Ako testovať, keď v novej verzii pribudli stĺpce v databáze?

Predstav si, že aplikácia prešla aktualizáciou a k tabuľke objednavky boli pridané nové stĺpce: zdroj\_objednavky a typ\_zakaznika. Backend už tieto hodnoty ukladá, UI ich zobrazuje, a reporty sa majú čoskoro meniť.

Ako to celé otestovať, aby si mal(a) istotu, že nič neuniklo?

**Krok za krokom:**

1. **Over štruktúru databázy**

* Tester si otvorí databázu (napr. cez SQL klient) a priamo si zobrazí štruktúru tabuľky, kde boli pridané stĺpce – napríklad pomocou príkazu `DESCRIBE objednavky;` (MySQL) alebo `\d objednavky` (PostgreSQL).
* Skontroluj, či nové stĺpce existujú, aký majú dátový typ, či majú nastavenú default hodnotu (ak má byť), či povoľujú `NULL`, a či sú na nich definované indexy (ak treba).

1. **Otestuj zápis a čítanie dát**

* Vytvor záznam cez aplikáciu alebo API a over, že sa nové stĺpce správne naplnia.
* Skús aj „edge cases“ – čo sa stane, ak pole necháš prázdne? Čo ak zadáš nesprávny typ?

1. **Zachovaj spätnú kompatibilitu**

* Aplikácia nesmie „spadnúť“, ak staršie záznamy tieto stĺpce ešte nemajú vyplnené.
* Testuj aj staršie funkcionality, ktoré s týmito stĺpcami vôbec nerátajú.

1. **Doplň testy na výpis / zobrazenie**

* Zobrazujú sa nové stĺpce v UI správne? Máš tam správne popisky, formátovanie, lokalizáciu?

1. **Over integráciu a prepojenia**

* Ak sa tieto stĺpce prenášajú do iných systémov (napr. CRM, BI reporty), treba otestovať export, synchronizáciu a mapovanie.

1. **Zmeny v migráciách a skriptoch**

* Skontroluj, že aktualizačný SQL skript prebehol úspešne (napr. `ALTER TABLE`) a nespôsobil chyby v nasadení.
* Over, že nové stĺpce naozaj vznikli (napr. pomocou DESCRIBE objednavky; v MySQL alebo `\d objednavky` v PostgreSQL).
* Sleduj logy po nasadení – nemali by sa objaviť chyby typu „**syntax error**“, „**unknown column**“ alebo „**duplicate column name**“.
* Ak databáza funguje vo viacerých verziách (napr. pre rôznych klientov), skontroluj, či migrácia funguje aj na starších dátach.
* Každý migračný skript by mal byť verzovaný a súčasťou zoznamu migrácií (napr. `V2025\_08\_01\_\_add\_new\_columns.sql`).

**Čo je to migrácia v databáze?**  
Migrácia je verzovaný skript (napr. v SQL), ktorý mení štruktúru databázy – pridáva nové stĺpce, upravuje typy, maže staré polia. Umožňuje vývojárom aj testerom riadene prejsť z jednej verzie databázy do druhej bez ručného zásahu. Každá verzia systému má tak jasne definovanú podobu databázy.

Napr. skript `V2025\_08\_01\_\_add\_new\_columns.sql` môže obsahovať:

```

sql

ALTER TABLE objednavky

ADD COLUMN zdroj\_objednavky VARCHAR(100),

ADD COLUMN typ\_zakaznika VARCHAR(50);

```

Migrácie sú súčasťou kódu, verzované v Gite a spúšťajú sa automaticky pri nasadení novej verzie aplikácie.

1. **Logy a monitoring**

Sleduj logy aplikácie po nasadení. Neobjavujú sa výnimky typu „unknown column“?

Pridanie stĺpcov v DB sa môže zdať ako drobná zmena – ale pre testera je to veľká príležitosť zachytiť nečakané dopady v systéme.

#testovanie #qualityassurance #databaza #backendtesting #manualtesting #teststrategy #sql #linkedinpretestera