Testplan REST

April 2, 2017

Contents

1 Versie **3**
1.1 Historie 3

2 Introductie **3**

3 Testomgeving **3**

4 Smoketest **3**

5 Unit tests **4**

1 Versie

Dit document is versie 0.2 van het testplan voor de REST applicatie.

1.1 Historie

Datum	Versie	Wijziging
22-03-2017	0.1	Eerste versie
02-04-2017	0.2	Integrationtests toegevoegd, smoketest uitgewerkt

2 Introductie

Dit project realiseerd een REST service geschreven in Java. Deze REST service verschaft toegang tot het aantal inwoners van een stad per jaar. Dit betreft de eerste versie van het project. Voor deze service zijn er unit tests gerealiseerd die de werking van de applicatie zullen testen. Ondanks dat het hier een zeer kleine applicatie betreft, zijn er toch integratietesten meegenomen.

3 Testomgeving

We gaan de applicatie testen door middel van unit tests en integration tests. Deze zijn op de commandline uit te voeren door naar de map van de applicatie te gaan en het volgende commando uit te voeren:

```
mvn clean install -B -U
```

De uitslag van de tests wordt na het uitvoeren van dit commando op het scherm getoond.

4 Smoketest

Deze tests zijn uitgevoerd door een derde partij. De uitkomst hiervan is hieronder beschreven:

Input (Naam, Jaar)	Output
a, 0	{"amount":0}
abc, 0	{"amount":1}
Hallo wereld, 0	{"amount":5}
Dit is een test, 5000	{"amount":67506}
Dit is een test, -100	{"amount":-1343}
Test, -1	{"amount":-1}

Het systeem heeft hierbij geen fouten gegeven en geeft de verwachte output.

5 Unit tests

De tests in dit project worden uitgevoerd door middel van unit tests. De code hiervan is hieronder beschreven. Zoals te zien is worden ook de itegratie tests uitgevoerd.

```java
public class DistrictServiceTest {
    private Map<String, Integer> input;
    private Map<String, Integer> output;
    private IntegrationTester t;

    @After
    public void tearDown() throws Exception {
        input = null;
        output = null;
        t = null;
    }

    @Before
    public void setUp() throws Exception {
        input = new HashMap<>();
        output = new HashMap<>();
        t = new IntegrationTester();


        input.put( k: "test",  v: 2017);
        output.put( k: "test",  v: 7263);

        input.put( k: "Test wijk",  v: 2017);
        output.put( k: "Test wijk",  v: 16341);

        input.put( k: "Echte wijk",  v: 2005);
        output.put( k: "Echte wijk",  v: 18049);
    }

    @org.junit.Test
    public void getInhabitants() throws Exception {
        this.input.keySet().forEach(k -> {
            int real = ServiceProvider.getParkingService().calcInhabitants(k, input.get(k)).getAmount();
            int expected = output.get(k);
            assertEquals(real, expected);
        });
    }

    @org.junit.Test
    public void integrationTest() throws Exception {
        t.run();
    }

}
```

Figure 1: Test sources

De output hiervan is te zien op het scherm nadat het commando in hoofdstuk 3 wordt uitgevoerd. De output hiervan is als volgt:

Figure 2: Test output