Replication Utilities

This repository includes the steps and information needed to replicate our study.
1- Detection of multi-language code smell occurrences.
2- Data Collection (Snapshots, logs, and developers' information).
3- Data Analysis.
4- Statistical Analysis performed.

This project aims to assess the perception of the defined design smells from the developers' perspective. We also aim to investigate developers' perception of the severity of the multi-language design smells and their impacts on software quality attributes.

1. **Detection of multi-language design smells occurrences**
   Location: Folder Detection Approach

   - **Approach:** Detection Approach
   - **Results**: Results of the detection
   - **Evaluation**: Evaluation of the Approach

Getting Started

Running

Run /MLS SAD/src/mlssad/DetectCodeSmellsAndAntiPatterns.java with the path to a directory or file as the only argument. The program will output a CSV file with the code smells and anti-patterns detected in the input source.

Customizing

Change the parameters in /MLS SAD/rsc/config.properties to adapt to your needs. It is currently configured following the default values as thresholds.

Running the tests

The directory /MLS SAD Tests contains tests for each code smell and anti-pattern individually, and two test suites (Applied with the PilotProject).

The tests require the pilot project for detection of anti-patterns and code smells in multi-language systems:

1- Clone the pilot project (PilotProjectAPCSMLS).
2- Create a junction between the folder MLS SAD Tests/rsc and the pilot project folder. On Windows, assuming that the two projects are in the same folder (otherwise, include their paths):
MKLINK /D /J "MLS-SAD\MLS SAD Tests\rsc" "PilotProjectAPCSMLS"

Dependencies

srcML - A parsing tool to convert code to srcML, a subset of XML

Apache Commons Compress - A library for working with archives

Acknowledgments
Loosely inspired by the SAD tool in Ptidej

2. **Data Collection (Snapshots, logs, and developers' information).**
   Location: Folder Data Collection
   Scripts: Contains the script used to clone the projects, collect the logs and developers information, and map the results with the smell occurrences
   Results: Contains the results of the logs and developers information, and mapping of the results with the smell occurrences.

Getting Started

Folder Scripts:

- Run the script gitclone.py to extract and the clone the projects from GitHub
- Run the script extract_snapshots.py to extract snapshots each 90 days interval
- Run the detection approach described in the previous section on these snapshots.
- Run the script extract_developers_modified_files.py to extract the information about the developers that contributed to the studied systems.
- Run the script merge_files.py to perform a mapping between information of smells and the developer's information.

Results in Folders:
"Developers-info" contains the script and data related to the developers who contributed to the analyzed systems.
"File-all-merged" contains the mapping between the smells and the developers who contributed to the smelly files.
"Snapshots_information" contains the logs related to the subject systems.
"Surveys": Contains the surveys used.

3. **Data Analysis**
   Approach: Combination of python script and manual analysis for the coding process
   src: Contains the scripts used for the data analysis (Folder MLS-survey)
   Results: Contains the results of the quantitative results, figures, and the statistical tests applied to answer our research questions

Getting Started

For RQ1, RQ3, and RQ5

- Input file all-cleaned-data.csv
- Run the scripts data_processing.py then suvey_data-analysis.py

- Input file and scripts for all the quantitative results, figures, and the statistical tests applied for RQ1 and RQ5. (check Notebooks: “MLS-SurveyDataAnalysis.ipynb”, “Survey-Data-Charts.ipynb”)

For RQ2, Manual Validation and attribution of tags and categories for the sheet “IntroductionSmells” in “EMSEDataSurvey.xlsx”.

For RQ4, Borda count:

- Input file “Smells-ranking-BordaDecCroissant.csv”
- Run the script borda_ranking.py to apply the algo of borda count
- Output file: Smells-ranking-BordaDecCroissant.txt

4. **Statistical tests (Folder MLS-survey)**
   src: Contains the scripts used for the statistical tests
   Results: Contains the statistical tests applied to answer our research questions

Getting Started

- Fisher exact tests (Fisher_test_reports)
  - For RQ1: “Fishers-test-validation.txt”
  - For RQ5: “Fishers-test-refactoring-new.txt”