

Praktické cvičenia: Identifikácia typu testovania

1. **Situácia:** Pri zadávaní emailovej adresy do formulára je zobrazené chybové hlásenie, ak je email neplatný (napr. bez „@“ alebo „.com“).
 - **Typ testovania: Testovanie funkčnosti formulárov (validácia vstupu)**
2. **Situácia:** Po zadaní nesprávneho používateľského mena a hesla sa zobrazí chybové hlásenie a používateľ nie je prihlásený.
 - **Typ testovania: Testovanie autentifikácie (funkčnosť prihlásenia)**
3. **Situácia:** Testovanie webovej stránky na rôznych zariadeniach (mobil, tablet, desktop) a kontrola, či sa stránka správne zobrazuje na každom z nich.
 - **Typ testovania: Testovanie responzivity**
4. **Situácia:** Po odoslaní objednávky cez nákupný košík systém správne vypočíta celkovú cenu a aplikuje zľavu.
 - **Typ testovania: Testovanie obchodných procesov (nákupný proces)**
5. **Situácia:** Používateľ klikne na neexistujúci odkaz a zobrazí sa stránka 404.
 - **Typ testovania: Testovanie chybových stránok**
6. **Situácia:** Používateľ vyplní prihlásenie, ale počas procesu zistí, že zabudol heslo a klikne na odkaz na obnovenie hesla.
 - **Typ testovania: Testovanie funkčnosti zabudnutého hesla**
7. **Situácia:** Otestovanie, či systém správne reaguje na pokus o SQL injection v prihlasovacom formulári.
 - **Typ testovania: Bezpečnostné testovanie (SQL injection)**
8. **Situácia:** Počas testovania na rôznych prehliadačoch (Chrome, Firefox, Safari) sa stránky načítajú správne bez vizuálnych a funkčných problémov.
 - **Typ testovania: Testovanie kompatibility prehliadačov**
9. **Situácia:** Po odoslaní formulára na stránke sa zobrazuje správne chybové hlásenie, ak niektoré polia zostanú prázdne.
 - **Typ testovania: Testovanie validácie formulára**
10. **Situácia:** Na stránke sa testuje, či sa obsah (obrázky, texty) zobrazuje správne a či sa stránka načíta rýchlo aj pri vyššom počte súčasných používateľov.
 - **Typ testovania: Výkonové testovanie**

Riešenia

1. **Situácia:** Pri zadávaní emailovej adresy do formulára je zobrazené chybové hlásenie, ak je email neplatný (napr. bez „@“ alebo „.com“).

- **Typ testovania podľa úrovne:** Testovanie jednotiek (Unit testing)
- **Typ testovania podľa prístupu:** Šedá box metóda (Gray-box)
- **Typ testovania podľa cieľa:** Testovanie funkčnosti
- **Typ testovania podľa automatizácie:** Automatizované testovanie
- **Typ testovania podľa typu softvéru:** Testovanie webovej aplikácie
- **Testovaná vlastnosť:** Validácia vstupov (formuláre)
- **Prístup overovania systému:** Overenie výstupov

2. **Situácia:** Po zadaní nesprávneho používateľského mena a hesla sa zobrazí chybové hlásenie a používateľ nie je prihlásený.

- **Typ testovania podľa úrovne:** Testovanie integrácie
- **Typ testovania podľa prístupu:** Čierna box metóda (Black-box)
- **Typ testovania podľa cieľa:** Testovanie funkčnosti
- **Typ testovania podľa automatizácie:** Automatizované testovanie
- **Typ testovania podľa typu softvéru:** Testovanie webovej aplikácie
- **Testovaná vlastnosť:** Autentifikácia
- **Prístup overovania systému:** Overenie výstupov

3. **Situácia:** Testovanie webovej stránky na rôznych zariadeniach (mobil, tablet, desktop) a kontrola, či sa stránka správne zobrazuje na každom z nich.

- **Typ testovania podľa úrovne:** Testovanie systému
- **Typ testovania podľa prístupu:** Biela box metóda (White-box)
- **Typ testovania podľa cieľa:** Testovanie použiteľnosti
- **Typ testovania podľa automatizácie:** Ručné testovanie
- **Typ testovania podľa typu softvéru:** Testovanie webovej aplikácie
- **Testovaná vlastnosť:** Responzivita
- **Prístup overovania systému:** Overenie správania systému

4. **Situácia:** Po odoslaní objednávky cez nákupný košík systém správne vypočíta celkovú cenu a aplikuje zľavu.

- **Typ testovania podľa úrovne:** Testovanie systému
- **Typ testovania podľa prístupu:** Šedá box metóda (Gray-box)

- **Typ testovania podľa cieľa:** Testovanie obchodných procesov
- **Typ testovania podľa automatizácie:** Automatizované testovanie
- **Typ testovania podľa typu softvéru:** Testovanie webovej aplikácie
- **Testovaná vlastnosť:** Správnosť výpočtov
- **Prístup overovania systému:** Overenie výstupov

5. **Situácia:** Používateľ klikne na neexistujúci odkaz a zobrazí sa stránka 404 „Stránka neexistuje“.

- **Typ testovania podľa úrovne:** Testovanie systému
- **Typ testovania podľa prístupu:** Čierna box metóda (Black-box)
- **Typ testovania podľa cieľa:** Testovanie správneho chovania systému
- **Typ testovania podľa automatizácie:** Ručné testovanie
- **Typ testovania podľa typu softvéru:** Testovanie webovej aplikácie
- **Testovaná vlastnosť:** Chybové stránky
- **Prístup overovania systému:** Overenie výstupov

6. **Situácia:** Používateľ vyplní prihlásenie, ale počas procesu zistí, že zabudol heslo a klikne na odkaz na obnovenie hesla.

- **Typ testovania podľa úrovne:** Testovanie integrácie
- **Typ testovania podľa prístupu:** Šedá box metóda (Gray-box)
- **Typ testovania podľa cieľa:** Testovanie funkčnosti
- **Typ testovania podľa automatizácie:** Automatizované testovanie
- **Typ testovania podľa typu softvéru:** Testovanie webovej aplikácie
- **Testovaná vlastnosť:** Obnova hesla
- **Prístup overovania systému:** Overenie správania systému

7. **Situácia:** Otestovanie, či systém správne reaguje na pokus o SQL injection v prihlasovacom formulári.

- **Typ testovania podľa úrovne:** Testovanie bezpečnosti
- **Typ testovania podľa prístupu:** Čierna box metóda (Black-box)
- **Typ testovania podľa cieľa:** Testovanie bezpečnosti
- **Typ testovania podľa automatizácie:** Automatizované testovanie
- **Typ testovania podľa typu softvéru:** Testovanie webovej aplikácie
- **Testovaná vlastnosť:** Ochrana proti SQL injection
- **Prístup overovania systému:** Overenie výstupov

8. **Situácia:** Počas testovania na rôznych prehliadačoch (Chrome, Firefox, Safari) sa stránky načítajú správne bez vizuálnych a funkčných problémov.

- **Typ testovania podľa úrovne:** Testovanie systému
- **Typ testovania podľa prístupu:** Biela box metóda (White-box)
- **Typ testovania podľa cieľa:** Testovanie kompatibility prehliadačov
- **Typ testovania podľa automatizácie:** Ručné testovanie
- **Typ testovania podľa typu softvéru:** Testovanie webovej aplikácie
- **Testovaná vlastnosť:** Kompatibilita prehliadačov
- **Prístup overovania systému:** Overenie správania systému

9. **Situácia:** Po odoslaní formulára na stránke sa zobrazuje správne chybové hlásenie, ak niektoré polia zostanú prázdne.

- **Typ testovania podľa úrovne:** Testovanie jednotiek (Unit testing)
- **Typ testovania podľa prístupu:** Biela box metóda (White-box)
- **Typ testovania podľa cieľa:** Testovanie funkčnosti
- **Typ testovania podľa automatizácie:** Automatizované testovanie
- **Typ testovania podľa typu softvéru:** Testovanie webovej aplikácie
- **Testovaná vlastnosť:** Validácia vstupov (formuláre)
- **Prístup overovania systému:** Overenie výstupov

10. **Situácia:** Na stránke sa testuje, či sa obsah (obrázky, texty) zobrazuje správne a či sa stránka načíta rýchlo aj pri vyššom počte súčasných používateľov.

- **Typ testovania podľa úrovne:** Testovanie systému
- **Typ testovania podľa prístupu:** Biela box metóda (White-box)
- **Typ testovania podľa cieľa:** Testovanie výkonu
- **Typ testovania podľa automatizácie:** Automatizované testovanie
- **Typ testovania podľa typu softvéru:** Testovanie webovej aplikácie
- **Testovaná vlastnosť:** Výkon a rýchlosť načítania
- **Prístup overovania systému:** Overenie správania systému

Praktické cvičenie: Identifikácia typu testovania 2

11. Testovanie rýchlosti načítania hlavnej stránky webovej aplikácie s použitím nástroja ako Google PageSpeed Insights., texty) zobrazuje správne a či sa stránka načíta rýchlo aj pri vyššom počte súčasných používateľov
12. Testovanie registračného formuláru, kde sa používateľ pokúša zadať používateľské meno, ktoré už existuje v databáze.
13. Testovanie zabezpečenia webovej stránky pred XSS (Cross-Site Scripting) útokmi. Otestuje sa, či je vstup používateľa správne ošetrovaný, aby nebolo možné vložiť škodlivý JavaScript kód.
14. Po kliknutí na tlačidlo "Zabudol/a som heslo" sa používateľovi odošle email s odkazom na obnovenie hesla.
15. Otestovanie, či aplikácia správne vypočíta zľavu pri nákupe cez mobilnú aplikáciu.
16. Po nahraní obrázka na profil používateľa systém správne skaluje obrázok podľa požiadaviek.
17. Testovanie systému pri vysokom počte používateľov, ktorí sa prihlásia do aplikácie súčasne, aby sa overil výkon servera.
18. Otestovanie, či aplikácia správne reaguje na neplatný formát dátumu, ktorý používateľ zadá do formuláru.
19. Otestovanie zobrazenia a fungovania navigačného menu v rôznych jazykoch (napr. angličtina, slovenčina) na webovej stránke.
20. Otestovanie pripojenia na externý API systém, ktorý poskytuje údaje pre aplikáciu (napr. počasie) a kontrola, či API správne reaguje.

Riešenia 2

Situácie:

11. **Situácia:** Testovanie rýchlosti načítania hlavnej stránky webovej aplikácie s použitím nástroja ako Google PageSpeed Insights.
 - **Typ testovania podľa úrovne:** Testovanie systému
 - **Typ testovania podľa prístupu:** Biela box metóda (White-box)
 - **Typ testovania podľa cieľa:** Testovanie výkonu
 - **Typ testovania podľa automatizácie:** Automatizované testovanie
 - **Typ testovania podľa typu softvéru:** Testovanie webovej aplikácie
 - **Testovaná vlastnosť:** Výkon načítania
 - **Prístup overovania systému:** Overenie správania systému
12. **Situácia:** Testovanie registračného formuláru, kde sa používateľ pokúša zadať používateľské meno, ktoré už existuje v databáze.
 - **Typ testovania podľa úrovne:** Testovanie integrácie
 - **Typ testovania podľa prístupu:** Čierna box metóda (Black-box)
 - **Typ testovania podľa cieľa:** Testovanie funkčnosti
 - **Typ testovania podľa automatizácie:** Ručné testovanie
 - **Typ testovania podľa typu softvéru:** Testovanie webovej aplikácie
 - **Testovaná vlastnosť:** Správnosť registrovanej funkcie
 - **Prístup overovania systému:** Overenie výstupov
13. **Situácia:** Testovanie zabezpečenia webovej stránky pred XSS (Cross-Site Scripting) útokmi. Otestuje sa, či je vstup používateľa správne ošetrovaný, aby nebolo možné vložiť škodlivý JavaScript kód.
 - **Typ testovania podľa úrovne:** Testovanie bezpečnosti
 - **Typ testovania podľa prístupu:** Čierna box metóda (Black-box)
 - **Typ testovania podľa cieľa:** Testovanie bezpečnosti
 - **Typ testovania podľa automatizácie:** Automatizované testovanie
 - **Typ testovania podľa typu softvéru:** Testovanie webovej aplikácie
 - **Testovaná vlastnosť:** Bezpečnostné zraniteľnosti
 - **Prístup overovania systému:** Overenie správania systému
14. **Situácia:** Po kliknutí na tlačidlo "Zabudol/a som heslo" sa používateľovi odošle email s odkazom na obnovenie hesla.

- **Typ testovania podľa úrovne: Testovanie integrácie**
- **Typ testovania podľa prístupu: Šedá box metóda (Gray-box)**
- **Typ testovania podľa cieľa: Testovanie funkčnosti**
- **Typ testovania podľa automatizácie: Automatizované testovanie**
- **Typ testovania podľa typu softvéru: Testovanie webovej aplikácie**
- **Testovaná vlastnosť: Obnova hesla**
- **Prístup overovania systému: Overenie výstupov**

15. **Situácia:** Otestovanie, či aplikácia správne vypočíta zľavu pri nákupe cez mobilnú aplikáciu.

- **Typ testovania podľa úrovne: Testovanie systému**
- **Typ testovania podľa prístupu: Biela box metóda (White-box)**
- **Typ testovania podľa cieľa: Testovanie funkčnosti**
- **Typ testovania podľa automatizácie: Automatizované testovanie**
- **Typ testovania podľa typu softvéru: Testovanie mobilnej aplikácie**
- **Testovaná vlastnosť: Správnosť výpočtov**
- **Prístup overovania systému: Overenie výstupov**

16. **Situácia:** Po nahraní obrázka na profil používateľa systém správne skaluje obrázok podľa požiadaviek.

- **Typ testovania podľa úrovne: Testovanie jednotiek (Unit testing)**
- **Typ testovania podľa prístupu: Biela box metóda (White-box)**
- **Typ testovania podľa cieľa: Testovanie funkčnosti**
- **Typ testovania podľa automatizácie: Automatizované testovanie**
- **Typ testovania podľa typu softvéru: Testovanie webovej aplikácie**
- **Testovaná vlastnosť: Správnosť spracovania obrázkov**
- **Prístup overovania systému: Overenie výstupov**

17. **Situácia:** Testovanie systému pri vysokom počte používateľov, ktorí sa prihlásia do aplikácie súčasne, aby sa overil výkon servera.

- **Typ testovania podľa úrovne: Testovanie výkonu**
- **Typ testovania podľa prístupu: Biela box metóda (White-box)**
- **Typ testovania podľa cieľa: Testovanie zaťaženia**
- **Typ testovania podľa automatizácie: Automatizované testovanie**
- **Typ testovania podľa typu softvéru: Testovanie serverovej aplikácie**

- **Testovaná vlastnosť: Výkon pri zaťažení**
- **Prístup overovania systému: Overenie správania systému**

18. **Situácia:** Otestovanie, či aplikácia správne reaguje na neplatný formát dátumu, ktorý používateľ zadá do formuláru.

- **Typ testovania podľa úrovne: Testovanie jednotiek (Unit testing)**
- **Typ testovania podľa prístupu: Biela box metóda (White-box)**
- **Typ testovania podľa cieľa: Testovanie funkčnosti**
- **Typ testovania podľa automatizácie: Automatizované testovanie**
- **Typ testovania podľa typu softvéru: Testovanie webovej aplikácie**
- **Testovaná vlastnosť: Validácia dátumu**
- **Prístup overovania systému: Overenie výstupov**

19. **Situácia:** Otestovanie zobrazenia a fungovania navigačného menu v rôznych jazykoch (napr. angličtina, slovenčina) na webovej stránke.

- **Typ testovania podľa úrovne: Testovanie systému**
- **Typ testovania podľa prístupu: Biela box metóda (White-box)**
- **Typ testovania podľa cieľa: Testovanie lokalizácie**
- **Typ testovania podľa automatizácie: Ručné testovanie**
- **Typ testovania podľa typu softvéru: Testovanie webovej aplikácie**
- **Testovaná vlastnosť: Lokalizácia a jazyková podpora**
- **Prístup overovania systému: Overenie správania systému**

20. **Situácia:** Otestovanie pripojenia na externý API systém, ktorý poskytuje údaje pre aplikáciu (napr. počasie) a kontrola, či API správne reaguje.

- **Typ testovania podľa úrovne: Testovanie integrácie**
- **Typ testovania podľa prístupu: Šedá box metóda (Gray-box)**
- **Typ testovania podľa cieľa: Testovanie funkčnosti**
- **Typ testovania podľa automatizácie: Automatizované testovanie**
- **Typ testovania podľa typu softvéru: Testovanie webovej aplikácie**
- **Testovaná vlastnosť: Integrácia s externými systémami**
- **Prístup overovania systému: Overenie správania systému**

Identifikácia vhodnosti exploratívneho testovania

Scenár 1: Aplikácia na správu osobných financií umožňuje používateľom pridať, upraviť a odstrániť výdavky. Po pridaní výdavku sa automaticky aktualizuje zostatok na účte a zobrazí sa upozornenie o prebytku alebo nedostatku financií.

Scenár 2: Aplikácia pre spravovanie zoznamu úloh umožňuje používateľom pridávať, upravovať, označovať ako dokončené a odstraňovať úlohy. Každá úloha má názov, popis a dátum splnenia.

Scenár 3: Aplikácia pre e-commerce (eshop) umožňuje zákazníkom pridať produkty do nákupného košíka, prejsť k pokladni, vybrať spôsob platby a dokončiť nákup.

Scenár 4: Aplikácia na spracovanie platieb zabezpečuje, že každý platobný príkaz je spracovaný podľa stanovených pravidiel a že údaje o platbách sú správne zaznamenané v databáze.

Scenár 5: Mobilná aplikácia na sledovanie športových výkonov zaznamenáva aktivitu používateľov, analyzuje ich výkony a poskytuje odporúčania na zlepšenie.

Je exploratívne testovanie vhodné pre tento scenár?

Ak nie, aký typ testovania je vhodný a prečo?

Riešenia

Scenár 1: Aplikácia na správu osobných financií umožňuje používateľom pridať, upraviť a odstrániť výdavky. Po pridaní výdavku sa automaticky aktualizuje zostatok na účte a zobrazí sa upozornenie o prebytku alebo nedostatku financií.

* **Exploratívne testovanie je vhodné.**

Tento scenár obsahuje funkčnosť, ktorá môže byť závislá na rôznych vstupoch a interakciách medzi časťami aplikácie (napr. pridať výdavky, upraviť zostatok, kontrola upozornení). Testeri môžu interagovať s aplikáciou, skúšať neštandardné hodnoty výdavkov alebo rýchlo meniť vstupy na overenie stability a správnosti aplikácie.

* **Typ testovania:**

Exploratívne testovanie je vhodné, pretože umožňuje testovať nepredvídateľné scenáre a zistiť, ako aplikácia reaguje na neočakávané vstupy.

Scenár 2:

Aplikácia pre spravovanie zoznamu úloh umožňuje používateľom pridávať, upravovať, označovať ako dokončené a odstraňovať úlohy. Každá úloha má názov, popis a dátum splnenia.

* **Exploratívne testovanie nie je najvhodnejšie.**

Tento scenár je pomerne priamočiary a predvídateľný. Testovanie môže byť vykonané pomocou **funkčného testovania**, ktoré overí správnosť všetkých funkcií v súlade s požiadavkami (pridanie, úprava, označenie ako dokončené, odstránenie úloh).

* **Typ testovania:**

Funkčné testovanie je vhodné, pretože ide o testovanie podľa vopred definovaných špecifikácií. Testy budú skúmať, či všetky funkcie aplikácie fungujú podľa požiadaviek.

Scenár 3:

Aplikácia pre e-commerce umožňuje zákazníkom pridať produkty do nákupného košíka, prejsť k pokladni, vybrať spôsob platby a dokončiť nákup.

* **Exploratívne testovanie je vhodné.**

Aplikácia môže obsahovať rôzne faktory, ktoré ovplyvňujú nákupný proces (napr. rôzne možnosti platby, zľavy, interakcie medzi produktmi). Testeri môžu skúšať rôzne spôsoby, ako prejsť nákupným procesom a testovať stabilitu aplikácie pri rôznych interakciách.

* **Typ testovania:**

Exploratívne testovanie je vhodné, pretože umožňuje testerovi využiť svoju kreativitu na objavenie možných problémov v nákupnom procese a identifikovanie skrytých chýb.

Scenár 4:

Aplikácia na spracovanie platieb zabezpečuje, že každý platobný príkaz je spracovaný podľa stanovených pravidiel a že údaje o platbách sú správne zaznamenané v databáze.

*** Exploratívne testovanie nie je najvhodnejšie.**

Tento scenár vyžaduje presné overenie, že systém spracováva platby správne a podľa stanovených pravidiel. Pre tento typ testovania je vhodnejšie **regresné testovanie** alebo **testovanie na presnosť** a **testovanie bezpečnosti**, pretože je dôležité overiť správnosť a integritu dát.

*** Typ testovania:**

Testovanie na presnosť a **regresné testovanie** sú vhodné, pretože je potrebné zaručiť, že platby sú správne spracované a že systém neporuší predchádzajúce funkcionality pri implementácii nových funkcií.

Scenár 5:

Mobilná aplikácia na sledovanie športových výkonov zaznamenáva aktivitu používateľov, analyzuje ich výkony a poskytuje odporúčania na zlepšenie.

*** Exploratívne testovanie je vhodné.**

Aplikácia môže obsahovať rôzne faktory, ktoré ovplyvňujú výsledky (napr. odlišné športové aktivity, rôzne spôsoby zadávania dát). Testerí môžu skúšať rôzne vstupy a interakcie, aby zistili, či aplikácia správne reaguje na neočakávané situácie a analyzuje výkon správne.

*** Typ testovania:**

Exploratívne testovanie je vhodné, pretože umožňuje testerovi interagovať s aplikáciou neštandardným spôsobom a testovať rôzne spôsoby zadávania dát.

Bonus exploratívne testovanie

Cvičenie 6: Identifikácia vhodnosti exploratívneho testovania

Scenár:

Mobilná aplikácia na rezerváciu hotelov umožňuje používateľom vyhľadávať dostupné izby, filtrovať výsledky podľa ceny a recenzií a vykonávať rezervácie. Aplikácia podporuje viacero spôsobov platby vrátane kreditných kariet a digitálnych peňaženiek.

Riešenie:

1. **Exploratívne testovanie je vhodné.**

Aplikácia obsahuje rôzne funkcie, ktoré interagujú s databázou a platobným systémom. Testerí môžu skúšať rôzne kombinácie filtrov, overovať dostupnosť izieb v extrémnych prípadoch (napr. maximálny počet hostí, dátumy vzdialené niekoľko rokov do budúcnosti) a testovať, ako sa aplikácia správa pri rôznych typoch platby.

2. **Typ testovania:**

Exploratívne testovanie je vhodné, pretože umožňuje testerovi objaviť skryté chyby pri kombinácii rôznych funkcií aplikácie. Okrem toho by sa malo vykonať **testovanie použiteľnosti**, aby sa overila intuitívnosť a jednoduché používanie aplikácie.

Cvičenie 7: Identifikácia vhodnosti exploratívneho testovania

Scenár:

V zdravotníckej aplikácii lekári zapisujú informácie o pacientoch, diagnostikách a predpisujú lieky. Aplikácia musí zabezpečiť, že sa údaje ukladajú správne a nemôžu byť zmenené neoprávnenými používateľmi.

Riešenie:

1. **Exploratívne testovanie nie je vhodné ako primárna metóda.**

Keďže aplikácia pracuje s citlivými údajmi, je nevyhnutné overiť presnosť a bezpečnosť údajov. Testovanie by malo byť prísne štruktúrované a založené na vopred definovaných testovacích prípadoch.

2. **Typ testovania:**

Bezpečnostné testovanie je nevyhnutné na ochranu osobných údajov.

Funkčné testovanie sa použije na overenie správnosti zapisovania a úpravy údajov. **Regresné testovanie** zabezpečí, že nové funkcie nenarušia existujúce časti systému.

Cvičenie 8: Identifikácia vhodnosti exploratívneho testovania

Scenár:

System na spracovanie faktúr automaticky importuje údaje z rôznych formátov (CSV, PDF, XML) a kontroluje ich správnosť. Každá faktúra sa ukladá do databázy a podlieha rôznym validačným pravidlám.

Riešenie:**1. Exploratívne testovanie nie je najlepšia metóda.**

Spracovanie faktúr podlieha presne definovaným pravidlám a štandardizovaným procesom. Je dôležité, aby každý vstup bol validovaný správnym spôsobom, čo si vyžaduje systematické testovanie.

2. Typ testovania:

Automatizované testovanie je vhodné na overenie veľkého množstva vstupov v rôznych formátoch. **Funkčné testovanie** je nevyhnutné na overenie, či validácie fungujú podľa špecifikácií. **Testovanie integrity údajov** zabezpečí, že informácie sú správne uložené v databáze.

Cvičenie 9: Identifikácia vhodnosti exploratívneho testovania**Scenár:**

Herná aplikácia pre viacerých hráčov umožňuje používateľom vytvárať postavy, plniť misie a komunikovať cez chat. Vývojári chcú otestovať rôzne interakcie medzi hráčmi a možné problémy s výkonom pri veľkom počte pripojených používateľov.

Riešenie:**1. Exploratívne testovanie je vhodné.**

Herná aplikácia obsahuje dynamické interakcie, ktoré sa môžu meniť v závislosti od správania hráčov. Tester môže skúšať rôzne akcie (napr. spamovanie správ v chate, rýchle prepínanie miestností, súčasné úlohy viacerých hráčov) a identifikovať problémy, ktoré by sa v klasických testovacích scenároch neobjavili.

2. Typ testovania:

Exploratívne testovanie je vhodné na odhalenie nečakaných chýb pri interakciách hráčov. **Zátťažové testovanie** je potrebné na zistenie, ako sa hra správa pri veľkom počte hráčov. **Bezpečnostné testovanie** zabezpečí, že chat a iné interakcie nie sú zneužiteľné na útoky.

Cvičenie 10: Identifikácia vhodnosti exploratívneho testovania

Scenár:

Bankový systém pre spracovanie transakcií umožňuje klientom posielat' platby, spravovať účty a nastavovať automatické prevody. Každá operácia musí byť zaznamenaná s presným časovým údajom a nesmie byť možné ju neoprávnene upraviť.

Riešenie:

1. **Exploratívne testovanie nie je vhodné ako primárna metóda.**
Bankový systém musí spĺňať prísne požiadavky na bezpečnosť a presnosť. Každá transakcia musí byť dôsledne testovaná podľa presne definovaných pravidiel, čo vyžaduje štruktúrované testovanie.
2. **Typ testovania:**
Bezpečnostné testovanie je nevyhnutné na ochranu údajov a transakcií.
Regresné testovanie zabezpečí, že aktualizácie systému nenarušia existujúce funkcie. **Testovanie výkonu** overí, ako systém zvládne veľké množstvo transakcií súčasne.

Čo je Chyba a Defekt?

1. Nesprávne rozdelenie úloh medzi tímom (Chyba)
2. Nefunkčné prihlásenie do systému (Defekt)
3. Nesprávne zobrazenie dát v tabuľke (Defekt)
4. Zabudnutie na testovanie (Chyba)
5. Nesprávna kalkulácia cien v nákupnom košíku (Defekt)
6. Chyba v komunikácii medzi tímami (Chyba)
7. Nesprávna správa verzií (Chyba)
8. Výpadok servera pri zvýšenom zaťažení (Defekt)
9. Nedostatočné školenie používateľov (Chyba)
10. Nezohľadnenie bezpečnostných aspektov (Chyba)
11. Pretečenie zásobníka (stack overflow) (Defekt)
12. Nezabezpečené uchovávanie hesiel (Defekt)

Zhrnutie:

- **Chyba (error)** je nesprávny kód, návrh alebo konfigurácia vytvorená človekom.
- **Defekt (bug, fault)** je následok tejto chyby, ktorý sa prejaví nesprávnym správaním softvéru.

Príklad 1: Nesprávny výpočet DPH v e-shope

- **Chyba (Error):** Programátor nesprávne implementoval výpočet DPH a použil sadzbu 18 % namiesto 20 %.
 - **Defekt (Bug):** Keď zákazník pridá produkt za 100 €, e-shop mu ukáže nesprávnu cenu s DPH (118 € namiesto 120 €).
-

Príklad 2: Zlyhanie tlačidla „Odoslať objednávku“

- **Chyba (Error):** Vývojár zabudol napísať funkciu na spracovanie objednávky alebo zle nastavil event listener v JavaScripte.
 - **Defekt (Bug):** Používateľ klikne na tlačidlo „Odoslať objednávku“, ale nič sa nestane.
-

Príklad 3: Nesprávne zobrazenie na mobilných zariadeniach

- **Chyba (Error):** Dizajnér nesprávne nastavil CSS pre mobilné rozlíšenia.
 - **Defekt (Bug):** Na mobilných zariadeniach sa webová stránka zobrazuje nesprávne – niektoré tlačidlá sú mimo obrazovky.
-

Príklad 4: Pád aplikácie pri zadávaní neplatných údajov

- **Chyba (Error):** Vývojár neimplementoval správne validáciu vstupov pri registrácii.
 - **Defekt (Bug):** Keď používateľ zadá špeciálne znaky do mena, aplikácia spadne.
-

Príklad 5: Chybný SQL dotaz spôsobí prázdne výsledky

- **Chyba (Error):** Programátor omylom použil nesprávne SQL podmienky v databázovom dotaze.
- **Defekt (Bug):** Používateľ vyhľadáva produkty, ale e-shop mu zobrazuje prázdny zoznam namiesto relevantných výsledkov.

Príklad 6: Neprihlásenie do systému

- **Chyba (Error):** Programátor nesprávne implementoval overenie hesla – porovnáva ho v nesprávnom formáte.
 - **Defekt (Bug):** Používateľ zadá správne heslo, ale systém mu hlási „Nesprávne heslo“ a neumožní prihlásenie.
-

Príklad 7: Nesprávne zoradenie položiek v zozname

- **Chyba (Error):** Vývojár omylom použil **abecedné** zoradenie namiesto **číselného** v databázovom dotaze.
 - **Defekt (Bug):** Namiesto správneho poradia cien (10 €, 20 €, 100 €) sa produkty zobrazujú ako (10 €, 100 €, 20 €).
-

Príklad 8: Aplikácia posielala nesprávne emaily

- **Chyba (Error):** Vývojár použil nesprávny šablónový súbor pre automatické e-maily.
 - **Defekt (Bug):** Používateľ dostane e-mail s potvrdením registrácie, aj keď sa len odhlásil z odberu noviniek.
-

Príklad 9: Stránka sa nenačíta v určitom prehliadači

- **Chyba (Error):** Frontend vývojár nepoužil kompatibilné CSS štýly pre staršie verzie Internet Explorera.
 - **Defekt (Bug):** Používateľ otvorí stránku v Internet Exploreri, ale vidí len bielu obrazovku.
-

Príklad 10: Nesprávny výpočet zľavy v nákupnom košíku

- **Chyba (Error):** Programátor nesprávne zaokrúhlil sumy pri aplikovaní percentuálnej zľavy.
- **Defekt (Bug):** Používateľovi mala byť udelená zľava 10 %, ale v skutočnosti dostane iba 9,8 % alebo 10,2 %.