# CADOCS
## Conversational Agent for the Detection of Community Smells

## Test Case Specification

https://github.com/gianwario/csDetector

### Team Members

Gianmario Voria
g.voria6@studenti.unisa.it

Antonio Della Porta
a.dellaporta26@studenti.unisa.it

### Reviewer

Stefano Lambiase
slambiase@unisa.it

18th July 2022

# Contents

# 1  Introduction

In this document, we will explicit the **Test Cases** for the test frames defined in the Test Plan.

# 2  Test Case Specification

## 2.1  CR_1-ATE: The system must be able to be integrated and executed through an Adapter

**Table 1** Test Case TC-ATE-1_1

| Test Case ID | Test Frame |
|---|---|
| **TC-ATE-1_1** | **GR1, GP1, SF1, OF1** |
| **Pre-conditions** | |
| <ul><li>the tool's environment has been installed and so the tool is executable;</li><li>a folder named **"senti"** has been created and contains SentiStrength data;</li><li>a folder named **"out"** has been created;</li><li>a **GitHub PAT** has been created and it correctly works for API calls.</li></ul> | |
| **Flow of events** | |
| The user writes the following parameters as input of the tool:<ul><li>*gitRepository*: https://github.com/tensorflow/serving</li><li>*gitPAT*: the GitHub PAT created</li><li>*sentiFolder*: senti</li><li>*outputFolder*: out</li></ul> | |
| **Oracle** | |
| The tool prints as output a list of the Community Smells detected. | |

**Table 2** Test Case TC-ATE-1_2

| Test Case ID | Test Frame |
|---|---|
| **TC-ATE-1_2** | **GR1, GP1, SF1, OF2** |

| Pre-conditions |
|---|
| <ul><li>the tool's environment has been installed and so the tool is executable;</li><li>a folder named **"senti"** has been created and contains SentiStrength data;</li><li>a folder named **"out"** has been created;</li><li>a **GitHub PAT** has been created and it correctly works for API calls.</li></ul> |

| Flow of events |
|---|
| The user writes the following parameters as input of the tool:<br><ul><li>*gitRepository*: https://github.com/tensorflow/serving</li><li>*gitPAT*: the GitHub PAT created</li><li>*sentiFolder*: senti</li><li>*outputFolder*: output</li></ul> |

| Oracle |
|---|
| The tool is not able to save output information and stops its execution without detecting smells. |

**Table 3** Test Case TC-ATE-1_3

| Test Case ID | Test Frame |
|---|---|
| **TC-ATE-1_3** | **GR1, GP1, SF1, OF3** |

| Pre-conditions |
|---|
| <ul><li>the tool's environment has been installed and so the tool is executable;</li><li>a folder named **"senti"** has been created and contains SentiStrength data;</li><li>a folder named **"out"** has been created;</li><li>a **GitHub PAT** has been created and it correctly works for API calls.</li></ul> |

| Flow of events |
|---|
| The user writes the following parameters as input of the tool:<br><ul><li>*gitRepository*: https://github.com/tensorflow/serving</li><li>*gitPAT*: the GitHub PAT created</li><li>*sentiFolder*: senti</li></ul> |

| Oracle |
|---|
| The tool warns the user that a needed parameter is missing and stops its execution without detecting smells. |

**Table 4** Test Case TC-ATE-1_4

| Test Case ID | Test Frame |
|---|---|
| **TC-ATE-1_4** | **GR1, GP1, SF2, OF1** |

| Pre-conditions |
|---|

- the tool's environment has been installed and so the tool is executable;
- a folder named **"senti"** has been created and contains SentiStrength data;
- a folder named **"out"** has been created;
- a **GitHub PAT** has been created and it correctly works for API calls.

| Flow of events |
|---|

The user writes the following parameters as input of the tool:

- *gitRepository*: https://github.com/tensorflow/serving
- *gitPAT*: the GitHub PAT created
- *sentiFolder*: sentifolder
- *sentiFolder*: out

| Oracle |
|---|

The tool does not find the required modules in the folder specified and stops its execution without detecting smells.

**Table 5** Test Case TC-ATE-1_5

| Test Case ID | Test Frame |
|---|---|
| **TC-ATE-1_5** | **GR1, GP1, SF3, OF1** |

| Pre-conditions |
|---|

- the tool's environment has been installed and so the tool is executable;
- a folder named **"senti"** has been created and contains SentiStrength data;
- a folder named **"out"** has been created;
- a **GitHub PAT** has been created and it correctly works for API calls.

| Flow of events |
|---|

The user writes the following parameters as input of the tool:

- *gitRepository*: https://github.com/tensorflow/serving
- *gitPAT*: the GitHub PAT created
- *sentiFolder*: out

| Oracle |
|---|

The tool warns the user that a needed parameter is missing and stops its execution without detecting smells.

**Table 6** Test Case TC-ATE-1_6

| Test Case ID | Test Frame |
|---|---|
| **TC-ATE-1_6** | **GR1, GP2, SF1, OF1** |

| Pre-conditions |
|---|
| <ul><li>the tool's environment has been installed and so the tool is executable;</li><li>a folder named **"senti"** has been created and contains SentiStrength data;</li><li>a folder named **"out"** has been created;</li><li>a **GitHub PAT** has been created and it correctly works for API calls.</li></ul> |

| Flow of events |
|---|
| The user writes the following parameters as input of the tool:<ul><li>*gitRepository*: https://github.com/tensorflow/serving</li><li>*sentiFolder*: senti</li><li>*sentiFolder*: out</li></ul> |

| Oracle |
|---|
| The tool warns the user that a needed parameter is missing and stops its execution without detecting smells. |

**Table 7** Test Case TC-ATE-1_7

| Test Case ID | Test Frame |
|---|---|
| **TC-ATE-1_7** | **GR2, GP1, SF1, OF1** |

| Pre-conditions |
|---|
| <ul><li>the tool's environment has been installed and so the tool is executable;</li><li>a folder named **"senti"** has been created and contains SentiStrength data;</li><li>a folder named **"out"** has been created;</li><li>a **GitHub PAT** has been created and it correctly works for API calls.</li></ul> |

| Flow of events |
|---|
| The user writes the following parameters as input of the tool:<ul><li>*gitRepository*: invalidurl</li><li>*gitPAT*: the GitHub PAT created</li><li>*sentiFolder*: senti</li><li>*sentiFolder*: out</li></ul> |

| Oracle |
|---|
| The tool can't find a repository at the specified URL and stops its execution without detecting smells. |

**Table 8** Test Case TC-ATE-1_8

| Test Case ID | Test Frame |
|---|---|
| **TC-ATE-1_8** | **GR3, GP1, SF1, OF1** |
| **Pre-conditions** | |

- the tool's environment has been installed and so the tool is executable;
- a folder named **"senti"** has been created and contains SentiStrength data;
- a folder named **"out"** has been created;
- a **GitHub PAT** has been created and it correctly works for API calls.

| **Flow of events** | |
|---|---|

The user writes the following parameters as input of the tool:

- *gitPAT*: the GitHub PAT created
- *sentiFolder*: senti
- *sentiFolder*: out

| **Oracle** | |
|---|---|

The tool warns the user that a needed parameter is missing and stops its execution without detecting smells.

## 2.2 CR_1-SSC: The system must be able to persist the results of the executions in a spreadsheet

**Table 9** Test Case TC-SSC-1_1

| Test Case ID | Test Frame |
|---|---|
| **TC-SSC-1_1** | **CF1, SD1, DS1** |
| **Pre-conditions** | |

- the tool is working and has been able to detect community smells;

| **Flow of events** | |
|---|---|

The tool has been able to collect the following pieces of information and uses them:

- *config*: [https://github.com/test/testRepository, testRepository, test]
- *starting_date*: 24/05/2022 00:26:00
- *detected_smells*: ["OSE", "SV", "UI"]

| **Oracle** | |
|---|---|

The system correctly saves the results in a csv file.

**Table 10** Test Case TC-SSC-1_2

| Test Case ID | Test Frame |
|---|---|
| **TC-SSC-1_2** | **CF1, SD1, DS2** |
| **Pre-conditions** | |

- the tool is working;

| **Flow of events** | |
|---|---|

The tool has been able to collect the following pieces of information and uses them:

- *config*: [https://github.com/test/testRepository, testRepository, test]
- *starting_date*: 24/05/2022 00:26:00

| **Oracle** | |
|---|---|

The system is not able to save results because they were not computed.


**Table 11** Test Case TC-SSC-1_3

| Test Case ID | Test Frame |
|---|---|
| **TC-SSC-1_3** | **CF1, SD2, DS1** |
| **Pre-conditions** | |

- the tool is working and has been able to detect community smells;

| **Flow of events** | |
|---|---|

The tool has been able to collect the following pieces of information and uses them:

- *config*: [https://github.com/test/testRepository, testRepository, test]
- *detected_smells*: ["OSE", "SV", "UI"]

| **Oracle** | |
|---|---|

The system correctly saves the results in a csv file but without a date.

**Table 12** Test Case TC-SSC-1_4

| Test Case ID | Test Frame |
|---|---|
| **TC-SSC-1_4** | **CF2, SD1, DS1** |
| **Pre-conditions** | |
| • the tool is working and has been able to detect community smells; | |
| **Flow of events** | |
| The tool has been able to collect the following pieces of information and uses them:<br><br>• *starting_date*: 24/05/2022 00:26:00<br>• *detected_smells*: ["OSE", "SV", "UI"] | |
| **Oracle** | |
| The system is not able to save the results of the execution. | |

## 2.3 CR_2-WSE: The system must be able to handle JSON input parameters and return a JSON formatted response

**Table 13** Test Case TC-WSE-1_1

| Test Case ID | Test Frame |
|---|---|
| **TC-WSE-1_1** | **GR1, GP1, US1** |
| **Pre-conditions** | |
| • the web service has been deployed;<br>• a valid GitPAT has been created; | |
| **Flow of events** | |
| The user makes an api call with the following parameters:<br><br>• *repo*: https://github.com/microsoft/QuantumKatas<br>• *pat*: the valid GitPAT<br>• *user*: UDRE54ED12 | |
| **Oracle** | |
| The system is able to detect community smells and print them. | |

**Table 14** Test Case TC-WSE-1_2

| Test Case ID | Test Frame |
|---|---|
| **TC-WSE-1_2** | **GR1, GP1, US2** |
| **Pre-conditions** | |
| <ul><li>the web service has been deployed;</li><li>a valid GitPAT has been created;</li></ul> | |
| **Flow of events** | |
| The user makes an api call with the following parameters:<ul><li>*repo*: https://github.com/microsoft/QuantumKatas</li><li>*pat*: the valid GitPAT</li></ul> | |
| **Oracle** | |
| The system is able to detect community smells and print them. | |

**Table 15** Test Case TC-WSE-1_3

| Test Case ID | Test Frame |
|---|---|
| **TC-WSE-1_3** | **GR1, GP2, US1** |
| **Pre-conditions** | |
| <ul><li>the web service has been deployed;</li></ul> | |
| **Flow of events** | |
| The user makes an api call with the following parameters:<ul><li>*repo*: https://github.com/microsoft/QuantumKatas</li><li>*user*: UDRE54ED12</li></ul> | |
| **Oracle** | |
| The system warns the user that a parameter is missing. | |

**Table 16** Test Case TC-WSE-1_4

| Test Case ID | Test Frame |
|---|---|
| **TC-WSE-1_4** | **GR2, GP1, US1** |
| **Pre-conditions** | |
| <ul><li>the web service has been deployed;</li><li>a valid GitPAT has been created;</li></ul> | |
| **Flow of events** | |
| The user makes an api call with the following parameters:<ul><li>*pat*: the valid GitPAT</li><li>*user*: UDRE54ED12</li></ul> | |
| **Oracle** | |
| The system warns the user that a parameter is missing. | |

## 2.4   CR_3-I1: The system must be able to execute CSDETECTOR on a given repository

**Table 17** Test Case TC-I1-1_1

| Test Case ID | Test Frame |
|---|---|
| **TC-I1-1_1** | **R1** |
| **Pre-conditions** | |
| <ul><li>the conversational agent is deployed on a messaging platform;</li></ul> | |
| **Flow of events** | |
| The user writes the following sentence in the channel:<ul><li>*repo*:     can     you     detect     community     smells     in     the     repository https://github.com/tensorflow/serving</li></ul> | |
| **Oracle** | |
| The tool is able to compute the sentence and may be able to detect community smells on the given repo. | |

**Table 18** Test Case TC-I1-1_2

| Test Case ID | Test Frame |
|---|---|
| **TC-I1-1_2** | **R2** |
| **Pre-conditions** | |

- the conversational agent is deployed on a messaging platform;

| **Flow of events** | |
|---|---|

The user writes the following sentence in the channel:

- *repo*: can you detect community smells

| **Oracle** | |
|---|---|

The tool is able to compute the sentence but can't detect smells because a parameter is missing.

## 2.5   CR_3-I2: The system must be able to execute CSDETECTOR on a given repository starting from a specified date

**Table 19** Test Case TC-I2-1_1

| Test Case ID | Test Frame |
|---|---|
| **TC-I2-1_1** | **R1, D1** |
| **Pre-conditions** | |

- the conversational agent is deployed on a messaging platform;

| **Flow of events** | |
|---|---|

The user writes the following sentence in the channel:

- *repo & date*:   can   you   detect   community   smells   in   the   repository https://github.com/tensorflow/serving starting from 01/01/2021

| **Oracle** | |
|---|---|

The tool is able to compute the sentence and may be able to detect community smells on the given repo starting from the specified date.

**Table 20** Test Case TC-I2-1_2

| Test Case ID | Test Frame |
|---|---|
| **TC-I2-1_2** | **R1, D2** |
| **Pre-conditions** | |
| • the conversational agent is deployed on a messaging platform; | |
| **Flow of events** | |
| The user writes the following sentence in the channel: <br><br> • *repo & date*: can you detect community smells in the repository https://github.com/tensorflow/serving | |
| **Oracle** | |
| The tool is able to compute the sentence and may be able to detect community smells on the given repo but not starting from a specified date. | |

**Table 21** Test Case TC-I2-1_3

| Test Case ID | Test Frame |
|---|---|
| **TC-I2-1_3** | **R2** |
| **Pre-conditions** | |
| • the conversational agent is deployed on a messaging platform; | |
| **Flow of events** | |
| The user writes the following sentence in the channel: <br><br> • *repo*: can you detect community smells | |
| **Oracle** | |
| The tool is able to compute the sentence but can't detect smells because a parameter is missing. | |