

Tesztelési módszerek csoportosítása

Szoftver használata szerint

I. Statikus:

A szoftvermunkatermékek vizsgálata manuálisan vagy adott eszközökkel történik, de nem kerül sor a szoftver futtatására. Nemcsak a szoftverkódra alkalmazható.

A **statikus teszt** tehát egy komponens vagy rendszer tesztelése specifikáció vagy implementáció szinten, a szoftver futtatása nélkül.

A statikus tesztelési technikáknak két fajtája van:

1) **Felülvizsgálat:**

Fehérdozós teszt, mivel kell hozzá a forráskód;

Manuálisan átnézzük a forráskódot és fejben futtatjuk vagy egyszerűen csak gyanús részeket keresünk benne.

2) **Statikus elemzés:**

Fehérdozós teszt, hiszen szükséges hozzá a forráskód;

Szoftverekkel nézetjük át automatikusan a forráskódot.

A következő hiba típusokat könnyebb statikus elemzéssel megtalálni, mint más technikákkal:

- null referenciára hivatkozás,
- tömbök túl vagy alul indexelése,
- nullával való osztás,
- lezáratlan adat folyam (unclosed stream),
- holtpontok (deadlock),
- kiéheztes (starvation).

A statikus tesztelési technikák előnye, hogy nagyon korán alkalmazhatóak, már akkor is, amikor még nincs is futtatható verzió. Így hamarabb lehet velük hibákat találni és így gazdaságosabb a hibajavítás.

A statikus teszteléssel könnyebben megtalálható hibatípusok a dokumentumokban:

- szabványtól való eltérések
- hiányzó követelmények
- műszaki tervezési hibák
- karbantarthatatlan kódspecifikációk
- inkonzisztens interfész-specifikációk

II. Dinamikus:

A szoftvert futásidőben teszteljük.

A dinamikus és statikus tesztelés egymást jól kiegészítő módszerek, mert általában különböző típusú programhibák eredményes és hatékony felfedésére alkalmasak.

Kód ismerete alapján történő tesztelési technikák

I. Fekete doboz (black-box) módszerek:

A specifikáció alapján készülnek a tesztesetek, anélkül, hogy bármit tudnánk a rendszer belső működéséről vagy a forráskódról. A tesztelő a felhasználói interfészen keresztül szolgáltatja a bemeneteket, és vizsgálja a kimeneteket.

- Hatékony és jól alkalmazható nagy kód szegmensben.
- Közepesen képzett tesztelők nagy számban tesztelhetik az alkalmazást, anélkül, hogy ismernék az implementációt, a programozási nyelvet vagy az operációs rendszert.
- A tesztelő csak korlátozott mértékben ismeri az alkalmazást, ezért kevésbé hatékony lehet a tesztelés.
- Korlátozott lefedettség: csak néhány kiválasztott teszt-forgatókönyv kerül végrehajtásra.

II. Fehér doboz (white-box) módszerek:

Strukturális teszt, a forráskód alapján készülnek a tesztesetek. A tesztelőnek meg kell ismernie a forráskódot, meg kell értenie, ezért magasan képzettnek kell lennie.

Általában ezeket a struktúrákat teszteljük:

- kódsorok,
- elágazások,
- metódusok,
- osztályok,
- funkciók,
- modulok.

III. Szürke dobozos (grey-box) tesztelés:

A fekete és a fehér dobozos megközelítés egyes elemei egyszerre kerülnek alkalmazásra. Ebben a megközelítésben a tesztelő ismeri a tesztelt egység bizonyos belső részeit, amit a célzott tesztesetek kidolgozásához felhasznál. A teszt lefuttatása azonban fekete doboz módjára történik, ugyanis semmilyen belső információt nem tartunk nyilván a tesztelt egységről, csak a kimeneteket vizsgáljuk.