

Warsztaty z testowania backendu za pomocą narzędzi SoapUI, Postman, JMeter.

SoapUI – Odpowiedzi

Legenda

PPM – prawy przycisk myszki, **LPM** – lewy przycisk myszki

Zadanie 4 - Pobierz listę ras

1. Kliknij na **“Norad Test Suite”**
2. PPM -> New Test Case
3. Podaj nazwę **“Exercise 4 – Get all races”**
4. Kliknij PPM na **“Test Steps (0)”** -> Add step -> REST Request
5. Wprowadź nazwę **“Get all races”**
6. Wybierz metodę restową, która będzie pobierać wszystkie rasy
/dictionaries/races -> getRacesUsingGET -> Request 1
7. Zadanie wykonane.
8. Sprawdź działanie klikając w zielony trójkącik.
9. Sprawdź wyniki w prawej części okienka – boczna zakładka **“JSON”**

Podpunkt A)

A.1) Kliknij w zielony plus aby dodać asercję

A.2) Wybierz **“Compliance, Status and Standards”** -> **“Valid HTTP Status Codes”**

A.3) Kliknij w przycisk **“Add”**

A.4) Wpisz 200

200 to http code określający poprawne wykonanie requesta po stronie backendu.

Hint: <https://jsonpath.com/> - strona wspierająca zrozumienie działania JsonPath

Podpunkt B)

B.1) Kliknij w zielony plus aby dodać asercję

B.2) Wybierz **“Property Content”** -> **“JsonPath Count”**

B.3) W polu **“JSONPath Expression”** wpisz **\$**

B.4) W polu **“Expected Result”** wprowadź wartość oczekiwaną **10**

- Hint: **“Select from current”** - pobiera wartość z ostatniego wyniku wywołania requesta

Podpunkt C)

- C.1) Kliknij w zielony plus aby dodać asercję
- C.2) Wybierz **“Property Content”** -> **“JsonPath Match”**
- C.3) W polu **“JSONPath Expression”** wpisz **`$(4).name`**
- C.4) W polu **“Expected Result”** wprowadź wartość oczekiwaną **Kot egzotyczny**
 - **“Select from current”** - pobiera wartość z ostatniego wyniku wywołania requesta

Zadanie 5 - Zarejestruj nowego użytkownika

1. Kliknij na **“Norad Test Suite”**
2. PPM -> New Test Case
3. Podaj nazwę **“Exercise 5 – Register new user”**
4. Kliknij PPM na **“Test Steps (0)”** -> Add step -> REST Request
5. Wprowadź nazwę **“Register user”**
6. Wybierz metodę restową, która tworzy nowego użytkownika
“/accounts->registerUsingPOST->Request 1”
7. Określ zawartość POSTa zgodnie z dokumentacją w Swaggerze dla endpointu, który wykonuje utworzenie nowego użytkownika: <http://51.38.129.181:8100/norad/swagger-ui.html#/account-endpoint/registerUsingPOST>

Podpunkt A)

- A.1) PPM na **“Exercise 5 – Register new User -> Test Steps (1)”**
- A.2) Wybierz **“Add Step -> Groovy Script”**, określ nazwę **“Generate unique email”**
- A.3) Upewnij się, że skrypt **“Generate unique email”** jest wykonany jako pierwszy
- A.4) Umieść zawartość skryptu

/postman-soapui-jmeter-qa-training/exercises-answers/soapui/ex-05-A-generate-random-email.groovy

Zamiast **“mszpiler”** wprowadź swoją pierwszą listę imienia i nazwisko.

- A.5) Zmień zawartość contentu wysyłanego w requesćie, pole email w JSONIE zmień na wartość **`#{TestCase#email}`**

Przykład znajdziesz w pliku:

/postman-soapui-jmeter-qa-training/exercises-answers/soapui/json/register-new-user.json

Podpunkt B)

B.1) Kliknij w zielony plusik

B.2) Wybierz **“Property Content”** -> **“JsonPath Match”**

B.3) W **“JSONPath expression”** wprowadź **\$.email**

B.4) W **“Expected result”** wprowadź **\${#TestCase#email}**

B.5) Żeby sprawdzić czy asercja działa, to dwukrotnie kliknij w nazwę test case **“Exercise 5 – Register new user”**, kliknij w zielony trójkącik, jeżeli w dolnym oknie żaden z kroków nie zapalił lampki na czerwono to test przebiegł pozytywnie.

Podpunkt C)

C.1) Kliknij w zielony plusik

C.2) Wybierz **“Script”** -> **“Script Assertion”**

C.3) Ustal nazwę asercji na **“Database verification”**

C.3) Umieść zawartość skryptu

/postman-soapui-jmeter-qa-training/exercises-answers/soapui/ex-05-C-check-database.groovy

B.4) Kliknij OK

Podpunkt D)

1. PPM na **“Exercise 5 – Register new User -> Test Steps (1)”**
2. Wybierz **“Add Step -> Groovy Script”**, określ nazwę **“Save user UUID in properties”**
3. Umieść zawartość skryptu

/postman-soapui-jmeter-qa-training/exercises-answers/soapui/ex-05-D-save-user-uuid-in-properties.groovy

Zadanie 6 - Dodaj nowe zwierzę do nowego użytkownika

1. Kliknij na **“Norad Test Suite”**
2. PPM -> New Test Case
3. Podaj nazwę **“Exercise 6 - Add new animal to new user”**

Krok 1.

1. Kliknij PPM na **“Test Steps (0)”** -> Add step -> Run TestCase
2. Podaj nazwę **“Run TestCase Register new user”**

3. W "Target TestCase" wybierz "Exercise 5 – Register new user"
4. Kliknij OK.

Krok 2.

1. Kliknij PPM na "Test Steps (1)" -> "Add Step" -> "Groovy Script"
2. Podaj nazwę "Generate random animal name"
3. Umieść zawartość pliku:

/postman-soapui-jmeter-qa-training/exercises-answers/soapui/ex-06-step-02-random-animal-name.groovy

Krok 3.

1. Kliknij PPM na "Test Steps (2)" -> "Add Step" -> "REST Request"
2. Podaj nazwę "Get races"
3. Wybierz request

/dictionaries/races/ -> getRacesUsingGET -> Request 1

4. Kliknij PPM na "Test Steps (3)" -> "Add Step" -> "Groovy Script"
5. Podaj nazwę "Fetch random race"
6. Umieść zawartość pliku:

/postman-soapui-jmeter-qa-training/exercises-answers/soapui/ex-06-step-03-fetch-random-race.groovy

Krok 4.

1. Kliknij PPM na "Test Steps (4)" -> "Add Step" -> "REST Request"
2. Podaj nazwę "Create new animal"
3. Wybierz request:

/animals -> createUsingPOST -> Request 1

4. W lewym dolnym oknie należy wkleić zawartość body requesta, przykład body znajduje się dokumentacji Swagger - <http://51.38.129.181:8100/norad/swagger-ui.html#/animal-endpoint/createUsingPOST>
5. Body jest tylko przykładem, należy sparametryzować pola **name**, **racelId**, **userUUID** w następujący sposób - odpowiednio **\${#TestCase#animalName}**, **\${#TestCase#racelId}**, **\${#TestSuite#userUUID}** zmienne te ustalone są w poprzednich krokach aktualnego TestCase'a.

Zauważ, że userUUID jest pobierane z kontekstu TestSuite a nie z TestCase.

6. Przykład body docelowego znajdziesz w pliku:

/postman-soapui-jmeter-qa-training/exercises-answers/soapui/json/create-new-animal.json

Krok 5.

1. Kliknij PPM na "Test Steps (5)" -> "Add Step" -> "Groovy Script"
2. Podaj nazwę "Save animal UUID in properties".
3. Umieść zawartość skryptu

/postman-soapui-jmeter-qa-training/exercises-answers/soapui/ex-06-step-05-save-animal-uuid-in-properties.groovy

Asercja A

1. Kliknij dwukrotnie na step **"Create new animal"**.
2. Kliknij w zielony krzyżyk aby dodać asercję.
3. Wybierz **"Script" -> "Script Assertion"**
4. Zmień nazwę asercji na **"Assert A – expected race"**
5. Umieść zawartość pliku:

/postman-soapui-jmeter-qa-training/exercises-answers/soapui/ex-06-assert-A-expected-race.groovy

Asercja B

1. Kliknij w zielony krzyżyk aby dodać asercję.
2. Wybierz **"Script" -> "Script Assertion"**
3. Zmień nazwę asercji na **"Assert B – expected user"**
4. Umieść zawartość pliku:

/postman-soapui-jmeter-qa-training/exercises-answers/soapui/ex-06-assert-B-expected-user.groovy

Asercja C

1. Kliknij w zielony krzyżyk aby dodać asercję.
2. Wybierz **"Property Content" -> "JsonPath Count"**
1. Zmień nazwę asercji na **"Assert C - expected images count"**
3. W okienku **"JSONPath Expression"** wprowadź tekst: **\$.images**
4. W okienku **"Expected result"** wprowadź wartość: **3**
5. Kliknij w tekst **"Test"** aby sprawdzić czy

Asercja D

1. Kliknij w zielony krzyżyk aby dodać asercję.
2. Wybierz **"Script" -> "Script Assertion"**
3. Zmień nazwę asercji na **"Assert D – uuids exists in images"**
4. Umieść zawartość pliku:

/postman-soapui-jmeter-qa-training/exercises-answers/soapui/ex-06-assert-D-uuid-exists-in-images.groovy

Zadanie 7 – Wykonaj upload zdjęcia - załącznik wskazany ręcznie

1. Kliknij na **"Norad Test Suite"**
2. PPM -> New Test Case
3. Podaj nazwę **"Exercise 7 – Upload file"**

Krok 1.

1. Kliknij PPM na **"Test Steps (0)"** -> **"Add Step"** -> **"REST Request"**
2. Podaj nazwę **"Upload file"**
3. Wybierz request:

/animal/{animalUUID}/images/{imageUUID} -> uploadImageUsingPOST -> Request 1

Krok 2.

1. Wróć do TestCase'u **"Exercise 6 - Add new animal to new user"**
2. Na zadaniu numer 6 kliknij PPM na **"Test Steps (5)"** -> **"Add Step"** -> **"Groovy Script"**
3. Podaj nazwę **"Save first image UUID in properties"**.
4. Umieść zawartość skryptu

/postman-soapui-jmeter-qa-training/exercises-answers/soapui/ex-07-step-02-save-first-image-uuid-in-properties.groovy

5. Wróć do TestCase'u **"Exercise 7 - upload file"**
6. Kliknij PPM na **"Test Steps (1)"** -> **"Add Step"** -> **"Groovy Script"**
7. Podaj nazwę **"Setup upload request properties"**
8. Umieść zawartość skryptu:

/postman-soapui-jmeter-qa-training/exercises-answers/soapui/ex-07-step-02-setup-upload-request-properties.groovy

9. Ustal poprawną kolejność stepów w TestCase – pierwszy step to **"Setup upload request properties"**, drugi step to **"Upload file"**.

Krok 3.

1. Kliknij w zakładkę **"Attachments"**, pojawi się okienko z listą załączników, na początku lista jest pusta.
2. Kliknij w zielony krzyżyk w okienku z załącznikami.
3. Wybierz plik ze zdjęciem dostępny na dysku.
4. W górnym okienku zawierającym listę parametrów **animalUUID**, **imageUUID** oraz **file** należy określić ich zawartość. Parametr **file** przyjmuje wartość **file:<nazwa_pliku>** gdzie **<nazwa_pliku>** to nazwa pliku ze zdjęciem, które wybrałeś na swoim dysku.

Asercja A

1. Na zaznaczonym stepie **"Upload file"** kliknij w zielony krzyżyk aby dodać asercję.
2. Wybierz **"Compliance, Status and Standards"** -> **"Valid Http Status Codes"**
3. Wprowadź wartość **200**

Asercja B

1. Należy przygotować parametry do wywołania requestu pobierającego zawartość pliku z serwera.
2. Kliknij PPM na **"Test Steps (2)"** -> **"Add Step"** -> **"Groovy Script"**
3. Podaj nazwę **"Setup download request properties"**
4. Umieść zawartość skryptu:

/postman-soapui-jmeter-qa-training/exercises-answers/soapui/ex-07-assert-B-setup-download-request-properties.groovy

Skrypt nie będzie od razu działał, ponieważ w skrypcie następuje odwołanie do kroku, który będzie utworzony poniżej.

5. Kliknij PPM na **“Test Steps (3)”** -> **“Add Step”** -> **“REST Request”**
6. Podaj nazwę **“Download file”**
7. Wybierz request:

/animal/{animalUUID}/images/{imageUUID} -> downloadImageUsingGET -> Request 1

8. W okienku kroku **“Download file”** kliknij w zielony krzyżyk aby dodać asercję.
9. Wybierz **“Script”** -> **“Script Assertion”**
10. Umieść zawartość pliku:

/postman-soapui-jmeter-qa-training/exercises-answers/soapui/ex-07-assert-B-validate-filesize.groovy

11. Na liście asercji zmień nazwę nowo utworzonej asercji na **“Validate file size”**.

Zadanie 8 – Wykonaj upload zdjęcia - załącznik wskazany za pomocą skryptu Groovy

1. Kliknij na **“Norad Test Suite”**
2. PPM -> New Test Case
3. Podaj nazwę **“Exercise 8 – Upload file - Groovy”**

Krok 1.

1. Kliknij PPM na **“Test Steps (0)”** -> **“Add Step”** -> **“REST Request”**
2. Podaj nazwę **“Upload file”**
3. Wybierz request:

/animal/{animalUUID}/images/{imageUUID} -> uploadImageUsingPOST -> Request 1

Krok 2.

1. Kliknij PPM na **“Test Steps (0)”** -> **“Add Step”** -> **“Groovy Script”**
2. Podaj nazwę **“Setup upload request properties”**
3. Umieść zawartość skryptu:

/postman-soapui-jmeter-qa-training/exercises-answers/soapui/ex-08-step-02-setup-upload-request-properties.groovy

4. Ustal poprawną kolejność stepów w TestCase – pierwszy step to **“Setup upload request properties”**, drugi step to **“Upload file”**.

Krok 3.

1. Kliknij na nazwę projektu w SoapUI.
2. Przejdź na zakładkę **“Custom Properties”**
3. Za pomocą zielonego krzyżyka dodaj dwa properties:

- ***imageFilePath***
 - ***imageFileName***
4. Wartości to ścieżka i nazwa do pliku ze zdjęciem na Twoim komputerze.

Krok 4.

1. Kliknij PPM na **"Test Steps (2)"** -> **"Add Step"** -> **"Groovy Script"**
4. Podaj nazwę **"Setup attachment"**
5. Umieść zawartość skryptu:

/postman-soapui-jmeter-qa-training/exercises-answers/soapui/ex-08-step-04-setup-attachment.groovy

6. Zweryfikuj czy kolejność kroków w TestCase jest poprawna. Kolejność jaka powinna obowiązywać to:
 - a. Setup upload request properties
 - b. Setup attachment
 - c. Upload file

Asercja A

1. Na zaznaczonym stepie **"Upload file"** kliknij w zielony krzyżyk aby dodać asercję.
2. Wybierz **"Compliance, Status and Standards"** -> **"Valid Http Status Codes"**
3. Wprowadź wartość **200**

Asercja B

1. Należy przygotować parametry do wywołania requestu pobierającego zawartość pliku z serwera.
2. Kliknij PPM na **"Test Steps (3)"** -> **"Add Step"** -> **"Groovy Script"**
3. Podaj nazwę **"Setup download request properties"**
4. Umieść zawartość skryptu:

/postman-soapui-jmeter-qa-training/exercises-answers/soapui/ex-08-assert-B-setup-download-request-properties.groovy

Skrypt nie będzie od razu działał, ponieważ w skrypcie następuje odwołanie do kroku, który będzie utworzony poniżej.

5. Kliknij PPM na **"Test Steps (4)"** -> **"Add Step"** -> **"REST Request"**
6. Podaj nazwę **"Download file"**
7. Wybierz request:

/animal/{animalUUID}/images/{imageUUID} -> downloadImageUsingGET -> Request 1

8. W okienku kroku **"Download file"** kliknij w zielony krzyżyk aby dodać asercję.
9. Wybierz **"Script"** -> **"Script Assertion"**
10. Umieść zawartość pliku:

/postman-soapui-jmeter-qa-training/exercises-answers/soapui/ex-08-assert-B-validate-filesize.groovy

11. Na liście asercji zmień nazwę nowo utworzonej asercji na **"Validate file size"**.

Zadanie 9 Dodaj zwierzęta pobierając dane z pliku CSV

1. Kliknij na **"Norad Test Suite"**
2. PPM -> New Test Case
3. Podaj nazwę **"Exercise 9 – Add many animals from CSV file"**

Krok 1.

1. Kliknij PPM na **"Test Steps (0)" -> "Add Step" -> "Run TestCase"**
2. Podaj nazwę: **"Run TestCase Register User"**
3. Wybierz z listy TestCase: **"Exercise 5 – Register new user"**

Krok 2.

1. Kliknij PPM na **"Test Steps (1)" -> "Add Step" -> "REST Request"**
2. Podaj nazwę: **"Create one animal"**.
3. Wybierz request:

/animals -> createUsingPOST -> Request 1

4. W lewym dolnym okienku należy określić zawartość body requesta tworzącego zwierzę
5. Przykład body znajdziesz w pliku:

/postman-soapui-jmeter-qa-training/exercises-answers/soapui/json/create-new-animal.json

6. Należy step **"Create one animal"** ustawić na **Disabled** ażeby domyślnie step nie był wykonywany przez SoapUI tylko na wyraźne wywołanie w skrypcie Groovy.

Krok 3.

1. Kliknij PPM na **"Test Steps (1)" -> "Add Step" -> "Groovy Script"**
2. Podaj nazwę: **"Loop CSV file"**
3. Umieść zawartość skryptu:

/postman-soapui-jmeter-qa-training/exercises-answers/soapui/ex-09-step-03-loop-csv-file.groovy

Asercja B:

1. Kliknij PPM na **"Test Steps (1)" -> "Add Step" -> "Groovy Script"**
2. Podaj nazwę: **"Validate animals in database"**
3. Umieść zawartość skryptu:

/postman-soapui-jmeter-qa-training/exercises-answers/soapui/ex-09-assert-B-validate-animals-in-database.groovy

Zadanie 10 Wiele kroków

1. Kliknij na **"Norad Test Suite"**

2. PPM -> New Test Case
3. Podaj nazwę **"Exercise 10 – Many steps"**

Krok 1.

1. Kliknij PPM na **"Test Steps (0)"** -> **"Add Step"** -> **"Run TestCase"**
2. Podaj nazwę: **"Run TestCase Register User"**
3. Wybierz z listy TestCase: **"Exercise 5 – Register new user"**

Krok 2.

1. Kliknij PPM na **"Test Steps (1)"** -> **"Add Step"** -> **"REST Request"**
2. Podaj nazwę: **"Create one animal"**.
3. Wybierz request:

/animals -> createUsingPOST -> Request 1

4. W lewym dolnym okienku należy określić zawartość body requesta tworzącego zwierzę
5. Przykład body znajdziesz w pliku:

/postman-soapui-jmeter-qa-training/exercises-answers/soapui/json/create-new-animal.json

6. Należy step **"Create one animal"** ustawić na **Disabled** ażeby domyślnie step nie był wykonywany przez SoapUI tylko na wyraźne wywołanie w skrypcie Groovy.

Krok 3.

1. Kliknij PPM na **"Test Steps (2)"** -> **"Add Step"** -> **"Groovy Script"**
2. Podaj nazwę: **"Loop CSV file"**
3. Umieść zawartość skryptu:

/postman-soapui-jmeter-qa-training/exercises-answers/soapui/ex-10-step-03-loop-csv-file.groovy

Krok 4.

1. Kliknij PPM na **"Test Steps (3)"** -> **"Add Step"** -> **"REST Request"**
2. Podaj nazwę **"Get animal details"**
3. Wybierz request:

/animals/{animalUUID} -> getAnimalUsingGET -> Request 1

4. Ustaw na **"Disabled"** krok **"Get animal details"** - krok będzie uruchamiany na wyraźne wywołanie w skrypcie Groovy.
5. Ustawienie wykonujemy PPM na stepie **"Get animal details"** -> **"Disable TestStep"**
6. Kliknij PPM na **"Test Steps (4)"** -> **"Add Step"** -> **"Groovy Script"**
7. Podaj nazwę: **"Validate animals using REST API"**
8. Umieść zawartość skryptu:

/postman-soapui-jmeter-qa-training/exercises-answers/soapui/ex-10-step-04-validate-animals-using-rest-api.groovy