív következményekkel járó esemény

Definíció

- Tesztelés (testing)
 - Az összes fejlesztési életrőlshoz kapcsolódó akár statikus, akár dinamikus folyamat, amely a szoftvertermékek tervezése, elkészítése, kiértékelése során megállapítja, hogy a szoftver teljesíti-e a meghatározott követelményeket, megfelel-e a célnak.
 - Minden esetben a tesztelés felelős a hibák megtalálásért.

Hibakeresés

- Hibakeresés (Debugging)
 - A szoftver meghibásodás okainak megtalálási, analízálási és eltávolítási folyamata
- A hibakeresés nem tesztelés!!!
- A hibakeresés fejlesztői tevékenység!

A tesztelés folyamata

- Teszttervezés és –irányítás
 - Meg kell határozni
 - a célokat
 - a tevékenységeket
 - a határidőket
 - Aktuális folyamat összehasonlítása a tervvel
 - Eltérések kiértékelése
 - Szükséges módosítások elvégzése

Tesztelemzés és műszaki teszttervezés

- A tesztbázis áttekintése
- Értékelés tesztelhetőség szempontjából
- A tesztfeltételek és azok prioritásának meghatározása
- A tesztesetek megtervezése és prioritásuk meghatározása
- A tesztadatok meghatározása
- A tesztkörnyezet megtervezése
- Nyomonkövethetőség kialakítása

A teszt megvalósítása

- A tesztesetek fejlesztése azok prioritásának meghatározása
- A teszteljárások fejlesztése és prioritásuk beállítása
- Tesztkészletek létrehozása
- Tesztkörnyezet ellenőrzése
- A prioritásnak megfelelően a tesztek végrehajtása
- Dokumentálás
- Az elvárt és a kapott eredmények összehasonlítása
- Eltérések jelentése

Kilépési feltételek, értékelés

- A tesztterv kilépési feltételeinek összehasonlítása a tesztnaplókkal
- Ezek alapján eldöntendő, hogy további tesztelés szükséges-e?
- Tesztösszefoglaló jelentés elkészítése

Teszt lezárása

- A meghatározott átadandó elemek ellenőrzése
- A problémák (issue) jelentések lezárása
- A rendszer átvételéről szóló dokumentum elkészítése
- A teszst infrastruktúra és a hozzá tartozó testware véglegesítése és archiválása
- A testware átadása a karbantartást végzőknek
- Tapasztalatok feldolgozása
- Információk hasznosítása a tesztfolyamat érettségének fejlesztéséhez

Ki teszteljen?

- Független tesztelő
 - Nem ellenérdekelt
 - Tesztelési technikák ismerete
 - Más látásmód, más szemlélet
 - Sok esetben költséghatékonyabb
- A szoftver fejlesztője
 - Ismeri a technológiát
 - Ismeri a kódot
 - Ismeri a követelményeket
 - És amúgy is teszteli saját munkáját

A függetlenség szintjei

- Fejlesztő tesztel
- A fejlesztő csapaton belül van tesztelő
- A tesztelő csapat független, de közös a vezetés
- A felhasználóktól független tesztelők
- Szakosodott tesztelői szakértő
- Külső cég által tesztelve

A jó tesztelő...

- Mi kell a hibák feltárásához?
 - Kíváncsiság
 - Célorientáltság
 - Szakszerű pesszimizmus
 - Kritikus szemlélet
 - A részletekre való aprólékos odafigyelés
 - Jó kommunikációs készség
 - Hatodik érzék – tapasztalat a hibasejtéshez

A jó tesztelő...

- Alapértelmezett tulajdonságok
 - Olvasási készség!!!!
 - Jelentések elkészítésében való jártasság
 - Megfelelő szintű tapasztalat
 - Alkalmazási és üzleti területen
 - Technológiában
 - Tesztelésben

Fontos a kommunikáció

- Fontos a problémák tárgyilagos tálalása
- Vegyük figyelembe a másik fél álláspontját, próbáljuk megérteni
- Mindig ellenőrizzük, hogy sikerült-e egymást megérteni
- A jót és a rosszat is jelezzük vissza, legyen teljes a kommunikációnk

A tesztelés alapelvei

- A tesztelés hibák jelenlétét jelzi: A tesztelés képes felfedni a hibákat, de azt nem, hogy nincs hiba. Növeli a szoftver minőségét és megbízhatóságát.
- Nem lehetséges kimerítő teszt: Minden bemeneti kombinációt nem lehet letesztelni és nem is érdemes. Csak a magas kockázatú és magas prioritású részeket teszteljük.
- Korai teszt: Érdemes a tesztelést az élelciklus minél korábbi szakaszában elkezdeni, mert minél hamar találunk meg egy hibát, annál olcsóbb javítani. Ez azt is jelenti, hogy nemcsak programot, hanem dokumentumokat is lehet tesztelni.
- Hibák csoportosulása: A tesztelésre csak véges időnk van, ezért a tesztelést azokra a modulokra kell koncentrálni, ahol a hibák a legvalószínűbbek, illetve azokra a bemenetekre kell tesztelnünk, amelyekre valószínűleg hibás a szoftver.

A tesztelés alapelvei

- A féregirtó paradoxon: Ha az újratesztelés során mindig ugyanazokat a teszteseteket futtatjuk, akkor egy idő után ezek már nem találnak több hibát. Ezért a tesztjeinket néha bővíteni kell.
- A tesztelés függ a körülményektől: Másképp tesztelünk egy atomerőműnek szánt programot és egy beadandót. Másképp tesztelünk, ha a tesztre 10 napunk vagy csak egy éjszakánk van.
- A hibátlan rendszer téveszméje: Hiába javítjuk ki a hibákat a szoftverben, azzal nem lesz elégedett a megrendelő, ha nem felel meg az igényeinek. Azaz használhatatlan szoftvert nem érdemes tesztelni.

Tesztelési technológiák

- Feketedobozos (black-box) vagy specifikáció alapú
 - A specifikáció alapján készülnek a tesztesetek, a tesztelő nem látja a forráskódot.
 - Van egy lefordított szoftverünk, és egy adott bemenetre tudjuk, milyen kimenetet kellene adni a programnak
- Fehéredobozos (white-box) vagy strukturális teszt – a forráskód alapján készülnek a tesztesetek.
 - Struktúra lefedettsége – a struktúra hány százalékát tudjuk tesztelni a meglévő tesztesetekkel.
 - Tesztelhetők a kódsorok, elágazások, metódusok, osztályok, funkciók, modulok.

Tesztelés szintjei

- Fejlesztői tesztek
 - Komponensteszt
 - A rendszer egy komponensét teszteli
 - Integrációs teszt
 - Kettő vagy több komponens együttműködési tesztje
 - Rendszerteszt
 - Az egész rendszert, azaz minden komponenst együtt tesztel
- Átvételi teszt
 - A felhasználók a kész rendszert tesztelik

Komponens teszt

- A rendszer önálló részeit teszteli általában a forráskód ismeretében
- Legismertebb fajtái
 - unit-teszt – metódusokat teszteli
 - Vizsgálja, hogy a tényleges visszatérési érték megegyezik-e az elvárttal.
 - A unit-tesztnek nem lehet mellékhatása!
 - Modulteszt - a modul nem-funkcionális tulajdonságát teszteli

Integrációs teszt

- A komponensek közti interfészeket, az operációs rendszer és a rendszer közti interfészt, illetve más rendszerek felé nyújtott interfészeket tesztelik
- Az összeillesztés során keletkező hibákat keresi
- Típusai:
 - Komponens – komponens integrációs teszt: A komponensek közötti kölcsönhatások tesztje a komponensteszt után.
 - Rendszer – rendszer integrációs teszt: A rendszer és más rendszerek közötti kölcsönhatásokat tesztje a rendszerteszt után.

Rendszerteszt

- A kész terméket teszteli, hogy megfelel-e a követelményeknek
- Követelmények:
 - a követelmény specifikáció
 - a funkcionális specifikáció
 - a rendszerterv
- Gyakran független cég végzi

Átvételi teszt

- Az egész rendszert teszteli
- A tesztelést a végfelhasználók végzik
- Típusai:
 - alfa teszt - teszt a fejlesztő cégnél
 - béta teszt – végfelhasználók szűk csoportja tesztel
 - felhasználói átvételi teszt – szélesebb béta teszt, majdnem minden felhasználóval
 - üzemeltetői átvételi teszt - rendszergazdák ellenőrzik, hogy a biztonsági funkciók, pl. a biztonsági mentés és a helyreállítás, helyesen működnek-e