# FINÁLNÍ PROJEKT č.1



Autor: Filip Rozum (Discord: F.Rozum/Denizer)

Datum: 24.6.2024

# OBSAH

ZADÁNÍ	3
TESTOVACÍ SCÉNÁŘE	4
EXEKUCE TESTŮ	5
BUG REPORT	5

# ZADÁNÍ

Čílem finálního projektu je otestovat funkčnost aplikace, která slouží k manipulaci s daty o studentech. Aplikace má rozhraní REST-API, které umožňuje vytvoření, smazání a získání dat..

# Přístupové údaje:

Databáze	database: qa_demo Host: aws.connect.psdb.cloud Username: sgqcu2878p6q2zzw5tw4 Password: *****
REST-API	http://108.143.193.45:8080/api/v1/students/

# Poznámky:

Nezapomeňte, že v IT se data musí někde uložit a poté získat. Proto ověřte, že data jsou správně uložena a získávána z databáze.

Nezapomeňte do testovacích scénářů uvést testovací data, očekávaný výsledek včetně těla odpovědi a stavových kódů.

# TESTOVACÍ SCÉNÁŘE

Na základě uvedených testovacích scénářů jsem ověřil(a) funkčnost aplikace.

1.GET

příprava dat: ID existujícího studenta = 961, SELECT \* FROM student WHERE id=961

abstrakt: Ukaž údaje existujícího studenta

}

Test pro neexistujícího studenta: Opakuj postup s id neexistujícího studenta Očekávaný výsledek - status code 404

#### 2.DELETE

příprava dat: ID existujícího studenta = 755, SELECT \* FROM student WHERE id=755

abstrakt: Vymaž všechny údaje existujícího studenta

Postup: Zadej adresu http://108.143.193.45:8080/api/v1/students/755, vyber možnost DELETE a klikni na Send

Očekávaný výsledek- status code 200 OK

- All data about particular student are REMOVED

-Poté kontrola smazání údajů v databázi

Test pro neexistujícího studenta: Opakuj postup s id neexistujícího studenta Očekávaný výsledek - status code 404

### 3.POST

příprava dat: Vymyslet data pro nového studenta (firstName = Vernon, lastName = Roach, email = vroach@stripes.com, age = 34)

abstrakt: Přidej nového studenta

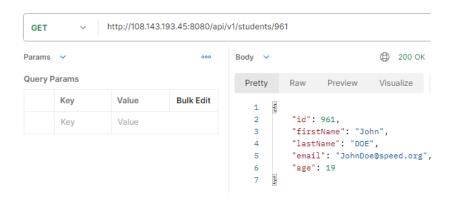
Postup: -Zadej adresu http://108.143.193.45:8080/api/v1/students/ a zvol možnost POST.

- -Poté kontrola přidaného studenta v databázi
- -Pro úspěšné založení studenta je nutné zadat všechny atributy (jinak status code 400 bad request)
- -age musí být 1-150 let, firstName a lastName 3-50 znaků, email ve formátu @.\*\*\*
   jinak status code 400 bad request
- -Všechny atributy se do databáze mají ukládat formou lowerCase

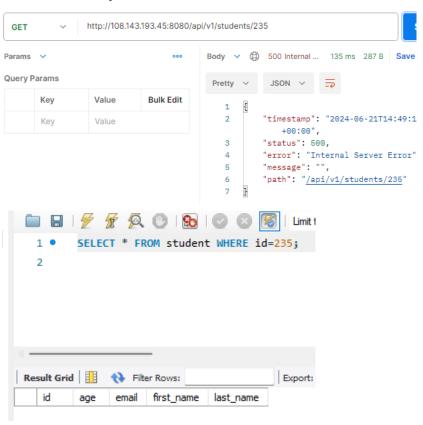
# **EXEKUCE TESTŮ**

Testovací scénáře jsem provedl(a), přikládám výsledky testů.

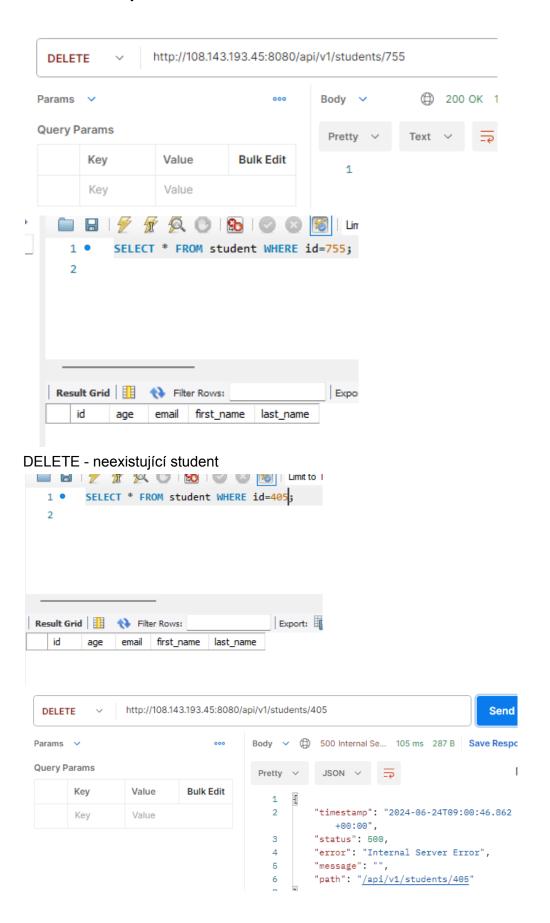
# GET - existující student



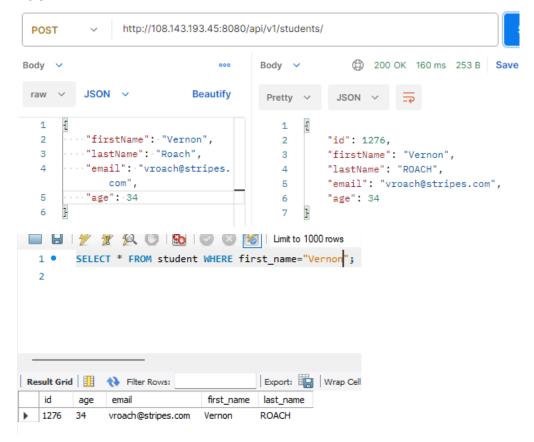
## GET - neexistující student



### DELETE - existující student



#### **POST**



### POST - další testy

1. chybí atribut



2. nepovolený počet znaků ve firstName

```
Body V
                                                              (f) 200 OK 17
  Pretty
            Raw
                    Preview
                                Visualize
                                            JSON ~
    1
    2
            "id": 1279,
            "firstName": "Li",
    3
            "lastName": "RENO",
    4
            "email": "ahfnf@hbg.cs",
    5
            "age": 35
    6
    7
```

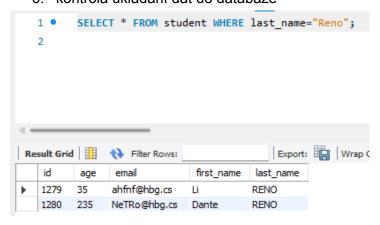
3. nepovolený věk

```
Body V
                                                               (f) 200 OK 1
   Pretty
                     Preview
             Raw
                                Visualize
                                            JSON V
     1
             "id": 1280,
     2
             "firstName": "Dante",
     3
             "lastName": "RENO",
     5
             "email": "NeTRo@hbg.cs",
             "age": 235
     6
     7
   4. nepovolený počet znaků v lastName
Body Cookies Headers (5) Test Results
                                                                (#) 200 OK 180
```

### 5. nepovolený formát emailu



### 6. kontrola ukládání dat do databáze



# **BUG REPORT**

Na základě provedených scénářů jsem objevil(a) uvedené chyby aplikace.

GET - Při zobrazení neexistujícího studenta - status code 500 (má být 404)

DELETE - Při smazání studenta chybí odezva "All data about particular student are REMOVED"

- Při mazání neexitujícího studenta - status code 500 (má být 404)

### POST a ukládání dat do databáze

- Při založení studenta status code 200 OK (má být 201 Created)
- Při chybějícím atributu status code 500 (má být 400)
- U všech případů špatně zadaného atributu byl student úspěšně založen (status code 200 OK) - Má se zobrazit status code 400 bad request
- Databáze neukládá data v lowerCase
  - email a firstName zůstává v původní formě
  - lastName se ukládá v upperCase