

FINÁLNÍ PROJEKT

č.1



ENGETO

Autor: Ján Jankovič
Datum: 24. 11. 2024

OBSAH

ZADÁNÍ	3
TESTOVACÍ SCÉNÁŘE	4
EXEKUCE TESTŮ	8
BUG REPORT	13

ZADÁNÍ

Cílem 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..

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

Scenár 1: Vrátene záznamov všetkých študentov, ktorí sú uložení v databázi pomocou metódy GET

- **Vstup:** GET <http://108.143.193.45:8080/api/v1/students/>
- **Očakávaný výsledok:** Status 200 OK, požiadavka bola úspešná a server poskytol odpoveď

Scenár 2: Vrátene záznamov konkrétneho študenta na základe ID študenta pomocou metódy GET

- **Vstup:** GET <http://108.143.193.45:8080/api/v1/students/<id-studenta>>
 - id-studenta = 352, 354, 357 apod.
- **Očakávaný výsledok:** Status 200 OK, požiadavka bola úspešná a server poskytol odpoveď
- **Response body:**

```
{
  "id": 354,
  "firstName": "alfred",
  "lastName": "HITCHCOCK",
  "email": "alfred@mail.com",
  "age": 18
}
```

Scenár 3: Zadanie textu namiesto čísla ID na konci odkazu pomocou metódy GET

- **Vstup:** GET <http://108.143.193.45:8080/api/v1/students/Jana>
- **Očakávaný výsledok:** Status 400 Bad Request, požiadavka je nesprávne zostavená alebo obsahuje neplatné údaje
- **Response body:**

```
{
  "status": "error",
  "message": "Invalid ID format. ID must be a numeric value.",
  "code": 400
}
```

Scenár 4: Zadanie neexistujúceho čísla ID na konci odkazu pomocou metódy GET

- **Vstup:** GET <http://108.143.193.45:8080/api/v1/students/11111>
- **Očakávaný výsledok:** Status 404 Not Found, požadovaný zdroj neexistuje alebo nebol nájdený

- **Response body:**

```
{
  "status": "error",
  "message": "Student with ID 11111 not found.",
  "code": 404
}
```

Scenár 5: Vytvorenie nového študenta pomocou metódy POST

- **Vstup:**

```
{
  "firstName": "Ján",
  "lastName": "Jankovič",
  "email": "jankovic@gmail.com",
  "age": 31
}
```

- **Očakávaný výsledok:** Status 201 Created, nový zdroj bol úspešne vytvorený a bolo mu pridelené vlastné ID. Vracia hodnotu zo vstupu rozšírenú o parameter **"id": "číslo_noveho_id"**

- **Response body:**

```
{
  "id": "číslo_noveho_id"
  "firstName": "Ján",
  "lastName": "Jankovič",
  "email": "jankovic@gmail.com",
  "age": 31
}
```

Scenár 6: Vytvorenie nového študenta pomocou metódy POST a zadanie číselných hodnôt pre "firstName" a "lastName"

- **Vstup:**

```
{
  "firstName": "1234",
  "lastName": "4321",
  "email": "1234@gmail.com",
  "age": 31
}
```

- **Očakávaný výsledok:** Status 400 Bad Request, požiadavka je nesprávne zostavená alebo obsahuje neplatné údaje

- **Response body:**

```
{
  "status": "error",
  "message": "Invalid input: 'firstName' and 'lastName' must contain alphabetic characters.",
  "code": 400
}
```

Scenár 7: Vytvorenie nového študenta pomocou metódy POST s neúplným zadáním parametrov študenta (bez parametru "age")

- **Vstup:**

```
{
  "firstName": "1234",
  "lastName": "4321",
  "email": "1234@gmail.com"
}
```

- **Očakávaný výsledok:** Status 400 Bad Request, požiadavka je nesprávne zostavená alebo obsahuje neplatné údaje

- **Response body:**

```
{
  "status": "error",
  "message": "Missing required parameter: 'age'.",
  "code": 400
}
```

Scenár 8: Vytvorenie nového študenta pomocou metódy POST a zadáním väčšieho počtu parametrov študenta

- **Vstup:**

```
{
  "firstName": "1234",
  "lastName": "4321",
  "email": "1234@gmail.com",
  "age": 23,
  "age1": 31
}
```

```
{
  "firstName": "1234",
  "lastName": "4321",
  "email": "1234@gmail.com",
  "age": 23,
  "firstName1": "1234",
  "lastName1": "4321",
  "email1": "1234@gmail.com",
  "age": 31
}
```

- **Očakávaný výsledok:** Status 400 Bad Request, požiadavka je nesprávne zostavená alebo obsahuje neplatné údaje

- **Response body:**

```
{
  "status": "error",
  "message": "Unexpected
parameter in request:
'age1'.",
  "code": 400
}
```

```
{
  "status": "error",
  "message": "Unexpected parameters
in request: 'firstName1', 'lastName1',
'email1', 'age1'.",
  "code": 400
}
```

Scenár 9: Vytvorenie nového študenta, ktorý už v databáze existuje a má pridelené ID pomocou metódy POST

- **Vstup:** {"firstName": "Ján", "lastName": "Jankovič", "email": "jankovic@gmail.com", "age": 31}
 - Vstup zadaný 2x
- **Očakávaný výsledok:** Status 400 Bad Request, zadaný študent sa už v databáze nachádza a má pridelené vlastné ID
- **Response body:**

```
{
  "status": "error",
  "message": "Student with the provided email already exists in the database with its own ID.",
  "code": 400
}
```

Scenár 10: Vytvorenie nového študenta s neplatnou e-mailovou adresou pomocou metódy POST

- **Vstup:** {"firstName": "Ján", "lastName": "Jankovič", "email": "jankovicgmail.com", "age": 31}
- **Očakávaný výsledok:** Status 400 Bad Request, požiadavka je nesprávne zostavená alebo obsahuje neplatné údaje
- **Response body:**

```
{
  "status": "error",
  "message": "Invalid email format: 'jankovicgmail.com'. A valid email must contain '@' and a domain.",
  "code": 400
}
```

Scenár 11: Odstránenie záznamov konkrétneho študenta na základe ID študenta pomocou metódy DELETE

- **Vstup:** DELETE http://108.143.193.45:8080/api/v1/students/<id-studenta>
 - id-studenta = 2338
- **Očakávaný výsledok:** Status 200 OK, požiadavka bola úspešná
- **Response body:**

```
{
  "status": "success",
  "message": "Student with ID 2338 has been successfully deleted."
  "code": 200
}
```

Scenár 12: Odstránenie záznamov už odstráneného študenta v scenári 11 na základe ID študenta pomocou metódy DELETE

- **Vstup:** DELETE http://108.143.193.45:8080/api/v1/students/<id-studenta>

- id-studenta = 2338
- **Očakávaný výsledok:** Status 404 Not Found, požadovaný zdroj neexistuje alebo nebol nájdený
- **Response body:**

```
{  
  "status": "error",  
  "message": "Student with ID 2338 not found."  
  "code": 404  
}
```


EXEKUCE TESTŮ

Scenár 1

Test bol úspešne vykonaný. API vrátilo správny stavový kód i odpoveď (response body). Response body zodpovedá očakávaniu. Podrobnosti viď screenshot nižšie.

GET http://108.143.193.45:8080/api/v1/students/

Params Authorization Headers (6) Body Pre-request Script Tests Settings

Query Params

Key	Value
Key	Value

Body Cookies Headers (5) Test Results 200 OK 1728 ms 112.79 KB

Pretty Raw Preview Visualize JSON

```
1 {
2   "id": 352,
3   "firstName": "Jana",
4   "lastName": "NOVAKOVA",
5   "email": "janovak@gmail.com",
6   "age": 19
7 },
8 {
9   "id": 354,
10  "firstName": "alfred",
11  "lastName": "HITCHCOCK",
12  "email": "alfred@mail.com",
13  "age": 18
14 },
15 }
```

Scenár 2

Test bol úspešne vykonaný. API vrátilo správny stavový kód i odpoveď. Response body zodpovedá očakávaniu. Podrobnosti viď screenshot nižšie.

GET http://108.143.193.45:8080/api/v1/student/352

Params Authorization Headers (6) Body Pre-request Script Tests Settings

Query Params

Key	Value
Key	Value

Body Cookies Headers (5) Test Results 200 OK 88 ms 252 B

Pretty Raw Preview Visualize JSON

```
1 {
2   "id": 352,
3   "firstName": "Jana",
4   "lastName": "NOVAKOVA",
5   "email": "janovak@gmail.com",
6   "age": 19
7 }
```

Scenár 3

Test bol úspešne vykonaný. API vrátilo správny stavový kód 400. Response body zodpovedá očakávaniu. Podrobnosti viď screenshot nižšie.

```

1  {
2    "timestamp": "2024-11-15T10:13:44.665+00:00",
3    "status": 400,
4    "error": "Bad Request",
5    "message": "",
6    "path": "/api/v1/students/Jana"
7  }

```

Scenár 4

Test nebol úspešný. API vrátilo nesprávny stavový kód 500 namiesto stavového kódu 404. Response body nezodpovedá očakávanému výsledku. Podrobnosti viď screenshot nižšie.

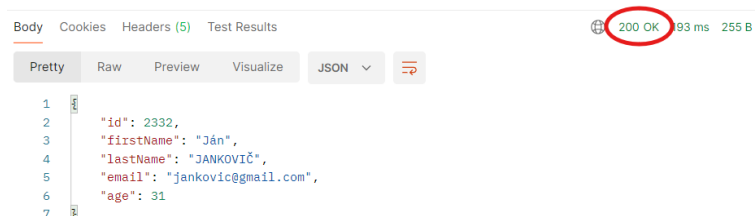
```

1  {
2    "timestamp": "2024-11-15T10:31:35.971+00:00",
3    "status": 500,
4    "error": "Internal Server Error",
5    "message": "",
6    "path": "/api/v1/students/11111"
7  }

```

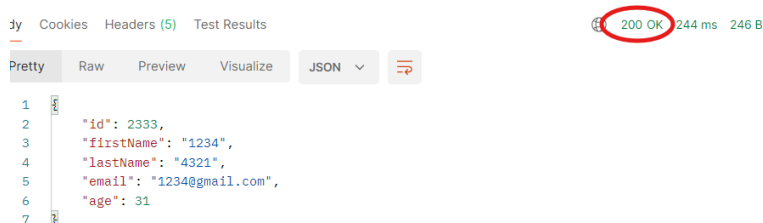
Scenár 5

Test nebol úspešný. API vrátilo nesprávny stavový kód 200 namiesto očakávaného stavového kódu 201. Response body nezodpovedá očakávanému výsledku. Podrobnosti viď screenshot nižšie.



Scenár 6

Test nebol úspešný. API vrátilo nesprávny stavový kód 200 namiesto očakávaného stavového kódu 400. Response body nezodpovedá očakávanému výsledku. Podrobnosti viď screenshot nižšie.



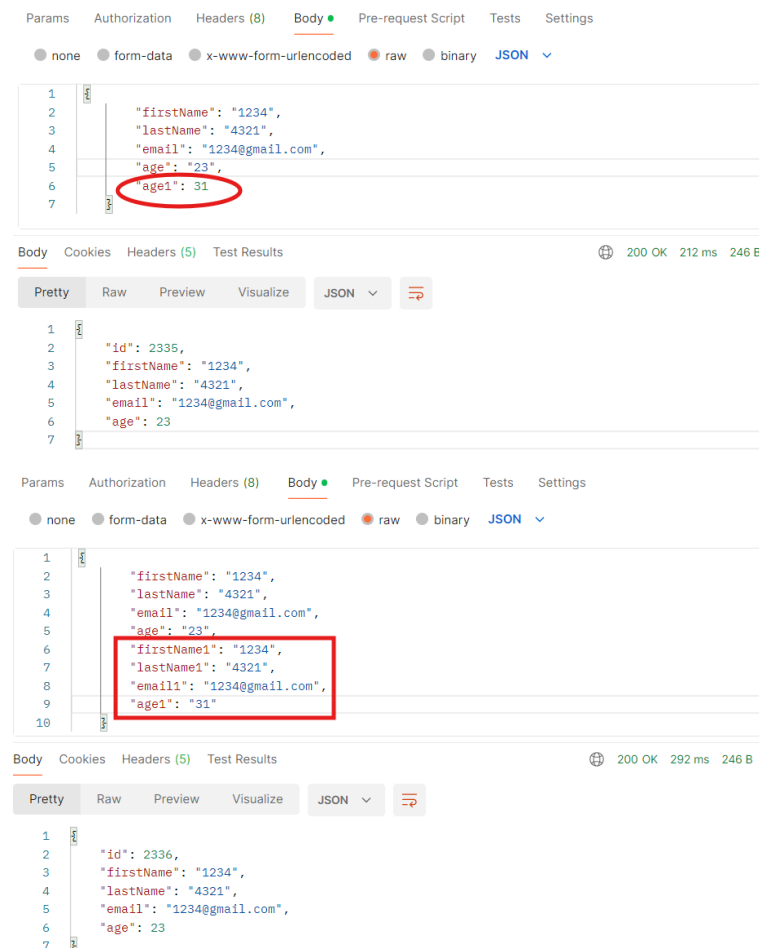
Scenár 7

Test nebol úspešný. API vrátilo nesprávny stavový kód 500 namiesto očakávaného stavového kódu 400 kvôli chýbajúcim parametrom. Response body nezodpovedá očakávanému výsledku. Podrobnosti viď screenshot nižšie.



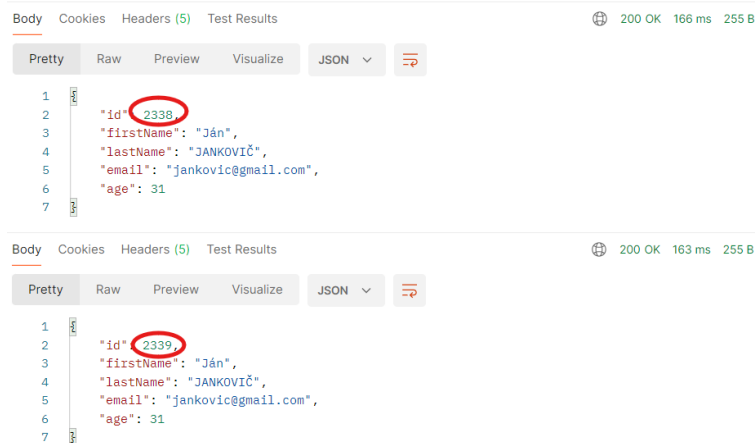
Scenár 8

Test nebol úspešný. API vrátilo nesprávny stavový kód 200 a ignoruje parametre zadané navyše namiesto očakávaného stavového kódu 400 kvôli väčšiemu množstvu parametrov. Zároveň vytvára rovnakého študenta pod iným ID. Response body nezodpovedá očakávanému výsledku. Podrobnosti viď 2 prípady v screenshotoch nižšie.



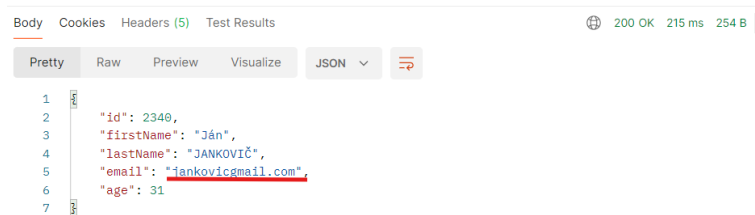
Scenár 9

Test nebol úspešný. API vrátilo nesprávny stavový kód 200 a vytvára rovnakých študentov, ktorým prideliuje rozdielne ID, namiesto očakávaného stavového kódu 400 kvôli vytvoreniu študenta, ktorý už v databáze existuje. Response body nezodpovedá očakávanému výsledku. Podrobnosti viď v screenshotoch nižšie.



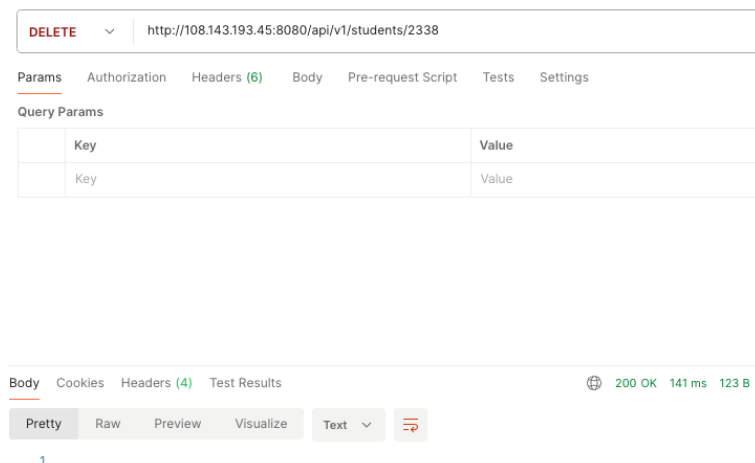
Scenár 10

Test nebol úspešný. API vrátilo nesprávny stavový kód 200 a vytvára nového študenta, s neplatnou e-mailovou adresou (bez znaku "@"), namiesto očakávaného stavového kódu 400, boli zadané neplatné údaje. Response body nezodpovedá očakávanému výsledku. Podrobnosti viď screenshot nižšie.



Scenár 11

Test bol úspešne vykonaný. API vrátilo správny stavový kód a vymazalo študenta so zadaným ID. Response body zodpovedá očakávaniu. Podrobnosti viď screenshot nižšie.



Scenár 12

Test nebol úspešný. API vrátilo nesprávny stavový kód 500 namiesto očakávaného stavového kódu 404 kvôli tomu, že sa snažíme odstrániť študenta pod ID, ktoré v databáze neexistuje. Response body nezodpovedá očakávanému výsledku. Podrobnosti viď screenshot nižšie.

DELETE
⌵
http://108.143.193.45:8080/api/v1/students/2338

Params
Authorization
Headers (6)
Body
Pre-request Script
Tests
Settings

Query Params

	Key	Value
	Key	Value

Body
Cookies
Headers (4)
Test Results
500 Internal Server Error
200 ms
288 B

Pretty
Raw
Preview
Visualize
JSON
⌵
🔗

```

1  {
2    "timestamp": "2024-11-18T15:50:21.692+00:00",
3    "status": 500,
4    "error": "Internal Server Error",
5    "message": "",
6    "path": "/api/v1/students/2338"
7  }
```

BUG REPORT

Na základě provedených scénářů jsem objevil uvedené chyby aplikace.

Bug ID	Pravdepodobnosť	Závažnosť	Číslo test. scénára	Mitigácia
1	Stredná	Stredná	Scenár 3	Oprava serverovej logiky pre správne spracovanie chýb a vrátenie kódu 404 Not Found pri neexistujúcom ID
2	Stredná	Stredná	Scenár 4	Oprava serverovej logiky pre správne spracovanie chýb a vrátenie kódu 201 Created pri vytvorení nového študenta
3	Vysoká	Stredná	Scenár 6	Pridať validáciu vstupných údajov na strane servera, ktorá overí, že "firstName" a "lastName" obsahujú len textové reťazce
4	Stredná	Stredná	Scenár 7	Oprava serverovej logiky pre správne spracovanie chýb a vrátenie kódu

				400 Bad Request pri chýbajúcom parametri
5	Stredná	Stredná	Scenár 8	Pridať validáciu vstupných údajov na strane servera, ktorá overí, že bol zadáný správny počet parametrov
6	Vysoká	Stredná	Scenár 8 a 9	Pridať validáciu vstupných údajov na strane servera, ktorá overí, že zadávaný študent s rovnakými parametrami ešte v databáze nie je
7	Vysoká	Stredná	Scenár 10	Pridať validáciu vstupných údajov na strane servera, ktorá overí, že parameter "email" je zadáný v správnom formáte (text@domena)
8	Stredná	Stredná	Scenár 12	Oprava serverovej logiky pre správne spracovanie chýb a vrátenie kódu 404 Not Found pri snahe o odstránenie neexistujúceho študenta