Test Rapport

Windmill webservice



Door: Jeroen de Kruijf

Klas: V2B

1. Versie

|  |  |
| --- | --- |
| **Versie** | **Bijgewerkt:** |
| 1.0 | 18-3-2017 |
| 1.0 | 25-3-2017 |

1. Introductie

Voor deze opdracht moest ik een webservice bedenken en realiseren in het kader van “Smart cities”. Ik heb een webservice gemaakt waarmee een gebruiker aan de hand van het vereiste kWh/jaar en een budget een lijst met mogelijke windturbines krijgt.

1. Testomgeving

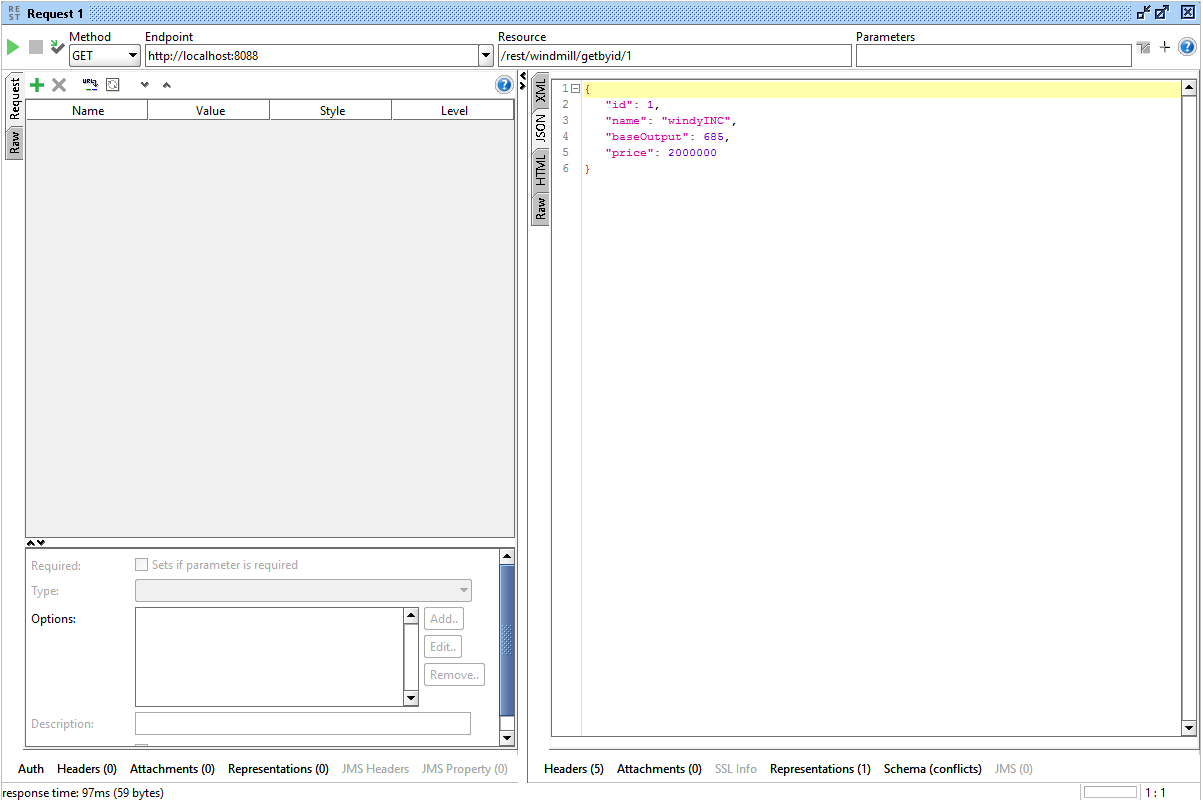
Voor het testen heb ik gebruik gemaakt van zowel Junit als SOAPUI. Junit heb ik gebruikt voor algemene tests en integratietests, en SOAPUI heb ik gebruikt voor de smoketest.

Ik heb de volgende services getest:

* getWindmill(id)
* getSuitableWindmills(requiredOutput, budget)
* getAllWindmills()
* addWindmill(windmill)

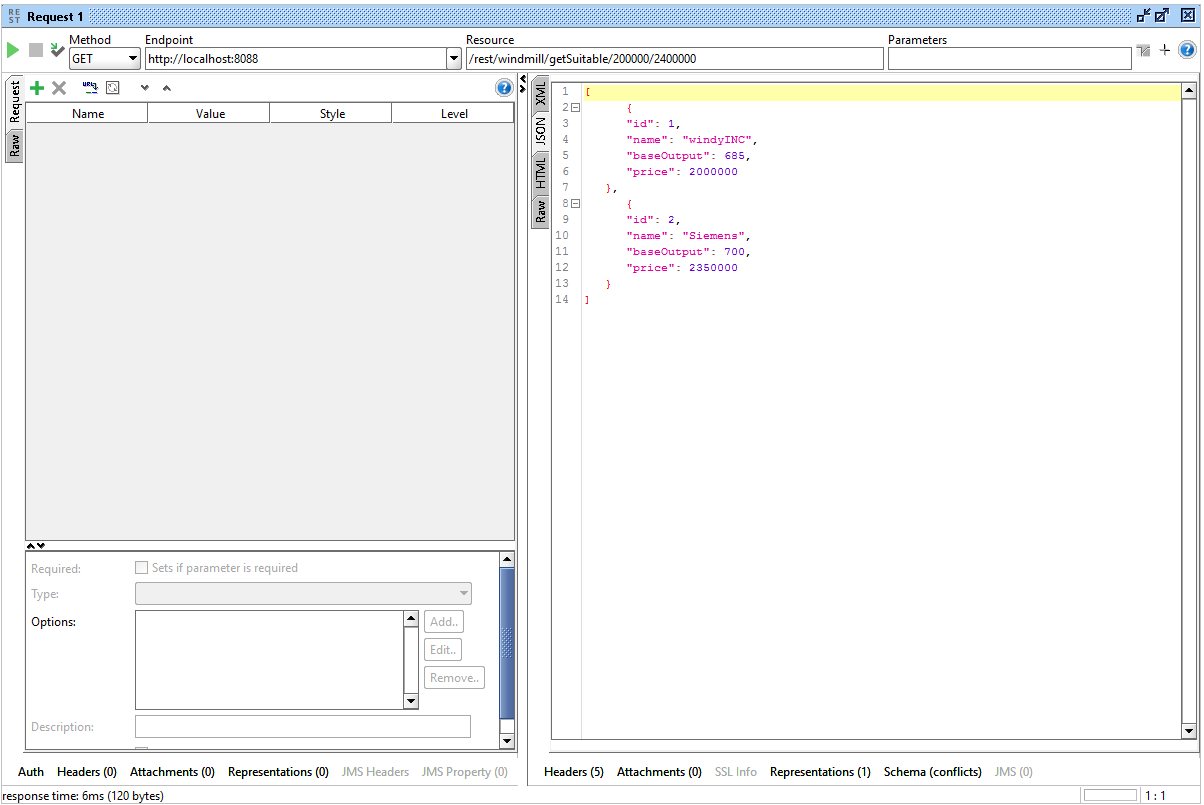
1. smoketest

# getWindmill(id)



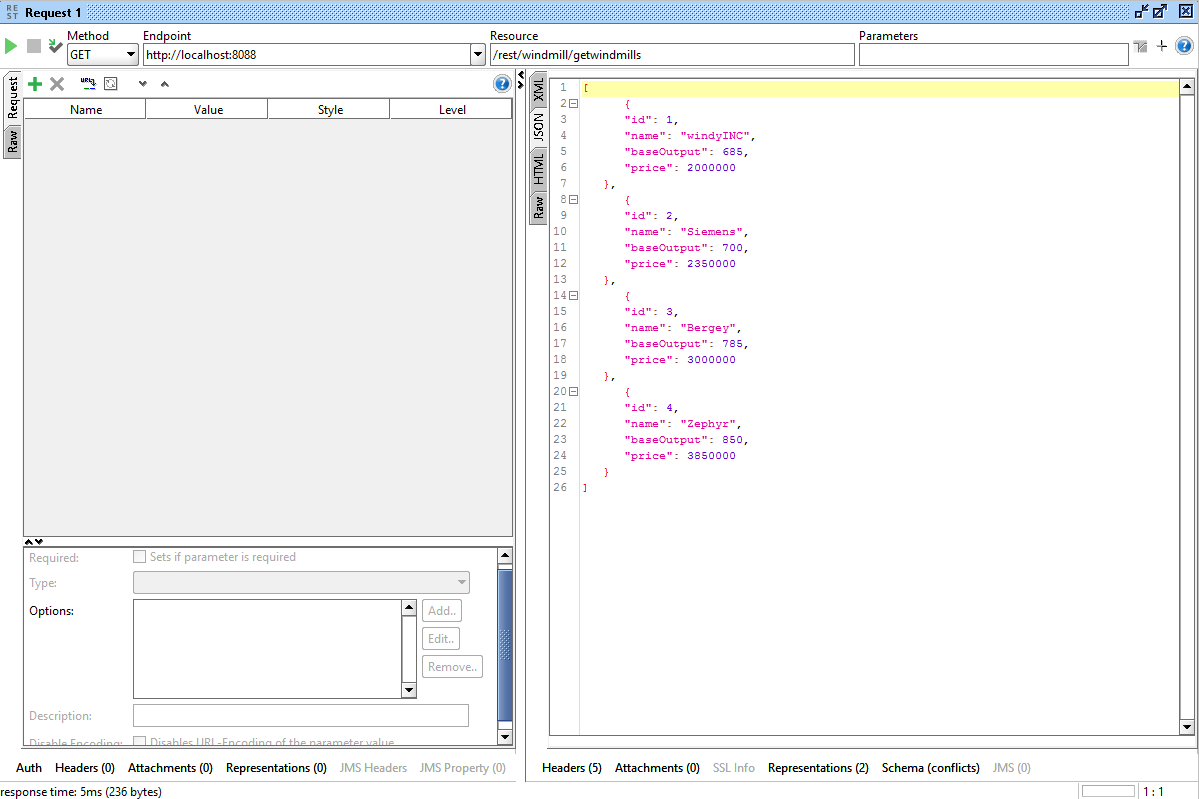
Zoals hierboven te zien is, returned de webservice de windturbine met de opgevraagde id.

# getSuitableWindmills(requiredOutput, budget)



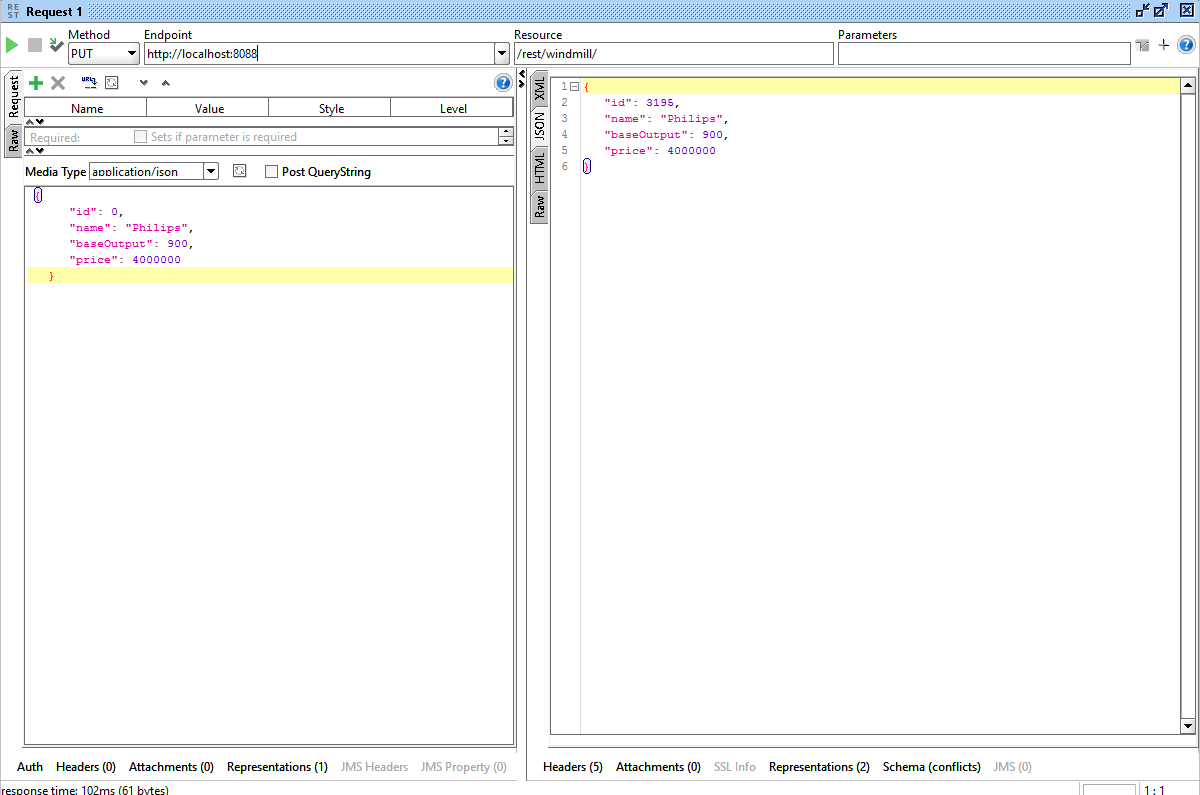
Deze functie functioneert naar toebehoren, als de waarden requiredOutput: 200000 en budget: 2400000 worden meegegeven worden de 2 mogelijke windturbines weergegeven.

# getAllWindmills()

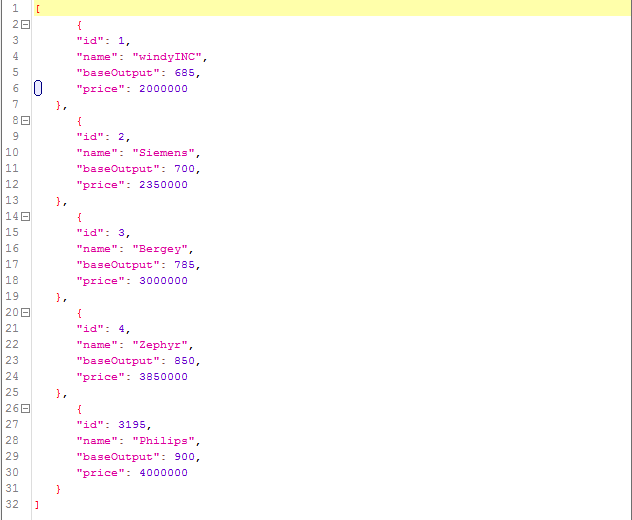


Functie returned alle windturbines die in het systeem staan.

# addWindmill(windmill)



Zoals hierboven te zien is word er een nieuwe windturbine toegevoegd aan de lijst met een gegenereerde id. Hieronder de nieuwe lijst:



1. black box tests

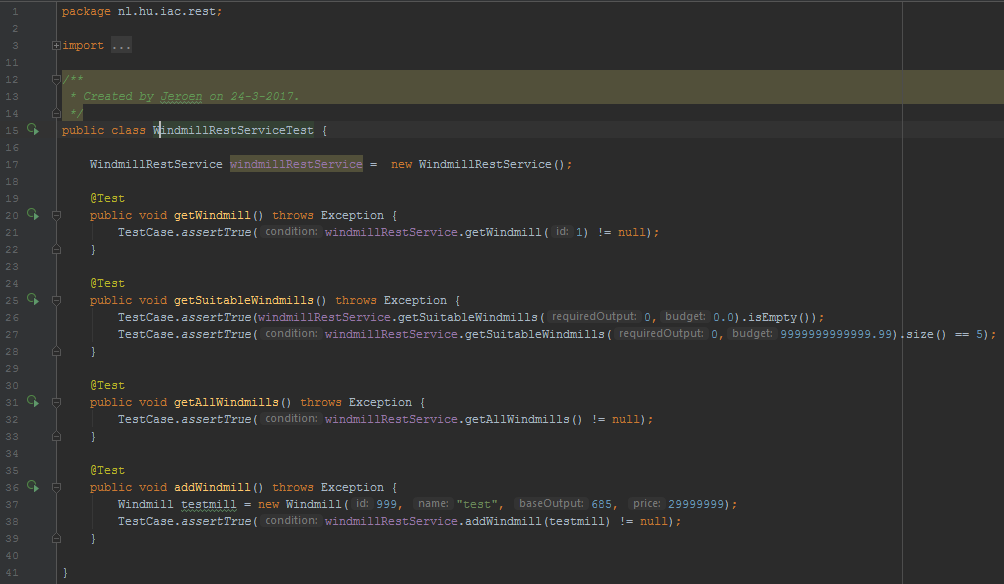
# getWindmill(id)

|  |  |  |  |
| --- | --- | --- | --- |
| **Input** | **Verwachte respone** | **Verkregen response** | **Geslaagd (Y/N)** |
| ID: 1 | {  "id": 1,  "name": "windyINC",  "baseOutput": 685,  "price": 2000000  } | {  "id": 1,  "name": "windyINC",  "baseOutput": 685,  "price": 2000000  } | Y |
| ID: 0 | Niks | Niks | Y |
| ID: Test | Niks | 404 not found | N |

# getSuitableWindmills(requiredOutput, budget)

|  |  |  |  |
| --- | --- | --- | --- |
| **Input** | **Verwachte respone** | **Verkregen response** | **Geslaagd (Y/N)** |
| requiredOutput:  200000  Budget:  2500000 | [  {  "id": 1,  "name": "windyINC",  "baseOutput": 685,  "price": 2000000  },  {  "id": 2,  "name": "Siemens",  "baseOutput": 700,  "price": 2350000  }  ] | [  {  "id": 1,  "name": "windyINC",  "baseOutput": 685,  "price": 2000000  },  {  "id": 2,  "name": "Siemens",  "baseOutput": 700,  "price": 2350000  }  ] | Y |
| requiredOutput:  0  Budget:  0 | Niks | Niks | Y |
| requiredOutput:  200000  Budget:  0 | Niks | Niks | Y |
| RequiredOutput:  0  Budget:  5000000 | [  {  "id": 1,  "name": "windyINC",  "baseOutput": 685,  "price": 2000000  },  {  "id": 2,  "name": "Siemens",  "baseOutput": 700,  "price": 2350000  },  {  "id": 3,  "name": "Bergey",  "baseOutput": 785,  "price": 3000000  },  {  "id": 4,  "name": "Zephyr",  "baseOutput": 850,  "price": 3850000  }  ] | [  {  "id": 1,  "name": "windyINC",  "baseOutput": 685,  "price": 2000000  },  {  "id": 2,  "name": "Siemens",  "baseOutput": 700,  "price": 2350000  },  {  "id": 3,  "name": "Bergey",  "baseOutput": 785,  "price": 3000000  },  {  "id": 4,  "name": "Zephyr",  "baseOutput": 850,  "price": 3850000  }  ] | Y |

1. Junit

**Junit test code:**

1. Integratietest

