Ako testovať prechod medzi prostrediami (napr. vývoj → test → produkcia)?

Príklad:

Predstavte si, že testujete aplikáciu, kde sa v testovacom prostredí zobrazuje nové tlačidlo *„Exportovať report“*. Na vývoji to fungovalo, ale na teste sa to správa inak – neodosiela žiadosť na server. Na produkcii sa tlačidlo ani nezobrazuje. Problém? Zlá konfigurácia prostredí.

Pri testovaní medzi prostrediami sa nekontroluje len funkcionalita, ale aj správna *konfigurácia*, *verzie*, *feature toggles* či *dáta*. Tu je zoznam, čo si pri tom treba postrážiť:

**Čo všetko otestovať pri prechode medzi prostrediami:**

1. **Konfiguračné rozdiely**
   * Skontroluj, či má každé prostredie správne nastavené URL, porty, API kľúče, CORS, SMTP, databázy, externé integrácie a ďalšie parametre.
   * Porovnaj config súbory (napr. config.yml, .env, application.properties) alebo si prebehni výpisy z CI/CD nástrojov (napr. GitLab CI, Jenkins), kde bývajú uvedené použité premenné, verzia buildov či nasadené konfigurácie.
2. **Feature toggles / flags**
   * Feature toggles sú prepínače, ktorými možno zapnúť alebo vypnúť konkrétnu funkcionalitu bez nutnosti meniť kód.
   * Napríklad v súbore config.sys sa nastaví parameter DHL=1, čím sa aktivuje export zásielok pre DHL kuriéra. Vďaka tomu je možné funkciu nasadiť do všetkých prostredí, ale povoliť ju len tam, kde je otestovaná a pripravená na použitie.
3. **Verzia aplikácie**
   * Má test aj produkcia rovnakú verziu (build number, commit hash – build number je číslo zostavenia, commit hash je jedinečný identifikátor konkrétneho commitu v repozitári)?
   * Over cez UI (verzia v pätičke) alebo API ((napr. /version endpoint – špeciálne API, ktoré vráti informácie o nasadenej verzii aplikácie)).
4. **Testovanie dát**
   * Sú testovacie dáta realistické? Neobsahujú citlivé údaje z produkcie?
   * Pri migrácii dát otestuj ich konzistenciu a integritu – teda či si zachovali vzťahy a štruktúru (napr. objednávky patria správnym zákazníkom) a či sa počas prenosu nič nepoškodilo alebo nestratilo. Porovnaj počet záznamov, kontrolné súčty, väzby v databáze alebo použij vzorkovanie dát z pôvodného a cieľového prostredia.
   * Ak sa importujú dáta z jedného prostredia do druhého, over ich úplnosť.
5. **Prístupové práva a role**
   * Over, že používateľské role majú správne práva v každom prostredí.
   * Často sa stane, že v test má admin oprávnenia, ktoré v produkcii nemá mať.
6. **Logy a monitorovanie**
   * Sú logy aktívne vo všetkých prostrediach? Nepíšu sa debug logy na produkciu? (Debug logy sú detailné technické výpisy určené pre vývojárov; na produkcii by mali byť vypnuté, aby neznižovali výkon alebo neodhalili citlivé údaje.)
   * Funguje prepojenie na monitoring (napr. Grafana, Kibana, Sentry...)?
7. **Bezpečnosť a obmedzenia**
   * Over SSL certifikáty (slúžia na šifrovanie komunikácie medzi klientom a serverom), správne redirecty (automatické presmerovanie na správne URL), HTTP headers (napr. Content-Security-Policy alebo X-Frame-Options nastavujú bezpečnostné pravidlá pre prehliadač) a ochranu pred CORS (Cross-Origin Resource Sharing – mechanizmus, ktorý určuje, ktoré domény majú povolený prístup k API).
   * Test prostredie by nemalo byť verejne prístupné ako produkcia.
8. **Prístupnosť prostredí**
   * Je test aj produkcia dostupná z rovnakého sieťového prostredia? (Napr. obe sú dostupné len cez VPN, alebo je produkcia verejná a test iba interne. Rôzne siete môžu ovplyvniť dostupnosť alebo spôsobiť rozdielne chovanie aplikácie.)
   * Over dostupnosť (či sa aplikácia vôbec načíta), latency (oneskorenie medzi požiadavkou a odpoveďou zo servera) a správne DNS smerovanie (preklad doménového mena na IP adresu servera).
9. **Fallbacky a defaulty**
   * Ak niečo chýba v konfigu, čo sa stane? Skončí aplikácia v chybe alebo má rozumné predvolené správanie?

Tester nie je len "klikací overovač", ale strážca *kontextu* – rozumie rozdielom medzi prostrediami a vie, že aj najlepšia funkcia zlyhá, ak má zlé podmienky na beh.

**#testing #qualityassurance #softwaretesting #environment #devtestprod #testovanie #QA #config #testerlife**