1. A tesztelés célja a szükséges lefedettség bztosítása a teszt tárgyának.

2. A tesztelés célja nem változnak.

3. A tesztelés a mnőségellenőrzés egyk formája.

4. Egy programon csak kötelezően ktűzött teszteseteket kell futtatn.

5. Az dőkorlátozás elősegít a roammunka kfejlődését ezzel roncsolva a fő terméket.

6. Leetséges, ogy egy szoftver már tervezésnél s bás.

7. A bajelentések segítk egy rendszer környezetének kalakítását.

8. A mnőséget befolyásolja a ordozatóság.

9. Egy program mnősége jó, a funkconálsan nem tartalmaz bát, de néa belassul.

10. A felasználó élmény kalakítása fontos a mnőség rányítás terén.

11. A tesztelés mennysége növelet, vagy csökkentet a kockázatot.

12. a nncsen látszólagos ba a programban, akkor jól működk.

13. A bafürtök csak egy bzonyos szoftver modulra vannak katásra.

14. A tesztelést érdemes csak a program elkészítése után elkezden.

15. Érdemes teszt esetről, teszt esetre teszteln a programot, ezzel s felgyorsítva a

tesztelést.

16. A tesztelés függ a környezettől. PL: 1 ember v. 100 ember asznál egy rendszert.

17. A tesztelés megelőzet az anyag károkat s.

18. A bákat le kell naplózn.

19. Egy ba kváltó ok megtalálását Dagnózssal tudjuk elvégezn.

1. A szoftver működés problémájánál nem kell fgyelembe venn a ardverkörnyezetet –

2. Az ember félreértés ugyanolyan problémát jelentet, mnt például a ardverba –

3. A tesztelés az ügyfél elégedettségét növel –

4. A mnőség fogalma nem defnálató –

5. A asználatóság a felasználó élményt segít –

6. A tesztelést a leető legkorábban kell kezden –

7. Van, amkor a bák javítása asztalan –

8. Nagyvállalat programok tesztelését neéz szmuláln –

9. Egy weboldal tesztelésénél nem szükséges a különböző böngészők és típusak

eltéréset fgyelembe venn –

10. A asználatóság külön tesztelés területnek számít –

11. Az áttekntetőség fontos a karbantartatóság szempontjából –

12. Környezet változásokra nem muszáj felkészíten a programot –

13. A kockázat negatív jövőöz kötető –

14. Létezk tesztelés előtt bátlan program –

15. szoftvertesztelés nem gényel jó fzka állapotot a résztvevőktől –

16. A rendszer feltételez, ogy az adatok változatnak –

17. A szoftverfejlesztés nem kerül költségekbe –

18. a a teszt nem talál bát, a kód tökéletes –

19. A fejlesztő csapat különböző területen dolgozó emberekből áll –

20. Érvényes konfgurácó mellett s leet problémás a működés –

1. A tesztelés 2 féle leet: dnamkus vagy statkus.

2. A tesztelés célja az s, ogy bzalmat építsen k a tárgy mnőségét lletően.

3. A tesztelés és bakeresés sznte ugyanaz a tevékenység.

4. A skeres tesztelés során mndg mndenre kterjedő tesztelés szükséges.

5. A tesztelésnek különböző környezetekben eltér.

6. A tesztfolyamat a teszttervezéssel kezdődk.

7. A tesztvégreajtás során létrejön a tesztkörnyezet.

8. Egy szoftver megbásodása származat a szoftverből és a szoftver működés

környezetéből.

9. A ardverek problémá általában nncsenek atással a szoftver működésére.

10. Az adatok struktúrájának sérülése nem felsmerető.

11. Egy szoftvernek több segédprogram s része, ezek bája a szoftver működését

befolyásolják.

12. A mnőségrányítás magában foglalja a szoftver mnőségével kapcsolatos alapelveket.

13. A mnőségbztosítás egy komponense a asználatóság.

14. A mnőséget befolyásolja például a megbízatóság, asználatóság és funkconáls

bák.

15. Van olyan funkconáls ba, am mellett még elfogadató mnőségű egy termék.

16. A atékonyság esetén azt vzsgáljuk, ogy egy adott szoftver, ogy gazdálkodk az

erőforrásaval, ennek kvtelezése nagyban függ a fejlesztő tudásától.

17. A tesztelés során fgyelembe kell venn, ogy mlyen célra készült a szoftver, és

mnden leetséges funkcót ugyan úgy kell teszteln.

18. A tesztelés mnél korább fejlesztés fázsban el kell kezden.

19. A bafürtök akár egy másk programrészben s okozat problémát.

20. A féregrtó paradoxon jelentése, ogy az alap tesztkészletre működő program

bamentes.

1. A szoftvertesztelés célja a bák elrejtése a felasználók elől. –

2. A funkconáls bák a szoftver működését érntk. –

3. Csak manuáls tesztelés létezk, az automatzált tesztelés nem szükséges. –

4. Az elfogadás tesztelés a végső tesztelés szakasz a felasználó elfogadás előtt. –

5. A tesztelés dokumentácó nem lényeges része a szoftver mnőségbztosításának. –

6. A szoftverbák mndg a szoftver belső működéséből erednek. –

7. A ardverba befolyásolatja a szoftver működését, és neezítet a bák

felsmerését. –

8. A álózat ba csak specáls álózatok esetén jelentet problémát. –

9. Az ember tévedés az egyk legkomolyabb szoftverba ok leet. –

10. A szoftver megfelelő működéséez a konfgurácós beállításoknak nncs

jelentősége. –

11. A mnőségrányítás célja a bák elkerülése és a folyamatok optmalzálása. –

12. A mnőségbztosítás csak a végtermék ellenőrzésére vonatkozk, nem az egész

fejlesztés folyamatra. –

13. A szoftverasználatóság (usablty) nem befolyásolja a felasználó élményt

(UX). –

14. A kockázatelemzés fontos része a mnőségbztosítás folyamatnak. –

15. A mnőségrányítás során az ügyfél gényet fgyelmen kívül leet agyn, a a

belső szabványok teljesülnek. –

16. Bár a tesztelés kmutatatja a bák jelenlétét, de nem képes igazolni, hogy

nincsenek bák. –

17. A tesztelés a szoftver vagy rendszerfejlesztés életcklusban a legvégén kell

elkezden. –

18. a mndg ugyanazokat a teszteket ajtjuk végre, akkor az azonos a tesztkészlet

egy dő után nem fog új bákat találn. –

19. A tesztelést különböző körülmények esetén különbözőképpen ajtják végre.

Például egy olyan rendszert. –

20. A kadást megelőző tesztelés során a megtalált bák többsége néány modulban

van, vagy legalábbs ezen modulok felelősek a működés bák többségéért. –

1. A tesztelés a bák okozása és megtalálása?

2. A tesztelés és a bakeresés különálló tevékenységek?

3. A tesztelés nem járul ozzá a skerez.

4. A kora tesztelés dő és pénzt spóról.

5. ogya ugyan azokat a teszteket, teszteseteket smételjük akkor atékonyak vagyunk?

6. Az ember tévedés megbásodásoz vezetet.

7. Probléma tud lenn a vállalat szervezés bá?

8. A teszt egyk lényege ogy az ügyfél elégedetségét növelje?

9. A mnőségbztosítás önmagában egy rendszer?

10. A ordozatóság befolyásolja a mnőséget.

11. A nem megfelelő tesztelés a vállalatra jó atást kelt.

12. A tesztelés segít abban ogy a szoftverben ne maradjanak bák.

13. A tesztelés ozzájárul a atékonyságoz.

14. a egy szoftver jól van tesztelve akkor a vállaltnak az késöbb kevesebb pénzbe

kerület.

15. a egy vállalat nem tesztel vagy csak keveset akkor nem vállal kockázatot.

16. A karbantartatóság atékonyságának a növelésében a kommunkácó s szerepet

játszk.

17. a a szoftverünk mnősége nem megfelelő akkor neezen tudjuk a felasználókat

késöbb fejlesztések után rávenn ogy mnőség.

18. Egy projekt során az dő és a nem megfelelő szervezés adja a legtöbb bát!

19. a valamt setetünk, akkor nem lesz jó a vége.

20. Nem fontos ogy legyen átfogó smeretünk a asznált tecnológákban.

1. a tesztlezárásoz kapcsolódó tevékenységeket a projekt mérföldkövek elérésekor ajtjuk

végre. -

2. a tesztelés jellemző célja közé tartozk: bák és a bákat okozó tényezők megkeresése -

3. a tesztelés és a bakeresés ugyan az a tevékenység -

4. az ember bák közé tartozk többek között a: rövd atárdő, nterakcók összetettsége -

5. a szoftver megbásodása csak a működés környezetből származat -

6. A szoftverek mnden esetben valamlyen ardver környezetben működnek, így azok

problémá közvetlen

atással vannak annak működésére -

7. A rendszerek tervezésénél fontos szempont az újrafelasználatóság. -

8. az adatbázs túltereltségéből tpkusan bák számaznak. -

9. egy elkészült rendszerez nem szükséges megatározn a megfelelő felasználó

környezetet. -

10. a tesztelést elég a szoftver/rendszer befejezésekor elkezden -

11. a mnőség önmagában neezen defnálató, mert függ a felasználóktól -

12. ezek a szempontok megatározzák a termék mnőségét: megbízatóság, atékonyság,

karbantartatóság -

13. funkconáls bák esetén megatározunk egy adott mennységet, típust és gyakorságot,

amn belül elfogadató mnőségről beszélünk -

14. egy rendszer akkor megbízató, a a funkcó folyamatosan működnek -

15. a asználatóság szempontjából nem számít a UX (user eXperence) -

16. a atékonyság esetén azt vzsgáljuk, ogy a szoftver ogyan gazdálkodk az

erőforrásokkal (pl. tárely) -

17. a karbantartatóság szempontjából, csak a felasználó környezetet kell fgyelembe venn

-

18. a tcket rendszert a bák számontartására asználják -

19. a ordozatóság szempontjából úgy kell felépíten a rendszert, ogy a környezet

változására megfelelően reagáljon -

20. a kockázat az a tényező, am a jövőben negatív következményeket okozat –

-

1. A tesztelés egyk fő célja, ogy a megbásodások okát, és a bákat megtaláljuk

2. A tesztelésnek nem célja, ogy csökkentsük a szoftvermnőség kockázat szntjét.

3. Azért tesztelünk, ogy a bák jelenlétét mutassuk k, nem a ányát.

4. A bakeresés folyamat első lépése az ok kjavítása.

5. A QA mnőség bztosítást/tesztelést jelent.

6. A kmerítő teszt leetséges.

7. A kora tesztelés dőt és pénzt s tud spóroln.

8. A tesztelés mnden környezetben ugyan úgy történk.

9. A teszt tevékenységeket páruzamosan s leet végezn, nem kötelező a sorrend.

10. Megbásodás csak és kzárólag a szoftver működéséből eredet.

11. A ardver környezet közvetlen atással van a program működésére.

12. A rendszerek tervezésénél nem számít az újrafelasználatóság.

13. A szoftver megbásodás eredet elkészítés során elkövetett bákból.

14. Az ember tévedés nem vezetet bára.

15. Azért tesztelünk, ogy a bák gyakorságát csökkentsük, ne megszűntessük.

16. A funkconáls bák nem befolyásolják a mnőséget.

17. A mnőség „Az a sznt, amkor egy komponens, rendszer vagy folyamat megfelel a

megatározott követelményeknek és/vagy a felasználó/ügyfél elvárásanak”

18. A atékonyság nem befolyásolja a mnőséget.

19. Fontos a mnőségnek a karbantartatóság.

20. A tesztesetek rendszeres felül vzsgálása felesleges.

1. A elytelenül működő szoftver sok veszteségez vezetet.

2. A szoftvertesztelés segít a szoftver működés közben megbásodás kockázatának

csökkentésében.

3. A tesztelés csak statkus.

4. A rendszerek bája az ember tévedése matt s származat.

5. Nem fontos, ogy a csapat számára egyértelmű legyen a cél.

6. bák gyakorsága attól függ, ogy a bás funkcó mlyen gyakran van asználatban.

7. A asználatóság angol megfelelője a usablty.

8. atékonyság esetén azt vzsgáljuk, ogy egy adott szoftver ogyan gazdálkodk az

erőforrásokkal.

9. Karbantartatóság szempontjából, olyan megoldásokat kell választan, am azt

bztosítja, ogy a rendszer üzemeltetése során, könnyen áttekntető legyen.

10. A kockázat nem okozat poztív következményket.

11. A tesztelés költségatékony módszert kínál a bák észlelésére.

12. Ember bákat ember lények követnek el.

13. A mínőség sznt egy mértékrendszer.

14. A tesztelés feladata ogy növelje az ügyfél elégedettségét.

15. A tesztelés kmutatatja a bák jelenlétét, de azt nem képes oln, ogy nncsenek

bák.

16. A tesztelést elég elkezden a projekt befejezésénél.

17. A rendszereknél tervezésénel fontos szempont az újra felasználatóság.

18. A tesztelés és a bakeresés ugyan az a tevékenység.

19. A álózat probléma s okozatja a megbásodást.

20. A szoftverek számos komponensből épülnek fel.

1. A szoftvertesztelés a szoftvermnőségét értékel, lletve segít a szoftver működés

közben megbásodás kockázatának csökkentésében. –

2. A tesztelés fő célja többek között, bzalom képítése a teszt tárgyának mnőségével

kapcsolatban –

3. A teszttárgy tesztelése során kzárólag a szoftver tesztelésére kell odafgyeln a

teszttárgy bára nem. –

4. Egy ba kjavítása nem okozat tovább bákat. –

5. A tesztelők bztosítják, ogy a felasználók gényet a fejlesztés életcklus során

fgyelembe vegyék. –

6. A tesztelés és a mnőségbztosítás sznonma szavak és ugyan azt s jelentk. –

7. A mnőségbztosítás mnd a fejlesztés, mnd a tesztelés folyamatra értelmezető és a

projektekben mndenk felelős. –

8. A tesztelés a bák jelenlétét mutatja k nem azok ányát. –

9. A teszteket elég csak a projekt befejeztével elvégezn szen így több dő és pénz

spórolató meg. –

10. A bafürtök megjelenése kzárólag egy szektorban okoznak bát. –

11. A tesztelés során a teszt adatokat módosítan kell aoz, ogy a tesztek atékonyak

legyenek. –

12. A szoftver megbásodások oka származat a szoftver működés környezetéből és

magából a szoftverből. –

13. ardver ba leet például egy olyan érzékelő megbásodása s, amely adatokkal lát

el összetettebb szoftvereket. Például, a egy őfok vagy nyomás érzékelő

adatokat ad, akkor a rendszer szoftver a adatok alapján vezérl az eszközt. -

14. a az adatok áramlása az egyes számítógépek között nem működk megfelelően akkor

az csak mnmálsan lassítja annak működését. –

15. A szoftverek megbásodása származat a külső körülményeken túl olyan okokból,

amely az elkészítéskor elkövetett bákra vezetetők vssza. –

16. A szoftverek számos komponensből épülnek fel, amelyek külön-külön való működése

valósítja meg szoftver célját. -

17. A szoftverfejlesztés és tesztelés gen komoly szellem tevékenység, am jó szellem és

fzka állapotot kíván meg a projekt résztvevőtől. –

18. A mnőség rányítás fgyelmen kívül agyja a mnőséggel kapcsolatos alapelveket. –

19. A asználatóság célja, ogy a szoftver mnél gyorsabban futasson így a felasználó

élményt mellőzve. –

20. A bák megtalálása és javítsa asztalan, a a kfejlesztett rendszer asználatatlan, és

nem felel meg a felasználók gényenek, elvárásanak. –

1. A tesztelés az összes bát feltárja. –

2. A tesztelés különböző környezetekben eltérő módon történk. –

3. A ardver problémá nncsenek atással a szoftver működésére. –

4. A asználatóság célja, ogy a felasználó élmény mnél jobb legyen. –

5. A ordozatóság szempontjából fontos, ogy a rendszer ne reagáljon a környezet

változásara. –

6. A tesztelés célja az ügyfél elégedettségét növeln. –

7. A kockázat az a tényező, amely a jövőben poztív következményeket okozat. -

8. A rendszer megbízatatlan, a folyamatosan leáll, belassul és így a funkcók nem

asználatóak. -

9. Az ember bát nem kell számításba venn teszteléskor -

10. a a rendszer nncs felkészítve adatsérülésre akkor az nem megfelelő működésez

vezet. -

11. Az ember tévedés nem gyakor ba ok. -

12. A tesztlezárásoz kapcsolódó tevékenységeket projektmérföldkövek elérésekor ajtjuk

végre. -

13. A teszttervezés részét nem képzk a tesztcélok megatározása. -

14. A tesztmegvalósítás során létreozzuk vagy beszerezzük a tesztvégreajtásoz

szükséges tesztvert. -

15. A tesztelést elszgetelten végzk. -

16. A tesztelés költségatékony módszert kínál a bák észlelésére. -

17. A mnőségbztosítás önmagában s egy rendszer. –

18. A atás számítása nem becsülető tapasztalt úton. -

19. A tesztelés és a bakeresés különálló tevékenységek. -

20. A tesztelés képes oln, ogy nncsenek bák. –

1. A tesztelés és a bakeresés nem különálló tevékenység.

2. A tesztelés egy költségatékony módszer a bák észlelésére.

3. A tesztelők bztosítják, ogy a felasználók génye a fejlesztés

4. során fgyelembe legyenek véve.

5. A tesztelés a mnőségellenőrzés egyk formája.

6. Ugyanazon tesztek lefuttatása számos alkalommal nem ront a

7. atékonyságán új bák megtalálásában.

8. A szoftver ardverkörnyezete nncs atással annak működésére.

9. Az egyk legkomolyabb ba ok az ember tévedés.

10. A segédprogramok bája a szoftver működését s befolyásolja.

11. A rendszereknek számos különböző környezetben kell működne

12. A szoftver tervezésekor nncs megatározva, ogy mlyen nput

13. és output adatokkal működk megfelelően.

14. A tesztelés szerepe közé tartozk a szoftver megbízatóságának

15. növelése.

16. A szoftver asználatósága nem befolyásolja a mnőséget.

17. A asználatóság célja, ogy a felasználó élmény

18. mnél jobb legyen.

19. Karbantartás szempontjából fontos az áttekntetőség.

20. A tesztelést különböző körülmények között különbözőképpen

21. ajtják végre.

22. Felesleges a leető legkorábban elkezden a tesztelést.

23. a a tesztelés során nem találnak bát akkor a

24. rendszer tökéletes.

25. Fontos a csapat számára egyértelmű célok és feladatok.

26. A tesztelés leet statkus vagy dnamkus.

1. A statkus tesztelésnél a szoftvert futtatn kell?

2. A tesztelés garantálja, ogy a szoftver soa nem fog megbásodn?

3. A tesztelés nem csak a szoftvertesztelők feladata?

4. A szoftvertesztelés során csak a szoftver működését kell ellenőrzn.

5. A szoftverbák nem mndg a szoftver bájából adódnak.

6. Az adatsérülés mndg a rendszer teljes összeomlásáoz vezet.

7. Egy érzékelő bája okozat nagyobb problémát egy összetett rendszerben.

8. A ardverbák csak az új eszközöknél fordulatnak elő.

9. A álózat bák csak a szoftverek működését befolyásolják.

10. A fejlesztőknek jó fzka és mentáls állapotban kell lennük a atékony munkáoz.

11. A felasználó szempontjából a mnőség szubjektív.

12. A szoftver mnőségét csak a funkconáls bák befolyásolják.

13. A tesztelést különböző körülmények esetén s ugyanúgy ajtják végre.

14. A tesztelést elég akkor elkezden, amkor kész a program.

15. Nem létezk tökéletes program.

16. A felasználó élmény köré egy külön terület nőtte k magát.

17. Az smeretány leet tecnológa, fejlesztéstecnka, módszertan vagy üzlet

smeretány.

18. dőnként a tesztek elavulnak és elyettük újakat kell írn.

19. Egy szoftver ba akár mllós kárt okozat.

20. A projekt tervezése során oda kell fgyeln, ogy megfelelő körülmények bztosítottak

legyenek a csapat számára.

1. a ugyanazokat a teszteket smételjük akkor azok nem lesznek atékonyak új bák

megtalálásában.

2. A tesztelés mnden környezetben ugyanolyan módon történk.

3. A teszttervezés része a tesztcélok megatározása és egy erre alkalmas megközelítés

kválasztása.

4. Lneárs tesztelésnek nevezzük amkor közvetlenül a teszt tárgyában találat bákat.

5. Dnamkus tesztelésnél a szoftver bát tárjuk fel.

6. A szoftver megbásodások oka származat a szoftver működés környezetéből és

magából a szoftverből.

7. A ardver ba soa nem vezetet teljes működésképtelenségez.

8. álózat bákat nagyon egyszerű feltárn egy tesztelőnek.

9. A rendszerek tervezésénél fontos szempont az újrafelasználatóság.

10. A szoftverek megbásodása származat a külső körülményeken túl olyan okokból,

amely az elkészítéskor elkövetett bákra vezetetők vssza.

11. Van olyan ogy az alkalmazás nem működk megfelelően más típusú böngészővel.

12. A szoftverfejlesztés nem gényel jó szellem állapotot.

13. Tesztelés feladata a bák csökkentése, megbízatóság növelése és az előírásoknak

való megfelelés bztosítása.

14. Funkconáls bák esetén megatározzuk azt a mennységet, típust és gyakorságot,

am mellett meg elfogadató a mnősége.

15. A asználatóság célja, ogy a fejlesztőnek egyszerűbb legyen tesztelne.

16. A tesztelés mennységénél fgyelembe kell venn a rendszer célját.

17. A legtöbb dőveszteséget egy projekt alatt az smeretek ánya, a nem megfelelő

szervezés és az dőnyomás generálja.

18. A tesztelés képes oln ogy nncs ba a szoftverbe.

19. A tesztelést a leető legkorábban el kell kezden, és előre megatározott célokra kell

összpontosítan.

20. A féregírtó paradoxon ellen a teszteseteket rendszeresen felül kell vzsgáln.

1. A tesztelés kmutatja a bák jelenlétét.

2. Létezk bamentes rendszer.

3. A különböző rendszereket különbözőképpen kell teszteln.

4. A tesztelést a szoftver legkorább fázsában kell elkezden.

5. Gyakran kell cseréln a teszteseteket.

6. Nem számít tesztcélnak a bák megtalálása.

7. Nem leet ba nélkül szoftvert írn.

8. A tesztelés és a bakeresés különálló tevékenységek.

9. A statkus tesztelés közvetlenül találja meg a bákat.

10. A tesztelés egy drága módszer a bák észlelésére.

11. Sokszor ardver ba okozza a szoftver nem megfelelően működését.

12. A rendszerek tervezésénél nem fontos szempont az újraasználatóság.

13. Felasználó nem okozat bát.

14. Adatsérülés nem okozat bát.

15. a nem olyan környezetben van a szoftver, aova tervezték, akkor nem garantálató a

működése.

16. A tesztelések csökkentk a szoftver megbízatóságát.

17. A mnőség: az a sznt, amkor egy komponens, rendszer vagy folyamat megfelel a

megatározott követelményeknek és a felasználó elvárásanak.

18. A usablty célja, ogy a felasználó élmény mnél jobb legyen.

19. A szoftverekben nem fontos a karbantartatóság és a atékonyság.

20. A tesztelés mennységének megállapítása során nem fontos fgyelembe venn, ogy a

szoftver mlyen célra készült.