



Testplan & testrapport Integration and Communication CODE TCIF-V2IAC1-15

HBO-ICT SIE Jaar 2 2016-2017

Naam: Wesley de Kraker

Studentnummer: 1675943

Datum
12 – 03 – 2017

Versie 1.0

© Hogeschool Utrecht, Utrecht, 2017

Inhoudsopgave

Inhoudsopgave	2
1. Introductie.....	3
2. Maven project	3
3. Testomgeving.....	3
4. Smoketesten	4
5. Unit testen.....	5
6. Integratietesten.....	6
7. Functionele test.....	6
8. Resultaten.....	7

1. Introductie

Ik heb voor de volgende opdracht gekozen: registreer windmolen (CRUD). De volgende services moeten voor deze opdracht aangeboden worden: create, read, update en delete. Ik heb ervoor gekozen om de windmolens in het geheugen op te slaan.

2. Maven project

Mijn Maven project heeft de volgende dependencies:

Group ID	Artifact ID	Version
junit	junit	4.12
javax.servlet	servlet-api	2.5
com.sun.xml.ws	jaxws-rt	2.2.10

Daarnaast maak mijn project gebruik van de volgende plugins:

Group ID	Artifact ID	Version
org.apache.tomcat.maven	tomcat7-maven-plugin	2.2
org.apache.maven.plugins	maven-compiler-plugin	3.6.1
org.jvnet.jax-ws-commons	jaxws-maven-plugin	2.1

Het project is getest op Apache Tomcat 7.0.75 met de volgende commando's:

- mvn clean install
- mvn tomcat7:deploy

3. Testomgeving

Ik ga alle services die ik gebouwd heb testen, dus create, read, update en delete. Deze services zijn erg afhankelijk van elkaar. Het is bijvoorbeeld alleen mogelijk om een windmolen te verwijderen als die eerst aangemaakt is.

Vandaar dat ik ervoor gekozen heb om unit testen te maken voor de foutafhandelingen en integratietesten voor de samenwerking tussen de verschillende onderdelen. Beide testen heb ik geautomatiseerd met JUnit. Daarnaast heb ik gebruik gemaakt van SoapUI om de functionele testen te automatiseren. Tot slot heb ik mijn browser gebruikt voor de smoketesten.

4. Smoketesten

Om te bepalen of mijn webservice succesvol gedeployed was heb ik telkens de volgende testen uitgevoerd:

Test	Middel
De homepagina werkt http://localhost:8080/windmillservice/ .	Browser
De SOAP-pagina werkt: http://localhost:8080/windmillservice/soap .	Browser
Een create request geeft een identificatienummer terug.	SoapUI



5. Unit testen

Om de foutafhandelingen te testen heb ik de volgende unit testen uitgevoerd:

Test	Middel
Er wordt een foutmelding gegeven als er geen windmolen gevonden wordt bij een <u>read request</u> .	JUnit
Er wordt een foutmelding gegeven als er geen windmolen gevonden wordt bij een <u>update request</u> .	JUnit
Er wordt een foutmelding gegeven als er geen windmolen gevonden wordt bij een <u>delete request</u> .	JUnit

Hieronder staat de code van deze testen:

```
@Test
public void read() throws Exception {
    try {
        checkIfWindmill1Exists();
        fail("De methode read geeft geen foutmelding als een object
niet gevonden is!");
    } catch (Fault e) { }
}

@Test
public void update() throws Exception {
    WindmillSuccess windmillSuccess = updateWindmill1ToWindmill2();
    assertTrue("De methode update geeft geen foutmelding als een
object niet gevonden is!",
        windmillSuccess.isSuccess() == false);
}

@Test
public void delete() throws Exception {
    WindmillSuccess windmillSuccess = deleteWindmill2();
    assertTrue("De methode delete geeft geen foutmelding als een
object niet gevonden is!!",
        windmillSuccess.isSuccess() == false);
}
```

De volledige code is te vinden in deze map:
windmillservice/src/test/java/nl/wesleydekraker/windmillservice

6. Integratietesten

Hieronder staan de integratietesten, de code hiervan staat in dezelfde map als de unit testen.

Test	Middel
Na een create request kan de windmolen ingelezen worden door een read request.	JUnit
Na een update request kan de bijgewerkte windmolen ingelezen worden met een read request.	JUnit
Na een delete request kan de windmolen niet meer opgehaald worden met een read request.	JUnit

7. Functionele test

De SOAP berichten van de functionele testen kunnen teruggevonden worden in de map: windmillservice/src/main/webapp/WEB-INF/soap. Daarnaast staat ook het SoapUI project in deze map.

Test	Middel
Een create request geeft een identificatienummer terug.	SoapUI
Een read request waar een string aan meegegeven wordt geeft een foutmelding.	SoapUI
Een read request van een windmolen die niet bestaat geeft een foutmelding terug.	SoapUI
Na een update request wordt er een positieve bevestiging teruggestuurd	SoapUI
Na een delete request wordt er een positieve bevestiging teruggestuurd	SoapUI

8. Resultaten

Zoals hieronder gezien kan worden zijn alle testen geslaagd.

Smoketesten

Test	Middel	
De homepagina werkt http://localhost:8080/windmillservice/ .	Browser	✓
De SOAP-pagina werkt: http://localhost:8080/windmillservice/soap .	Browser	✓
Een create request geeft een identificatienummer terug.	SoapUI	✓

Unit testen

Test	Middel	
Er wordt een foutmelding gegeven als er geen windmolen gevonden wordt bij een <u>read request</u> .	JUnit	✓
Er wordt een foutmelding gegeven als er geen windmolen gevonden wordt bij een <u>update request</u> .	JUnit	✓
Er wordt een foutmelding gegeven als er geen windmolen gevonden wordt bij een <u>delete request</u> .	JUnit	✓

Integratietesten

Test	Middel	
Na een create request kan de windmolen ingelezen worden door een read request.	JUnit	✓
Na een update request kan de bijgewerkte windmolen ingelezen worden met een read request.	JUnit	✓
Na een delete request kan de windmolen niet meer opgehaald worden met een read request.	JUnit	✓

Functionele testen

Test	Middel	
Een create request geeft een identificatienummer terug.	SoapUI	✓
Een read request waar een string aan meegegeven wordt geeft een foutmelding.	SoapUI	✓
Een read request van een windmolen die niet bestaat geeft een foutmelding terug.	SoapUI	✓

Na een update request wordt er een positieve bevestiging teruggestuurd	SoapUI	✓
Na een delete request wordt er een positieve bevestiging teruggestuurd	SoapUI	✓