Wprowadzenie do systemów rozproszonych Rest Client

Norbert Białas

Wydział Fizyki i Informatyki Stosowanej

Ćwiczenie 3

Rest Client

Ćwiczenie to polegało na zaimplementowaniu wywołań serwera po stronie klienta. Serwer był już zaimplementowany. Modyfikacji wymagała klasa wdsr.exercise3.client.*ProductService* a mianowicie metody w niej zawarte które miały spełaniać swoje role zgodnie z dokumentacją.Serwer pod testy był ustawiony pod adresem http://localhost:8091

REST styl architektury oprogramowania wykorzystuje m.in. jednorodny interfejs, bezstanową komunikację, zasoby, reprezentacje, hipermedia.

Należało uzupełnić 6 metod.

Serwer został uruchomiony i działa poprawienie:

```
Problems @ Javadoc Declaration Console Declaration Gradle Tasks Gradle Executions

Main (2) [Java Application] C:\Program Files\Java\jre1.8.0_131\bin\javaw.exe (28.09.2017, 22:44:13)

[INFO] 2017-09-28 22:44:14,628 [main] [] org.xnio <clinit> - XNIO NIO Implementation Version 3.3.4.Final

[INFO] 2017-09-28 22:44:15,226 [main] [] org.xnio.nio <clinit> - XNIO NIO Implementation Version 3.3.4.Final

[INFO] 2017-09-28 22:44:15,226 [main] [] org.jboss.resteasy.resteasy.jaxrs.i18n processApplication - RESTEASY002225: Deployin

[INFO] 2017-09-28 22:44:15,232 [main] [] org.jboss.resteasy.resteasy.jaxrs.i18n processApplication - RESTEASY002200: Adding c

[INFO] 2017-09-28 22:44:15,233 [main] [] org.jboss.resteasy.resteasy.jaxrs.i18n processApplication - RESTEASY002205: Adding p

[INFO] 2017-09-28 22:44:15,233 [main] [] org.jboss.resteasy.resteasy.jaxrs.i18n processApplication - RESTEASY002200: Adding c

[INFO] 2017-09-28 22:44:15,234 [main] [] org.jboss.resteasy.resteasy.jaxrs.i18n processApplication - RESTEASY002200: Adding p

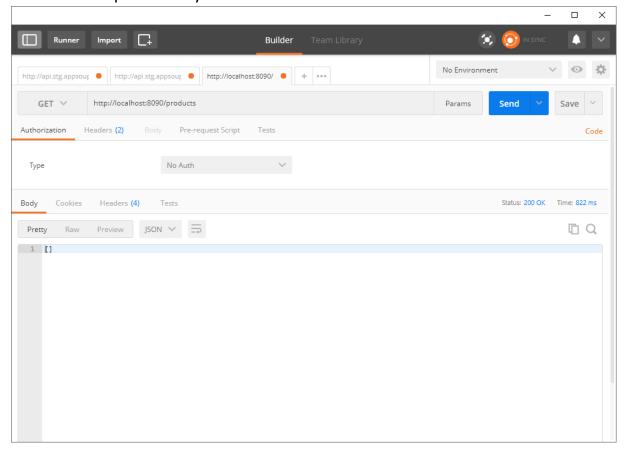
[INFO] 2017-09-28 22:44:15,235 [main] [] org.jboss.resteasy.resteasy.jaxrs.i18n processApplication - RESTEASY002200: Adding p

[INFO] 2017-09-28 22:44:15,236 [main] [] org.jboss.resteasy.resteasy.jaxrs.i18n processApplication - RESTEASY002200: Adding p

[INFO] 2017-09-28 22:44:15,239 [main] [] org.jboss.resteasy.resteasy.jaxrs.i18n processApplication - RESTEASY002200: Adding p

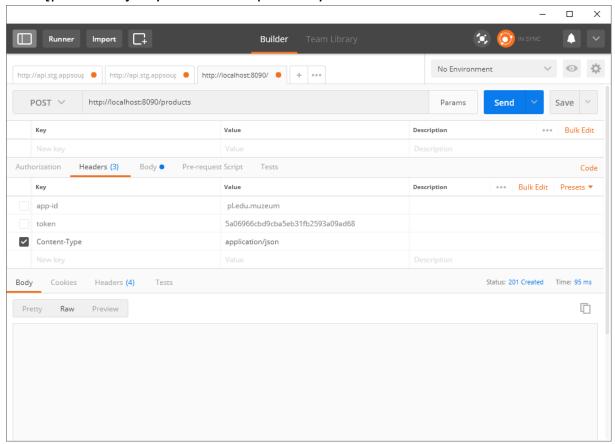
[INFO] 2017-09-28 22:44:15,239 [main] [] org.jboss.resteasy.resteasy.jaxrs.i18n processApplication - RESTEASY002200: Adding p
```

Postmanem sprawdzamy co zwraca serwer:

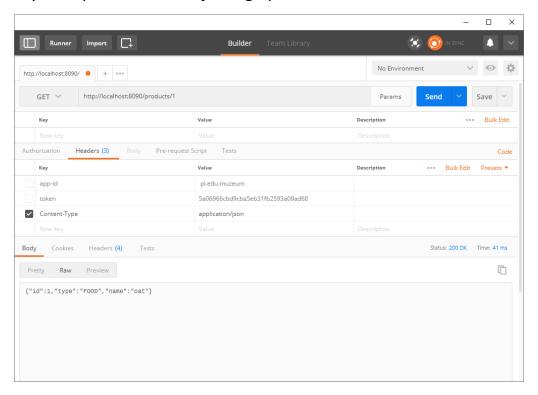


Jest to pusta tablica. Status 200 OK.

Następnie dodajemy do serwera produkty



Szybkie sprawdzenie dla jednego produktu:



W początkowej fazie testy zakończyły się błędami.

```
ProductService.java UndertowJaxrsServer.class Undertow.class ProductService.java
# Package Explorer ☑ JUnit ⊠
                                                       12
13
           public class ProductService extends RestClientBase {
Finished after 2.88 seconds
                                                               protected ProductService(final String serverHost, final int serverPort, final Client client) super(serverHost, serverPort, client);
                                                       15
16
 Runs: 15/15 

☐ Errors: 4 
☐ Failures: 8
   wdsr.exercise3.client.StatusCheckerTest [Runner: JUnit 4] (2,032 s)

* Looks up all p
                                                                * @param types Set of types to be looked up

* @return A list of found products - possibly empty, never null.
   wdsr.exercise3.client.ProductServiceTest [
      storeNewProduct_shouldCreateNewProduct (0,1
      retrieveProducts_shouldReturnEmptyListWhenN
                                                               public List<Product> retrieveProducts(Set<ProductType> types) {
      retrieveProduct_shouldThrowExceptionIfProduct
      retrieveProducts_shouldReturnProductListWhen
                                                                    return null;
      updateProduct_shouldUpdateProductIfFound (0
      률 retrieveAllProducts_shouldReturnEmptyListWhei
      retrieveAllProducts_shouldReturnProductListWh
                                                                 * Looks up all products known to the server.
      updateProduct shouldThrowExceptionIfProduct
                                                                 * @return A list of all products - possibly empty, never null.
      deleteProduct_shouldThrowExceptionIfProductN
```

Należało zaimplementować odpowiednie metody.

retrieveProducts:

Wyszukiwanie wszystkich produktów określonych typów znanych na serwerze.

retriveAllProducts:

Wyszukiwanie wszystkich produktów znanych na serwerze.

retrieveProduct:

Wyszukiwanie produktu poprzez jego ID na serwerze.

StoreNewProduct:

Stworzenie nowego produktu na serwerze.

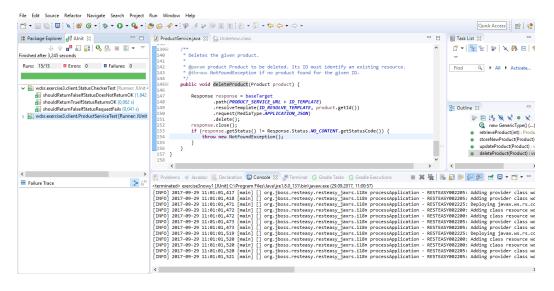
updateProduct:

Aktualizuje dany produkt.

deleteProduct

Usuwa dany produkt

Po utworzeniu ciała dla każdej metody w klasie ProductService.java należało przetestować czy działają one zgodnie z javadoc:



Jak widzimy w testach nie wystąpił żaden błąd.