ASM Linter - Software Requirements Specification

Needs

- Grading process for CSSE 374 is too slow
- No automatic linting process for design smells exists
- No automatic identifying process for design pattern exists
- No way to identify design patterns without looking at UML
- Design patterns should be recognized by analyzing code
- System should be easily changeable
- Must satisfy CSSE 374 final project rubric
- Extra features required for an A

Features

- Ability to perform cursory style checks
- Ability to perform principle violation checks
- Ability to detect design patterns
- Ability to autocorrect byte code
- Ability to generate UML when analyzing code
- The program is user interactable
- Outputs data in a format readable for the users

Software Requirements

- Uses the ASM Bytecode library
- Coded in Java
- Outputs linter results through the console
- Output linter results in a report file
- Accessible through the console
- Accessible through a friendly UI

Use Cases

Use Case 1:

Name: Run Linter Normal

Actor: User

Description: User checks for good design of their java program

Preconditions: User has run linter program in command line.

Postconditions: Results have been printed to the console

Basic Flow:

System prompts user with a list of checks they can perform

1. User selects checks they would like to perform
2. System prompts user if they would like to generate a UML
3. User selects whether they would like to generate a UML
4. System prompts user for file path of their .class files
5. User enters filepath
6. System displays results of checks

Alternate Flow:

Step 2: User selects Single Responsibility Check a. System prompts user to select maximum methods b. User enters maximum methods Resume Basic Flow at Step 3

Step 2: User selects Poor Naming Convention Check a. System prompts user if they would like to auto-correct the names of the classes, methods, and fields b. User enters answer Resume Basic Flow at Step 3

Step 4: User answers that they would like to generate an UML a. System prompts for output directory of UML b. User enters output directory Resume Basic Flow at Step 5