

FINÁLNÍ PROJEKT

č.1



ENGETO

Autor: Ludmila Sluková
Datum: 22. 8. 2024

OBSAH

ZADÁNÍ	3
TESTOVACÍ SCÉNÁŘE	4
EXEKUCE TESTŮ	8
BUG REPORT	11

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..

Přístupové údaje:

Databáze	database: qa_demo Host: aws.connect.psdb.cloud Username: im3s0o2b4w0lpcboxk0k Password: pscale_pw_8Nc3bfP1t3ECEJm8cGPze47M egBUWQzgH7J8XsLghBi
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 ověřila funkčnost aplikace:

Testovací scénář 01: Všechna pole musí být vyplněna

- *Cíl:* Zkontrolovat, zda databáze povolí nevyplněná pole.
- *URL:* <http://108.143.193.45:8080/api/v1/students/268>
- *Metoda:* GET
- *Vstupní data:* id existujícího studenta (268)
- *Kroky:*
 1. Zkopírovat ID studenta a otestovat v Postman metodou GET
 2. Odeslat požadavek
 3. Zkontrolovat údaje
- *Očekávaný výsledek:* Všechna pole jsou vyplněna.
- *Status:* fail

Testovací scénář 02: Do textových polí nelze napsat číslo

- *Cíl:* Zkontrolovat, zda je možné do textových polí napsat číslo.
- *URL:* <http://108.143.193.45:8080/api/v1/students/271>
- *Metoda:* GET
- *Vstupní data:* id existujícího studenta (271)
- *Kroky:*
 1. Zkopírovat ID studenta a otestovat v Postman metodou GET
 2. Odeslat požadavek
 3. Zkontrolovat údaje
- *Očekávaný výsledek:* Do textových polí je možné napsat pouze text (ne čísla).
- *Status:* fail

Testovací scénář 03: Do pole „věk“ nelze napsat čísla mimo rozmezí

- *Cíl:* Zkontrolovat, zda je možné do pole „věk“ napsat čísla mimo určené rozmezí.
- *URL:* <http://108.143.193.45:8080/api/v1/students/283>
- *Metoda:* GET
- *Vstupní data:* id existujícího studenta (283)
- *Kroky:*
 1. Zkopírovat ID studenta a otestovat v Postman metodou GET
 2. Odeslat požadavek
 3. Zkontrolovat hodnoty v poli „věk“

- *Očekávaný výsledek:* Do pole „věk“ je možné napsat číslo v rozmezí 6–65 (např., nevím, jaké jsou hraniční hodnoty).
- *Status:* fail

Testovací scénář 04: Do pole „e-mail“ nelze zapsat e-mail v neplatném znění

- *Cíl:* Zkontrolovat, zda je možné do pole „e-mail“ zapsat e-mail v neplatném znění.
- *URL:* <http://108.143.193.45:8080/api/v1/students/283>
- *Metoda:* GET
- *Vstupní data:* id existujícího studenta (283)
- *Kroky:*
 1. Zkopírovat ID studenta a otestovat v Postman metodou GET
 2. Odeslat požadavek
 3. Zkontrolovat zda, je e-mail zapsán správně
- *Očekávaný výsledek:* Do pole „e-mail“ je možné napsat pouze platnou e-mailovou doménu a ve správném znění (záleží, co je určeno, ale např. @gmail.com, @seznam.cz apod.).
- *Status:* fail

Testovací scénář 05: Do polí nelze psát speciální znaky

- *Cíl:* Zkontrolovat, zda je možné do polí vepsat speciální znaky.
- *URL:* <http://108.143.193.45:8080/api/v1/students/306>
- *Metoda:* GET
- *Vstupní data:* id existujícího studenta (306)
- *Kroky:*
 1. Zkopírovat ID studenta a otestovat v Postman metodou GET
 2. Odeslat požadavek
 3. Zkontrolovat pole
- *Očekávaný výsledek:* Do pole nelze vepsat speciální znaky.
- *Status:* fail

Testovací scénář 06: Záznamy se nemohou opakovat

- *Cíl:* Zkontrolovat, zda je možné, že se některé záznamy budou opakovat.
- *URL:* <http://108.143.193.45:8080/api/v1/students/>
- *Metoda:* exploratory
- *Vstupní data:* databáze, příjmení Vdaný
- *Kroky:*
 1. Projít tabulku
 2. Zkontrolovat, zda se záznamy neopakují
 3. Pomocí kódu v SQL vypsat všechny, kteří mají stejné příjmení (např. Vdaný)

- *Očekávaný výsledek:* Není možné mít stejný záznam více než jednou.
- *Status:* fail

Testovací scénář 07: V tabulce nejsou žádná testovací data

- *Cíl:* Zkontrolovat, zda po testování nezbyla žádná testovací data.
- *URL:* <http://108.143.193.45:8080/api/v1/students/>
- *Metoda:* exploratory
- *Vstupní data:* databáze
- *Kroky:*
 1. Projít tabulku
 2. Zkontrolovat, zda nezůstala v tabulce nějaká testovací data (jména Test apod.)
- *Očekávaný výsledek:* Všechna testovací data byla smazána.
- *Status:* fail

Testovací scénář 08: Tabulka neobsahuje žádné HTML kódy

- *Cíl:* Zkontrolovat, zda v tabulce nejsou žádné HTML kódy.
- *URL:* <http://108.143.193.45:8080/api/v1/students/>
- *Metoda:* exploratory
- *Vstupní data:* databáze
- *Kroky:*
 1. Projít tabulku
 2. Zkontrolovat, zda nezůstaly v tabulce nějaké kódy
- *Očekávaný výsledek:* Tabulka obsahuje pouze slova a čísla.
- *Status:* fail

Testovací scénář 09: Do pole je možné napsat pouze určené množství znaků

- *Cíl:* Zkontrolovat, zda mají pole nějaké maximum v počtu znaků.
- *URL:* <http://108.143.193.45:8080/api/v1/students/991>
- *Metoda:* GET
- *Vstupní data:* id existujícího studenta (991)
- *Kroky:*
 1. Zkopírovat ID studenta a otestovat v Postman metodou GET
 2. Odeslat požadavek
 3. Zkontrolovat zda, text převyšuje počet stanovených maximálních znaků
- *Očekávaný výsledek:* Pole nedovolí vepsat text o více znacích (než např. 15).
- *Status:* fail

Testovací scénář 10: Data z tabulky lze smazat

- *Cíl:* Zkontrolovat, zda je možné data z tabulky smazat..
- *URL:* <http://108.143.193.45:8080/api/v1/students/991>
- *Metoda:* DELETE

- *Vstupní data:* id existujícího studenta (991)
- *Kroky:*
 1. Zkopírovat ID studenta a smazat v Postman metodou DELETE
 2. Odeslat požadavek
- *Očekávaný výsledek:* Data jsou smazána.
- *Status:* pass

Testovací scénář 11: Lze založit záznam o studentovi

- *Cíl:* Zkontrolovat, zda je možné založit záznam o studentovi.
- *URL:* <http://108.143.193.45:8080/api/v1/students/1007>
- *Metoda:* POST
- *Vstupní data:* id existujícího studenta (1007)
- *Kroky:*
 1. Zkopírovat ID studenta a založit záznam o studentovi v Postman metodou POST
 2. Odeslat požadavek
- *Očekávaný výsledek:* Záznam je založen.
- *Status:* fail

EXEKUCE TESTŮ

Testovací scénáře jsem provedla, přikládám výsledky testů.

ID 01:

GET

http://108.143.193.45:8080/api/v1/students/268

ParamsAuthorizationHeaders (6)BodyPre-request ScriptTestsSettings

Query Params

	Key	Value
	Key	Value

BodyCookiesHeaders (5)Test Results200 OK161 ms233 B

PrettyRawPreviewVisualizeJSON

```
1  {
2    "id": 268,
3    "firstName": "John",
4    "lastName": "JOSHUA",
5    "email": "",
6    "age": 34
7  }
```

ID 02:

GET

http://108.143.193.45:8080/api/v1/students/271

ParamsAuthorizationHeaders (6)BodyPre-request ScriptTestsSettings

Query Params

	Key	Value
	Key	Value

BodyCookiesHeaders (5)Test Results200 OK1255 ms236 B

PrettyRawPreviewVisualizeJSON

```
1  {
2    "id": 271,
3    "firstName": "John",
4    "lastName": "JOSHUA",
5    "email": "548",
6    "age": 34
7  }
```

ID 03:

GET

http://108.143.193.45:8080/api/v1/students/283

ParamsAuthorizationHeaders (6)BodyPre-request ScriptTestsSettings

Query Params

	Key	Value
	Key	Value

BodyCookiesHeaders (5)Test Results200 OK241 ms260 B

PrettyRawPreviewVisualizeJSON

```
1  {
2    "id": 283,
3    "firstName": "PoPoLUska",
4    "lastName": "ČRIEVIČKOVÁ",
5    "email": "poLusa@gmail.",
6    "age": 170
7  }
```


ID 04:

GET

▼

http://108.143.193.45:8080/api/v1/students/283

Params

Authorization

Headers (6)

Body

Pre-request Script

Tests

Settings

Query Params

Key	Value
Key	Value

```
body  Cookies  Headers (5)  Test Results  200 OK  241 ms  260 B  JS
Pretty  Raw  Preview  Visualize  JSON  {
  1  {
  2    "id": 283,
  3    "firstName": "PoPoLUska",
  4    "lastName": "ČRIEVIČKOVÁ",
  5    "email": "polusa@gmail.",
  6    "age": 170
  7  }
```

ID 05:

GET

▼

http://108.143.193.45:8080/api/v1/students/306

Params

Authorization

Headers (6)

Body

Pre-request Script

Tests

Settings

Query Params

Key	Value
Key	Value

ID 06:

The screenshot shows a SQL query editor with a toolbar at the top containing icons for file operations, undo, redo, and search. The query text is:

```
1 • SELECT * FROM student
2 WHERE last_name = 'VDANY';
```

Below the query editor, there is a progress bar and a status bar showing "100%" and "22:1". The "Result Grid" tab is active, displaying a table with the following data:

id	age	email	first_name	last_name
4	19	ter4@yourmail.com	Peter	VDANY
6	19	ter4@yourmail.com	Peter	VDANY
6	19	ter4@yourmail.com	Peter	VDANY
7	19	ter4@yourmail.com	Peter	VDANY
NULL	NULL	NULL	NULL	NULL

ID 07:

378	34	jjohn34@yourmail.com	Tef	TESTSDF
379	34	jjohn34@yourmail.com	Tef	TESTSDF
382	34	jjohn34@yourmail.com	Tef	TESTSDF
392	34	jjohn34@yourmail.com	Tef	TESTSDF
393	34	jjohn34@yourmail.com	Tef	TESTSDF
394	34	jjohn34@yourmail.com	Tef	TESTSDF
395	22	petr@pokus.com	Petr	NOVAK
396	34	jjohn34@yourmail.com	Tef	TESTSDF
397	99	janko@yourmail.com	Janko	TEST
398	-10	john.test@com	James	ANDERSON
401	12	test12@21test.com	JT	NEW

ID 08:

547	37	<html><head><title></title></head>	<!DOCTYPE HTML>
-----	----	------------------------------------	-----------------

ID 09:

GET

http://108.143.193.45:8080/api/v1/students/991

Send

ParamsAuthorizationHeaders (6)BodyPre-request ScriptTestsSettingsCookies

Query Params

Key	Value	Bulk Edit
Key	Value	

BodyCookiesHeaders (5)Test Results200 OK252 ms398 BSave Response

PrettyRawPreviewVisualizeJSON

```
1{"id": 991,
2  "firstName":
3    "nesmyslnedlouhejmenokterebynemelobytulozeneblablablablablablablablablablablbl
4    "lastName": "VELMIDLOUHEJMENO",
5    "email": "dlouhejmeno@gmail.cz",
6    "age": 20
}
```

ID 10:

DELETE

http://108.143.193.45:8080/api/v1/students/991

Send

ParamsAuthorizationHeaders (6)BodyPre-request ScriptTestsSettingsCookies

Query Params

Key	Value	Bulk Edit
Key	Value	

BodyCookiesHeaders (4)Test Results200 OK258 ms123 BSave Response

ID 11:

POST

http://108.143.193.45:8080/api/v1/students/1007

Send

ParamsAuthorizationHeaders (7)BodyPre-request ScriptTestsSettingsCookies

Query Params

Key	Value	Bulk Edit
Key	Value	

BodyCookiesHeaders (6)Test Results405 Method Not Allowed170 ms331 BSave Response

PrettyRawPreviewVisualizeJSON

```
1{"timestamp": "2024-08-11T16:49:53.912+08:00",
2  "status": 405,
3  "error": "Method Not Allowed",
4  "message": "",
5  "path": "/api/v1/students/1007"
}
```

BUG REPORT

Na základě provedených scénářů jsem objevila uvedené chyby aplikace.

Testovací scénář 01: Všechna pole musí být vyplněna

- *Cíl:* Zkontrolovat, zda databáze povolí nevyplněná pole.
- *URL:* `http://108.143.193.45:8080/api/v1/students/268`
- *Metoda:* GET
- *Vstupní data:* id existujícího studenta (268)
- *Kroky:*
 4. Zkopírovat ID studenta a otestovat v Postman metodou GET
 5. Odeslat požadavek
 6. Zkontrolovat údaje
- *Očekávaný výsledek:* Všechna pole jsou vyplněna.
- *Status:* fail
- *Reálný výsledek:* Pole „e-mail” je prázdné.

Testovací scénář 02: Do textových polí nelze napsat číslo

- *Cíl:* Zkontrolovat, zda je možné do textových polí napsat číslo.
- *URL:* `http://108.143.193.45:8080/api/v1/students/271`
- *Metoda:* GET
- *Vstupní data:* id existujícího studenta (271)
- *Kroky:*
 1. Zkopírovat ID studenta a otestovat v Postman metodou GET
 2. Odeslat požadavek
 3. Zkontrolovat údaje
- *Očekávaný výsledek:* Do textových polí je možné napsat pouze text (ne čísla).
- *Status:* fail
- *Reálný výsledek:* Do pole „e-mail” lze napsat číslo.

Testovací scénář 03: Do pole „věk” nelze napsat čísla mimo rozmezí

- *Cíl:* Zkontrolovat, zda je možné do pole „věk” napsat čísla mimo určené rozmezí.
- *URL:* `http://108.143.193.45:8080/api/v1/students/283`
- *Metoda:* GET
- *Vstupní data:* id existujícího studenta (283)
- *Kroky:*
 1. Zkopírovat ID studenta a otestovat v Postman metodou GET
 2. Odeslat požadavek
 3. Zkontrolovat hodnoty v poli „věk”
- *Očekávaný výsledek:* Do pole „věk” je možné napsat číslo v rozmezí 6–65 (např., nevím, jaké jsou hraniční hodnoty).

- *Status:* fail
- *Reálný výsledek:* Do pole „věk“ lze napsat číslo mimo rozmezí.

Testovací scénář 04: Do pole „e-mail“ nelze zapsat e-mail v neplatném znění

- *Cíl:* Zkontrolovat, zda je možné do pole „e-mail“ zapsat e-mail v neplatném znění.
- *URL:* <http://108.143.193.45:8080/api/v1/students/283>
- *Metoda:* GET
- *Vstupní data:* id existujícího studenta (283)
- *Kroky:*
 1. Zkopírovat ID studenta a otestovat v Postman metodou GET
 2. Odeslat požadavek
 3. Zkontrolovat zda, je e-mail zapsán správně
- *Očekávaný výsledek:* Do pole „e-mail“ je možné napsat pouze platnou e-mailovou doménu a ve správném znění (záleží, co je určeno, ale např. @gmail.com, @seznam.cz apod.).
- *Status:* fail
- *Reálný výsledek:* Do pole „e-mail“ lze napsat neplatná adresa – např. @gmail.

Testovací scénář 05: Do polí nelze psát speciální znaky

- *Cíl:* Zkontrolovat, zda je možné do polí vepsat speciální znaky.
- *URL:* <http://108.143.193.45:8080/api/v1/students/306>
- *Metoda:* GET
- *Vstupní data:* id existujícího studenta (306)
- *Kroky:*
 1. Zkopírovat ID studenta a otestovat v Postman metodou GET
 2. Odeslat požadavek
 3. Zkontrolovat pole
- *Očekávaný výsledek:* Do pole nelze vepsat speciální znaky.
- *Status:* fail
- *Reálný výsledek:* Do všech polí lze psát speciální znaky.

Testovací scénář 06: Záznamy se nemohou opakovat

- *Cíl:* Zkontrolovat, zda je možné, že se některé záznamy budou opakovat.
- *URL:* <http://108.143.193.45:8080/api/v1/students/>
- *Metoda:* exploratory
- *Vstupní data:* databáze, příjmení Vdaný
- *Kroky:*
 1. Projít tabulku
 2. Zkontrolovat, zda se záznamy neopakují
 3. Pomocí kódu v SQL vypsat všechny, kteří mají stejné příjmení (např. Vdaný)

- *Očekávaný výsledek:* Není možné mít stejný záznam více než jednou.
- *Status:* fail
- *Reálný výsledek:* Tabulka obsahuje testovací data.

Testovací scénář 08: Tabulka neobsahuje žádné HTML kódy

- *Cíl:* Zkontrolovat, zda v tabulce nejsou žádné HTML kódy.
- *URL:* <http://108.143.193.45:8080/api/v1/students/>
- *Metoda:* exploratory
- *Vstupní data:* databáze
- *Kroky:*
 1. Projít tabulku
 2. Zkontrolovat, zda nezůstaly v tabulce nějaké kódy
- *Očekávaný výsledek:* Tabulka obsahuje pouze slova a čísla.
- *Status:* fail
- *Reálný výsledek:* V tabulce jsou HTML kódy.

Testovací scénář 09: Do pole je možné napsat pouze určené množství znaků

- *Cíl:* Zkontrolovat, zda mají pole nějaké maximum v počtu znaků.
- *URL:* <http://108.143.193.45:8080/api/v1/students/991>
- *Metoda:* GET
- *Vstupní data:* id existujícího studenta (991)
- *Kroky:*
 1. Zkopírovat ID studenta a otestovat v Postman metodou GET
 2. Odeslat požadavek
 3. Zkontrolovat zda, text převyšuje počet stanovených maximálních znaků
- *Očekávaný výsledek:* Pole nedovolí vepsat text o více znacích (než např. 15).
- *Status:* fail
- *Reálný výsledek:* Do pole lze napsat libovolný počet znaků.

Testovací scénář 11: Lze založit záznam o studentovi

- *Cíl:* Zkontrolovat, zda je možné založit záznam o studentovi.
- *URL:* <http://108.143.193.45:8080/api/v1/students/1007>
- *Metoda:* POST
- *Vstupní data:* id existujícího studenta (1007)
- *Kroky:*
 1. Zkopírovat ID studenta a založit záznam o studentovi v Postman metodou POST
 2. Odeslat požadavek
- *Očekávaný výsledek:* Záznam je založen.
- *Status:* fail
- *Reálný výsledek:* 405 Method Not Allowed