Ako testovať „nezmazateľný“ záznam (soft delete)

V niektorých systémoch býva **soft delete** implementovaný veľmi jednoducho – záznam sa fyzicky neodstráni, len sa v databáze nastaví určitá hodnota (napr. is\_deleted = true alebo stav = 'zmazané').  
Vďaka tomu sa dá záznam opäť zobraziť úpravou tejto hodnoty, ak je na to oprávnenie. V logoch alebo histórii záznamu býva zaznamenané, **kto a kedy** záznam zmazal.

Takéto riešenie je praktické, ale vyžaduje **dôsledné testovanie**. Na čo všetko sa zamerať?

**Čo otestovať pri soft delete?**

1️⃣ **Zobrazenie v UI**

* Bežný používateľ záznam **nevidí**
* Admin/tester ho **vidí**, prípadne s info „zmazané“
* V detaile záznamu môže byť hláška „Tento záznam bol zmazaný“

2️⃣ **API správanie**

* GET môže vrátiť 404, 410 alebo špeciálnu hlášku
* PUT/PATCH/DELETE: povolené len pre vybrané roly?
* Zabezpečiť, aby sa „mŕtvy“ záznam nedal modifikovať bez oprávnení

3️⃣ **Zmeny v databáze**

* Pole is\_deleted, deleted\_at, deleted\_by (alebo omyl) má správnu hodnotu
* Záznam **nebol fyzicky zmazaný**
* Dáta sú konzistentné (žiadne siroty v iných tabuľkách)

4️⃣ **Logy a audit**

* Kto, kedy a ako záznam zmazal?
* Je zaznamenaný aj pokus o prístup k zmazanému záznamu?

5️⃣ **Obnovenie záznamu**

* Dá sa obnoviť cez UI alebo iba cez databázu?
* Obnova vracia záznam do pôvodného stavu?
* Funguje aj prepojenie na ďalšie dáta (napr. komentáre, prílohy)?

6️⃣ **Bezpečnostné testy**

* Vie si bežný používateľ načítať zmazané dáta cez API/URL?
* Ošetrené sú aj pokusy o úmyselnú obnovu dát bez práv?

**Tip pre testerov:**  
Pri testovaní soft delete sa pozeraj nielen na **to, čo vidíš**, ale aj na to, **čo už vidieť nemáš** – a či sa to nedá obísť. Kombinuj testy cez UI, API a priamu kontrolu v databáze.

**#SoftDelete #Testovanie #ManualQA #DatabaseTesting #SecurityInTesting #TesterTip**