

## Zadanie 5 (10b)

Zadanie 5 je zamerané na implementovanie API rozhrania nad návrhom databázy zo zadania 4. V prípade potreby je možné návrh databázy upraviť podľa potreby pre implementovanie zadania 5.

**Termín odovzdania:** 23:59 7.5.2023 pre všetky časti odovzdania (AIS, Github a Nahlasenie linku na docker image cez formulár pre zadanie 5)

### **Pokyny k odovzdaniu:**

- **AIS** – odovzdávajú sa zdrojové kódy (tak aby bolo možné skompilovať a spustiť projekt) spolu s číslom commitu, ktorý predstavuje finálnu verziu Vášho zadania 5.
- **Github** – vytvorenie nového repozitára pre zadanie 5 v rámci predmetového github classroomu:
  - o Balazia po-16:00: <https://classroom.github.com/a/jpXMFbme>
  - o Bencel stv-10:00: <https://classroom.github.com/a/ceS3v43a>
  - o Binder po - 14:00: [https://classroom.github.com/a/qHh\\_xgA7](https://classroom.github.com/a/qHh_xgA7)
  - o Binder po - 16:00: <https://classroom.github.com/a/1hcriQFS>
  - o Dubec stv - 8:00: <https://classroom.github.com/a/JHCqqtxF>
  - o Findura po - 14:00: <https://classroom.github.com/a/RttPpDVH>
  - o Findura stv - 8:00: <https://classroom.github.com/a/9o57DO0f>
  - o Findura stv - 10:00: <https://classroom.github.com/a/qcJASz6V>
- **Formulár** – nahlásenie linku na Váš docker image v rámci vášho Github repozitára s príslušným označením podľa pokynov nižšie
  - o link: <https://forms.gle/enLBbo6weYyTYaNdA>

Odobzdanie je považované za kompletne, ak sú odobzdané všetky časti (AIS, Github, Link na docker image) do termínu odobzdania. V prípade neskorého odobzdania je zadanie považované ako neodobzdané.

### **Bodové hodnotenie**

1b kvalita kódu  
1b dokumentácia  
8b endpointy

### **Popis zadania**

V rámci nového repozitára je potrebné implementovať API pre navrhnutú databázu zo zadania 4. Definovanie API sa nachádza v rámci yml súboru dostupného na <https://github.com/FIIT-Databases/api-assignment>. Tento súbor je možné otvoriť napr. <https://stoplight.io>. V rámci API nie je riešená žiadna autentifikácia požiadaviek.

V rámci jednotlivých endpointov je potrebné kontrolovať formát vstupov podľa definovania v API a v prípade zistenia nesprávneho formátu je potrebné vrátiť odpoveď s HTTP response code 400.

V rámci implementácie je potrebné uskutočniť automatické migrácie tj. v prípade, že neexistuje štruktúra (relačná schéma) Vašej databázy v rámci servera, tak je vytvorená. Treba kontrolovať pri tom ako sa spúšťa Váš aplikačný server. Pri každom odovzdaní do testera sa Vám vytvára nová databáza na serveri.

Spustenie tejto migrácie **je nevyhnutné** pre možnosti overenia a vyhodnotenia funkčnosti Vášho riešenia. V prípade, že Vaše riešenie nedokáže uskutočniť túto začiatočnú migráciu, tak je Vaše riešenie považované za neakceptované.

Pre overenie funkčnosti Vášho implementovaného API **je tiež nevyhnutné**, aby Vám fungovali endpointy typu POST a GET. Pre možnosť otestovania zadania je možné použiť <https://tester-dbs.fiiit.stuba.sk>. V prípade metódy POST sa nachádza v rámci popisu aj ID záznamu. Toto ID záznamu sa v normálnych prípadoch negeneruje na strane klienta, ale na strane servera. V rámci zadania je ho potrebné generovať na strane klienta kvôli možnostiam testovania správnosti Vášho API. V praxi **je ho nutné** generovať na strane servera.

V rámci zadania 5 je možné počas implementácie používať aj **ORM**.

Okrem implementovanie samotného API je potrebné vyhotoviť dokumentáciu k Vášmu riešeniu, ktoré bude obsahovať:

- použité SQL dopyty pre jednotlivé endpointy s ich popisom.
- zmeny v návrhu DB oproti zadaniu 4.

Dokumentácia môže byť realizovaná ako PDF alebo markdown dokumentácia s tým, že sa bude nachádzať v AIS odovzdaní a aj v samotnom github repozitári.