

# Automated Design Smell Detection and Refactoring Pipeline

## Overview

These two Python scripts(`refactoring_pipeline.py`, `refactoring_pipe_groq.py`) automate the detection and refactoring of design smells in Java code hosted on GitHub. One script leverages Google's Gemini AI, while the other uses Groq AI. Both follow a similar workflow but differ in how they analyze and refactor code.

## Features

- **Scans Java files** in a target directory within a GitHub repository.
- **Detects design smells** such as God Class, Feature Envy, Long Method, etc.
- **Uses Gemini AI or Groq AI** for automated analysis and refactoring.
- **Creates a new branch** and updates files with improved code.
- **Generates a pull request (PR)** for repository maintainers to review.
- **Runs periodically** as per cron job settings.

## Setup & Requirements

```
# Configuration from environment variables
GEMINI_API_KEY = os.environ["GEMINI_API_KEY"] # Your Gemini API key
GITHUB_TOKEN = os.environ["GITHUB_TOKEN"]
REPO_NAME = os.environ["GITHUB_REPOSITORY"] # e.g.,
"SE-course-serc/project-1-team-9"
TARGET_FOLDER = os.environ.get("TARGET_FOLDER",
"reader-core/src/main/java/com/sismics/reader/core/dao/lucene")
BASE_BRANCH = os.environ.get("BASE_BRANCH", "master")
```

## Environment Variables

Ensure the following environment variables are set:

```
export AI_API_KEY="your_gemini_or_groq_api_key"
export GITHUB_TOKEN="your_github_token"
export GITHUB_REPOSITORY="user/repo"
export TARGET_FOLDER="path/to/java/files"
export BASE_BRANCH="master"
```

## Install Dependencies

```
pip install google-generativeai groq PyGithub lizard
```

## Execution

Run the script manually:

```
python script.py
```

Or schedule it with a cron job for periodic execution:

```
crontab -e
```

Add the following line to run every day at midnight:

```
0 0 * * * /usr/bin/python3 /path/to/script.py
```

# Workflow

## 1. Retrieve Java Files

- Connects to the specified GitHub repository.
- Fetches Java files from the target directory.
- Uses the GitHub API to read file contents and track changes.

## 2. Detect Design Smells

- Uses Gemini or Groq AI to analyze code and detect design issues.
- The Gemini script processes two files at a time, analyzing for predefined smells.
- The Groq script includes additional code metrics (lines of code, function count, cyclomatic complexity) for better context.
- If no issues are found, the process stops.

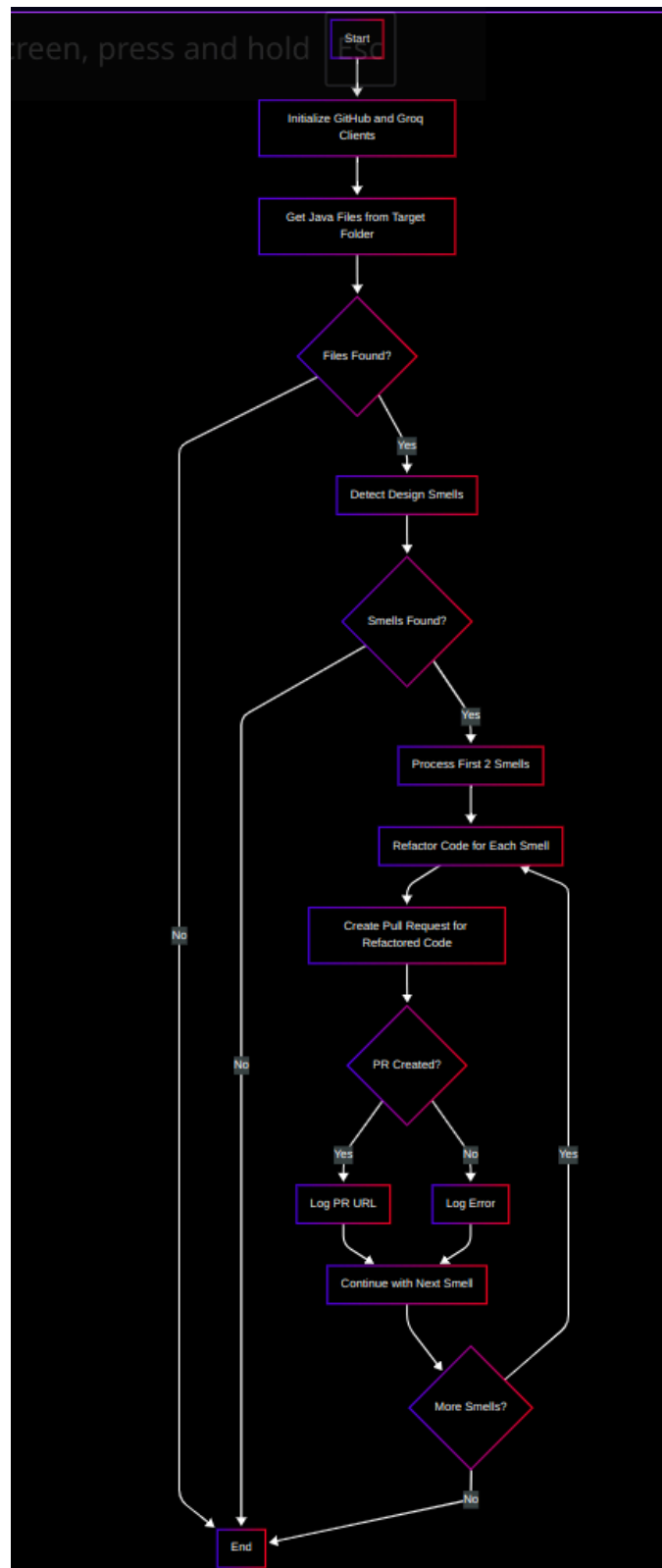
## 3. Refactor Code

- AI generates refactored code for each detected issue.
- Ensures improved structure while maintaining functionality.

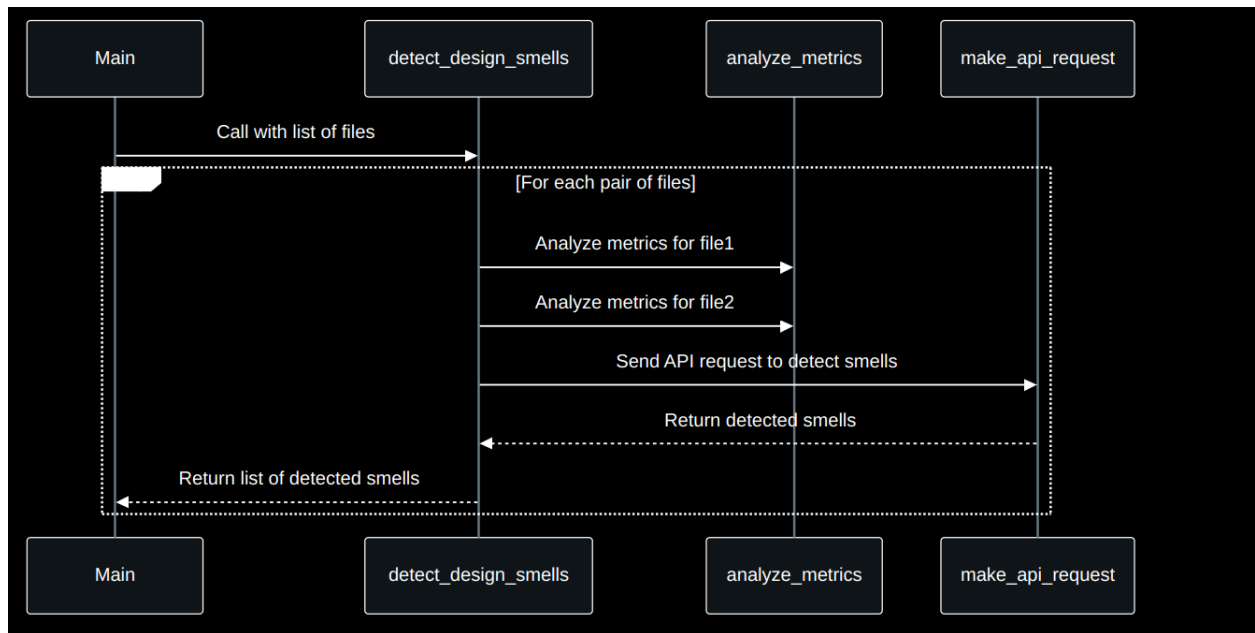
## 4. Create Pull Request

- Generates a new branch for refactored changes.
- Updates affected files in the repository.
- Submits a pull request with a summary of the improvements.

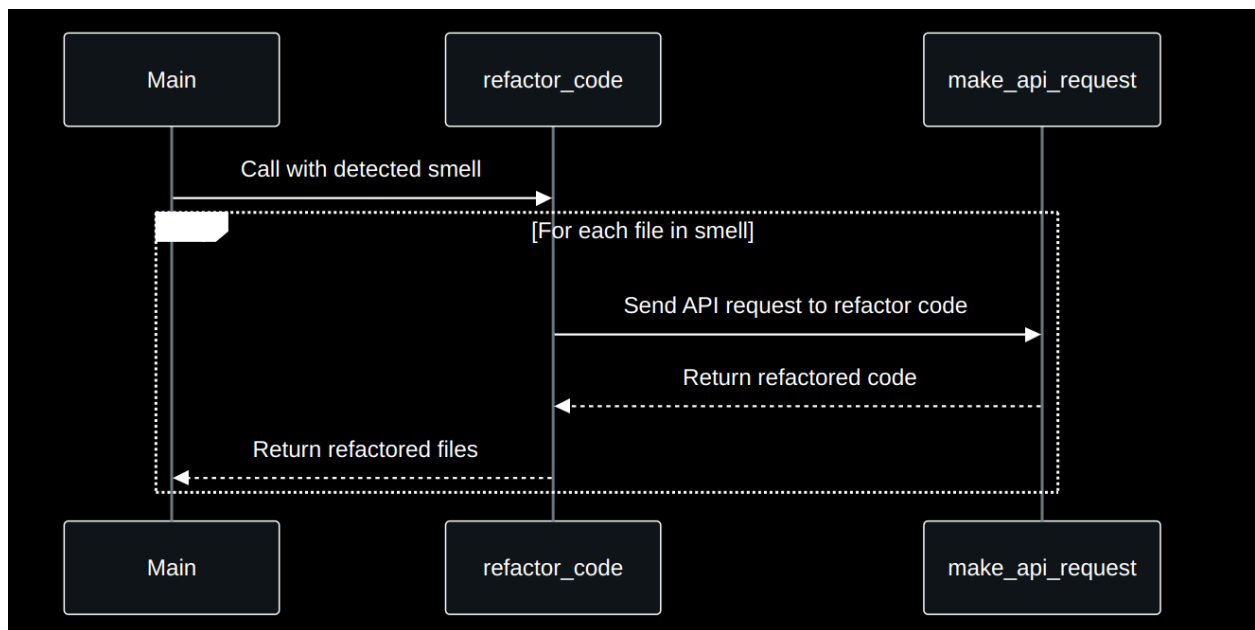
# Flowchart of Main Execution Process



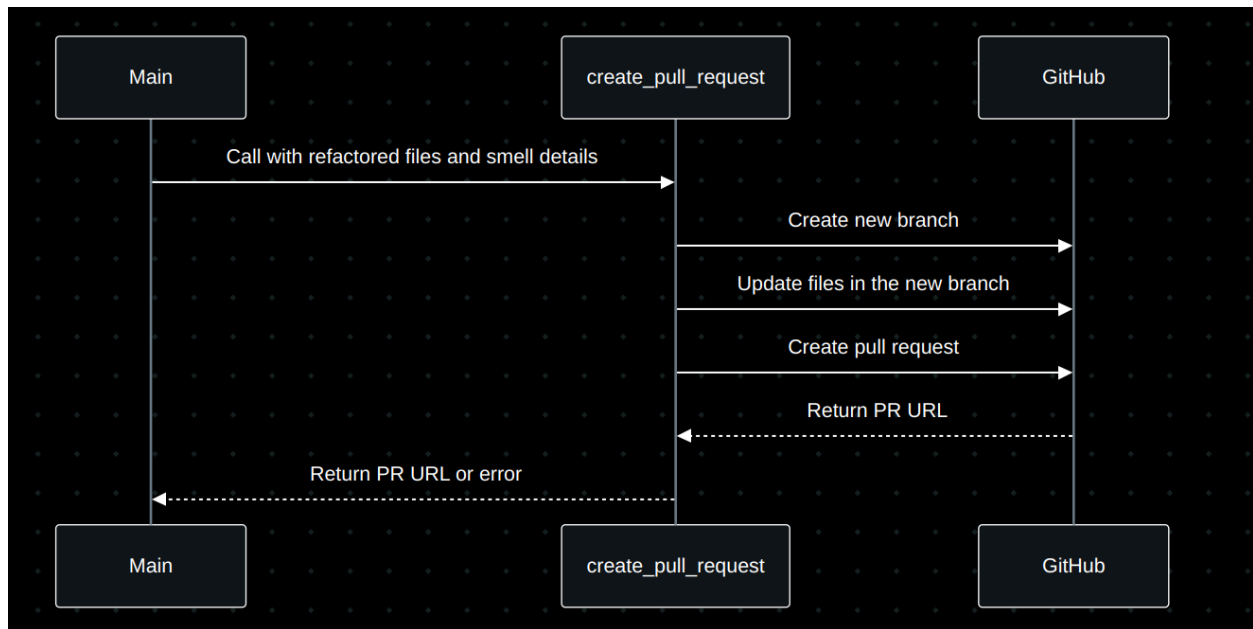
## Sequence Diagram for `detect_design_smells`



## Sequence Diagram for `refactor_code`



## Sequence Diagram for `create_pull_request`



## GitHub Actions Workflow

The automation is integrated with GitHub Actions to run on various triggers:

name: Automated Design Smell Detection using Gemini & Refactoring with Metrics

```
on:
  push:
    branches:
      - '**' # Matches any branch
  schedule:
    # Runs every 3 hours
    - cron: '0 */3 * * *'
  workflow_dispatch: # Allows manual trigger

jobs:
  refactor:
    runs-on: ubuntu-latest
```

steps:

- name: Checkout Repository  
uses: actions/checkout@v3
- name: Set up Python  
uses: actions/setup-python@v4  
with:  
python-version: '3.x'
- name: Install Dependencies  
run: |  
pip install PyGithub google-generativeai lizard
- name: Run Refactoring Pipeline  
env:  
GEMINI\_API\_KEY: \${ secrets.GEMINI\_API\_KEY }  
GITHUB\_TOKEN: \${ secrets.GITHUB\_TOKEN }  
GITHUB\_REPOSITORY: "SE-course-serc/project-1-team-9"  
TARGET\_FOLDER:  
"reader-core/src/main/java/com/sismics/reader/core/dao/lucene"  
BASE\_BRANCH: "master"  
run: |  
python refactoring\_pipeline.py

← Automated Design Smell Detection & Refactoring with Metrics

🟢 Consolidated all the intermediate docs #5

Re-run all jobs ⋮

Summary

Jobs

Run details

Usage

Workflow file

refactor

succeeded 4 hours ago in 52s

Search logs

🔄 ⚙️

> Set up job 0s

> Checkout Repository 1s

> Set up Python 0s

> Install Dependencies 9s

> Run Refactoring Pipeline 39s

> Post Set up Python 0s

> Post Checkout Repository 0s

> Complete job 0s

1 Run python refactoring\_pipeline.py

16 2025-02-13 13:21:42,277 - INFO - Scanning folder: reader-core/src/main/java/com/sismics/reader/core/dao/lucene

17 2025-02-13 13:21:42,277 - INFO - Found Java file: reader-core/src/main/java/com/sismics/reader/core/dao/lucene/ArticleDao.java

18 2025-02-13 13:21:42,650 - INFO - Found Java file: reader-core/src/main/java/com/sismics/reader/core/dao/lucene/ReaderStandardAnalyzer.java

19 2025-02-13 13:21:43,860 - INFO - Analyzing pair:

20 - File 1: reader-core/src/main/java/com/sismics/reader/core/dao/lucene/ArticleDao.java

21 - File 2: reader-core/src/main/java/com/sismics/reader/core/dao/lucene/ReaderStandardAnalyzer.java

22 2025-02-13 13:21:45,166 - INFO - Detected smell: Feature envy

23 2025-02-13 13:21:45,166 - INFO - Found 1 design smells

24 2025-02-13 13:21:45,166 - INFO - Refactoring file: reader-core/src/main/java/com/sismics/reader/core/dao/lucene/ArticleDao.java

25 2025-02-13 13:22:00,326 - INFO - Successfully refactored reader-core/src/main/java/com/sismics/reader/core/dao/lucene/ArticleDao.java

26 2025-02-13 13:22:06,180 - INFO - Refactoring file: reader-core/src/main/java/com/sismics/reader/core/dao/lucene/ReaderStandardAnalyzer.java

27 2025-02-13 13:22:12,696 - INFO - Successfully refactored reader-core/src/main/java/com/sismics/reader/core/dao/lucene/ReaderStandardAnalyzer.java

28 2025-02-13 13:22:17,946 - INFO - Pull Request created: <https://github.com/SE-course-serc/project-1-team-9/pull/14>

29 2025-02-13 13:22:17,946 - INFO - Successfully created pull request: <https://github.com/SE-course-serc/project-1-team-9/pull/14>

30 WARNING: All log messages before shh::InitialLogup() is called are written to STDERR

31 E0000 00-00:1730452940.022628 1750 init.cc:232] grpc\_wait\_for\_shutdown\_with\_timeout() timed out.

ActionsProjectsSecurityInsightsSettings

All workflows

Filter workflow runs

Showing runs from all workflows

17 workflow runs		Event	Status	Branch	Actor
Consolidated all the intermediate docs	Automated Design Smell Detection & Refactoring with Metrics #5: Commit a50ae33 pushed by sriyasahu	test	4 hours ago	1m 2s	...
3B CodeMr Fix document and 3D(pipelineRefactoring) documents	Automated Design Smell Detection & Refactoring with Metrics #4: Commit 19c6fd1 pushed by bdsh18	test	4 hours ago	32s	...
test	Automated Design Smell Detection & Refactoring with Metrics #3: Commit 42d190e pushed by ss-369	test	5 hours ago	26s	...
added refactoring_pipeline for gemini	Automated Design Smell Detection & Refactoring with Metrics #2: Commit f0e6604 pushed by ss-369	test	5 hours ago	30s	...
added refactoring_pipeline for gemini	Automated Design Smell Detection & Refactoring with Metrics #1: Commit 921326e pushed by ss-369	test	5 hours ago	27s	...
Refactor: Fix feature_envy in reader-core/src/main/java/com/sismics/r...	Refactoring Pipeline #12: Commit c87ebea pushed by ss-369	refactor/feature-envy/202...	7 hours ago	24s	...
Refactor: Fix feature_envy in reader-core/src/main/java/com/sismics/r...	Refactoring Pipeline #11: Commit b2f521d pushed by ss-369	refactor/feature-envy/202...	7 hours ago	28s	...
test	Refactoring Pipeline #10: Commit 82b3991 pushed by ss-369	refactor/feature-envy/202...	7 hours ago	31s	...
test	Refactoring Pipeline #9: Commit 82b3991 pushed by ss-369	test	yesterday	23s	...

ActionsProjectsSecurityInsightsSettings

Filters

is:pr is:open

Labels10

Milestones0

New pull request

23 Open2 Closed		Author	Label	Projects	Milestones	Reviews	Assignee	Sort
Refactor: Fix long_method	#31 opened 3 hours ago by ss-369							
Refactor: Fix god_class	#30 opened 3 hours ago by ss-369							
Refactor: Fix data_class	#29 opened 3 hours ago by ss-369							
Refactor: Fix god_class	#28 opened 3 hours ago by ss-369							
Refactor: Fix feature_envy	#27 opened 3 hours ago by ss-369							
Refactor: Fix inappropriate_intimacy	#26 opened 3 hours ago by ss-369							
Refactor: Fix feature_envy	#25 opened 3 hours ago by ss-369							
Refactor: Fix large_class	#24 opened 3 hours ago by ss-369							
Refactor: Fix long_method	#23 opened 3 hours ago by ss-369							
Refactor: Fix feature_envy	#22 opened 3 hours ago by ss-369							
Refactor: Fix for file 1, the search_method_is_doing_a_lot	#21 opened 3 hours ago by ss-369							
Refactor: Fix magic_numbers								



## Key Differences Between Scripts

- **AI Service:** One script uses Gemini, the other uses Groq.
- **Code Metrics:** The Groq script incorporates additional code metrics for better AI analysis.
- **Retry Logic:** The Groq script includes robust retry logic to handle API rate limits.
- **Smell Lists:** Though similar, the lists of design smells each script detects differ slightly.

## Logging & Error Handling

- Logs all key actions and errors.
- Handles API failures and GitHub access issues.
- Implements retry mechanisms for robustness (Groq script only).
- Skips problematic files while continuing with the process.

### Sample outputs:

```
2025-02-13 19:45:40,165 - INFO - Scanning folder:
reader-core/src/main/java/com/sismics/reader/core/dao/lucene
2025-02-13 19:45:40,165 - INFO - Found Java file:
reader-core/src/main/java/com/sismics/reader/core/dao/lucene/ArticleDao.java
2025-02-13 19:45:40,762 - INFO - Found Java file:
reader-core/src/main/java/com/sismics/reader/core/dao/lucene/ReaderStandardA
nalyzer.java
2025-02-13 19:45:41,354 - INFO - Analyzing pair:
- File 1:
reader-core/src/main/java/com/sismics/reader/core/dao/lucene/ArticleDao.java
- File 2:
reader-core/src/main/java/com/sismics/reader/core/dao/lucene/ReaderStandardA
nalyzer.java

2025-02-13 19:45:42,061 - INFO - HTTP Request: POST
https://api.groq.com/openai/v1/chat/completions "HTTP/1.1 200 OK"
2025-02-13 19:45:42,074 - INFO - Raw API response: safe
2025-02-13 19:45:42,074 - INFO - Found 0 design smells
(refac) → project-1-team-9 git:(test) X python3 oen.py
2025-02-13 19:49:57,337 - INFO - Scanning folder:
```

```
reader-core/src/main/java/com/sismics/reader/core/dao/lucene
2025-02-13 19:49:57,338 - INFO - Found Java file:
reader-core/src/main/java/com/sismics/reader/core/dao/lucene/ArticleDao.java
2025-02-13 19:49:57,894 - INFO - Found Java file:
reader-core/src/main/java/com/sismics/reader/core/dao/lucene/ReaderStandardA
nalyzer.java
2025-02-13 19:49:58,461 - INFO - Analyzing pair:
  - File 1:
reader-core/src/main/java/com/sismics/reader/core/dao/lucene/ArticleDao.java
  - File 2:
reader-core/src/main/java/com/sismics/reader/core/dao/lucene/ReaderStandardA
nalyzer.java
2025-02-13 19:50:01,371 - INFO - HTTP Request: POST
https://api.groq.com/openai/v1/chat/completions "HTTP/1.1 200 OK"
2025-02-13 19:50:01,381 - INFO - Raw API response: god_class: ArticleDao
class has multiple responsibilities such as rebuilding index, creating,
updating, and deleting articles, as well as searching for articles, which
makes it a god class.
long_method: The search method in ArticleDao is quite long and complex,
making it hard to understand and maintain.
primitive_obsession: The ArticleDao class uses many primitive types such as
strings and integers, which can make the code harder to understand and
maintain.
magic_numbers: The ArticleDao class uses magic numbers such as 100000 and
20, which can make the code harder to understand and maintain.
long_parameter_list: The search method in ArticleDao has a long parameter
list, which can make the code harder to understand and maintain.
tight_coupling: The ArticleDao class is tightly coupled with the Lucene
library, which can make it harder to change or replace the library.
data_class: The Article class seems to be a data class, as it only has
getter and setter methods, which can make the code less object-oriented.
large_class: The ArticleDao class has many methods and fields, which can
make it harder to understand and maintain.
deep_nesting: The search method in ArticleDao has deep nesting, which can
make the code harder to understand and maintain.
complex_conditional: The search method in ArticleDao has complex
conditionals, which can make the code harder to understand and maintain.
global_data: The ArticleDao class uses global data such as the ApplicationContext,
which can make the code harder to understand and maintain.
feature_envy: The ArticleDao class seems to envy the features of the Lucene
library, as it uses many of its classes and methods directly.
2025-02-13 19:50:01,381 - INFO - Detected smell: god_class
2025-02-13 19:50:01,382 - INFO - Detected smell: long_method
```

```
2025-02-13 19:50:01,382 - INFO - Detected smell: primitive_obsession
2025-02-13 19:50:01,382 - INFO - Detected smell: magic_numbers
2025-02-13 19:50:01,382 - INFO - Detected smell: long_parameter_list
2025-02-13 19:50:01,382 - INFO - Detected smell: tight_coupling
2025-02-13 19:50:01,382 - INFO - Detected smell: data_class
2025-02-13 19:50:01,383 - INFO - Detected smell: large_class
2025-02-13 19:50:01,383 - INFO - Detected smell: deep_nesting
2025-02-13 19:50:01,383 - INFO - Detected smell: complex_conditional
2025-02-13 19:50:01,383 - INFO - Detected smell: global_data
2025-02-13 19:50:01,384 - INFO - Detected smell: feature_envy
2025-02-13 19:50:01,384 - INFO - Found 12 design smells
2025-02-13 19:50:01,384 - INFO - More than 2 design smells detected.
Processing only the first 2.
2025-02-13 19:50:01,384 - INFO - Refactoring file:
reader-core/src/main/java/com/sismics/reader/core/dao/lucene/ArticleDao.java
2025-02-13 19:50:08,019 - INFO - HTTP Request: POST
https://api.groq.com/openai/v1/chat/completions "HTTP/1.1 200 OK"
2025-02-13 19:50:08,024 - INFO - Successfully refactored
reader-core/src/main/java/com/sismics/reader/core/dao/lucene/ArticleDao.java
2025-02-13 19:50:08,025 - INFO - Refactoring file:
reader-core/src/main/java/com/sismics/reader/core/dao/lucene/ReaderStandardA
nalyzer.java
2025-02-13 19:50:08,306 - INFO - HTTP Request: POST
https://api.groq.com/openai/v1/chat/completions "HTTP/1.1 429 Too Many
Requests"
2025-02-13 19:50:08,308 - INFO - Retrying request to
/openai/v1/chat/completions in 16.000000 seconds
2025-02-13 19:50:29,591 - INFO - HTTP Request: POST
https://api.groq.com/openai/v1/chat/completions "HTTP/1.1 200 OK"
2025-02-13 19:50:29,592 - INFO - Successfully refactored
reader-core/src/main/java/com/sismics/reader/core/dao/lucene/ReaderStandardA
nalyzer.java
2025-02-13 19:50:36,082 - INFO - Pull Request created:
https://github.com/SE-course-serc/project-1-team-9/pull/30
2025-02-13 19:50:36,082 - INFO - Successfully created pull request:
https://github.com/SE-course-serc/project-1-team-9/pull/30
2025-02-13 19:50:36,082 - INFO - Refactoring file:
reader-core/src/main/java/com/sismics/reader/core/dao/lucene/ArticleDao.java
2025-02-13 19:50:36,463 - INFO - HTTP Request: POST
https://api.groq.com/openai/v1/chat/completions "HTTP/1.1 429 Too Many
Requests"
2025-02-13 19:50:36,464 - INFO - Retrying request to
/openai/v1/chat/completions in 20.000000 seconds
```

```
2025-02-13 19:51:03,907 - INFO - HTTP Request: POST
https://api.groq.com/openai/v1/chat/completions "HTTP/1.1 200 OK"
2025-02-13 19:51:03,914 - INFO - Successfully refactored
reader-core/src/main/java/com/sismics/reader/core/dao/lucene/ArticleDao.java
2025-02-13 19:51:03,914 - INFO - Refactoring file:
reader-core/src/main/java/com/sismics/reader/core/dao/lucene/ReaderStandardA
nalyzer.java
2025-02-13 19:51:04,198 - INFO - HTTP Request: POST
https://api.groq.com/openai/v1/chat/completions "HTTP/1.1 429 Too Many
Requests"
2025-02-13 19:51:04,198 - INFO - Retrying request to
/openai/v1/chat/completions in 21.000000 seconds
2025-02-13 19:51:28,453 - INFO - HTTP Request: POST
https://api.groq.com/openai/v1/chat/completions "HTTP/1.1 200 OK"
2025-02-13 19:51:28,456 - INFO - Successfully refactored
reader-core/src/main/java/com/sismics/reader/core/dao/lucene/ReaderStandardA
nalyzer.java
2025-02-13 19:51:35,085 - INFO - Pull Request created:
https://github.com/SE-course-serc/project-1-team-9/pull/31
2025-02-13 19:51:35,085 - INFO - Successfully created pull request:
https://github.com/SE-course-serc/project-1-team-9/pull/31
```

```
(refac) → project-1-team-9 git:(test) X python3 refactoring_pipeline.py
2025-02-13 19:37:12,661 - INFO - Scanning folder:
reader-core/src/main/java/com/sismics/reader/core/dao/lucene
2025-02-13 19:37:12,661 - INFO - Found Java file:
reader-core/src/main/java/com/sismics/reader/core/dao/lucene/ArticleDao.java
2025-02-13 19:37:13,243 - INFO - Found Java file:
reader-core/src/main/java/com/sismics/reader/core/dao/lucene/ReaderStandardAn
alyzer.java
2025-02-13 19:37:13,821 - INFO - Analyzing pair:
- File 1:
reader-core/src/main/java/com/sismics/reader/core/dao/lucene/ArticleDao.java
- File 2:
reader-core/src/main/java/com/sismics/reader/core/dao/lucene/ReaderStandardAn
alyzer.java
2025-02-13 19:37:21,775 - INFO - Detected smell: data_class
2025-02-13 19:37:21,776 - INFO - Found 1 design smells
2025-02-13 19:37:21,776 - INFO - Refactoring file:
reader-core/src/main/java/com/sismics/reader/core/dao/lucene/ArticleDao.java
2025-02-13 19:38:13,284 - INFO - Successfully refactored
reader-core/src/main/java/com/sismics/reader/core/dao/lucene/ArticleDao.java
```

```
2025-02-13 19:38:13,285 - INFO - Refactoring file:
reader-core/src/main/java/com/sismics/reader/core/dao/lucene/ReaderStandardAn
alyzer.java
2025-02-13 19:38:41,747 - INFO - Successfully refactored
reader-core/src/main/java/com/sismics/reader/core/dao/lucene/ReaderStandardAn
alyzer.java
2025-02-13 19:38:48,258 - INFO - Pull Request created:
https://github.com/SE-course-serc/project-1-team-9/pull/29
2025-02-13 19:38:48,259 - INFO - Successfully created pull request:
https://github.com/SE-course-serc/project-1-team-9/pull/29
```

## Conclusion

These scripts automate the detection and refactoring of design smells in Java code using AI, ensuring better maintainability and code quality in repositories. Maintain and update the scripts as needed to enhance detection capabilities or update AI models.