



ESCUELA SUPERIOR
PÓLITECNICA DEL LITORAL

WORKSHOP OF CODING STANDARDS

AUTORES:

- AGUILAR MORA OSWALDO
- BERMUDEZ MOREIRA KAREN
- BERNAL MOREIRA GUILLERMO
- ORTIZ HOLGUIN EDUARDO
- WONG PAVON HUGO

SUBJECT: ING. SOFTWARE II

TUTOR: DR. MERA CARLOS

DEADLINE: 2020/06/11

1. Abstract

This document contains the technical report corresponding to the first group workshop called **"WORKSHOP OF CODING STANDARDS"** of **GROUP#4** belonging to the **SOFTWARE II ENGINEERING** course of **2020-PAO I**.

The report contains a description of the workshop, the code, the JAVA implementation and the checklist reports.

2. Description

Checkstyle is an open source tool that enforces coding conventions and best practice rules for Java code. It automates the process of checking Java code to spare humans of this boring (but important) task. This makes it ideal for projects that want to enforce a coding standard.

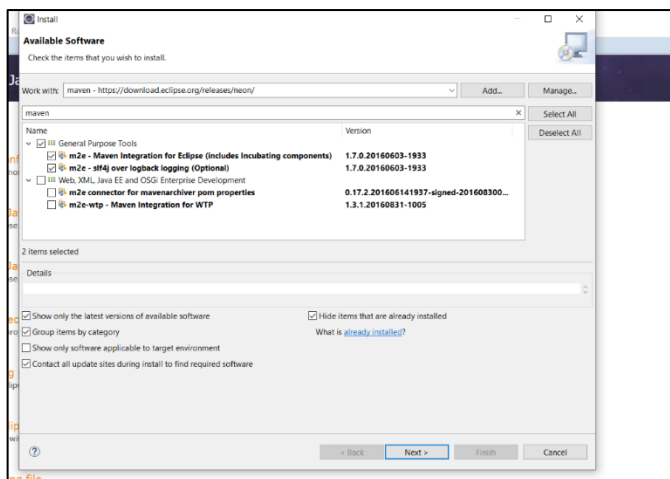
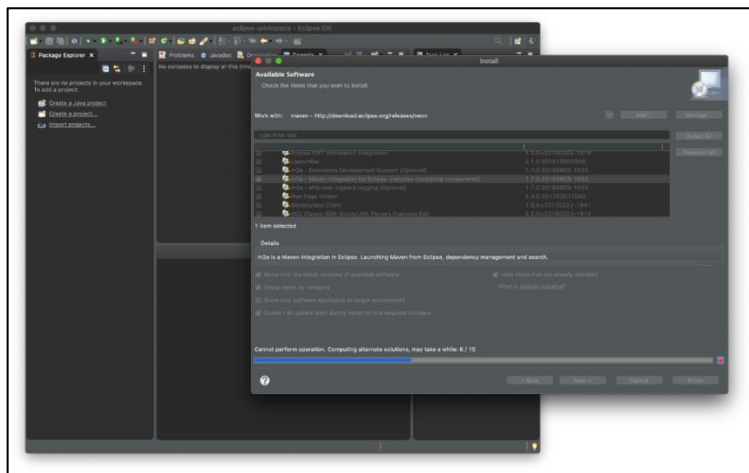
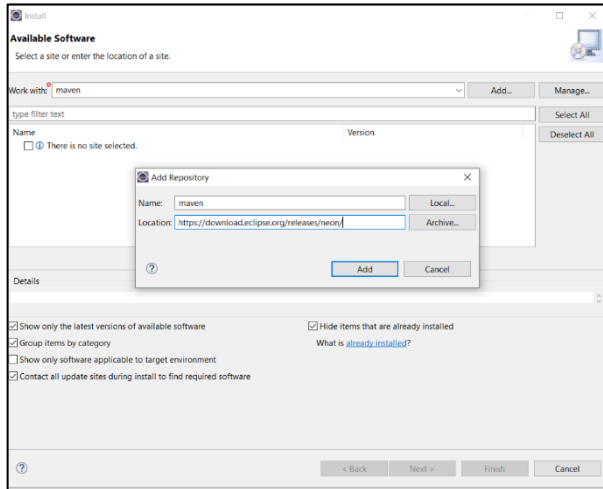
Checkstyle can check many aspects of your source code. It can find class design problems, method design problems. It also can check code layout, formatting issues and generate projectwide reports that summarize the breaches found. Checkstyle provides many checks that you can apply to your source code:

- Annotations
- Block Checks
- Class Design
- Coding
- Headers
- Imports
- Javadoc Comments
- Metrics
- Miscellaneous
- Modifiers
- Naming Conventions
- Regexp
- Size Violations
- Whitespace

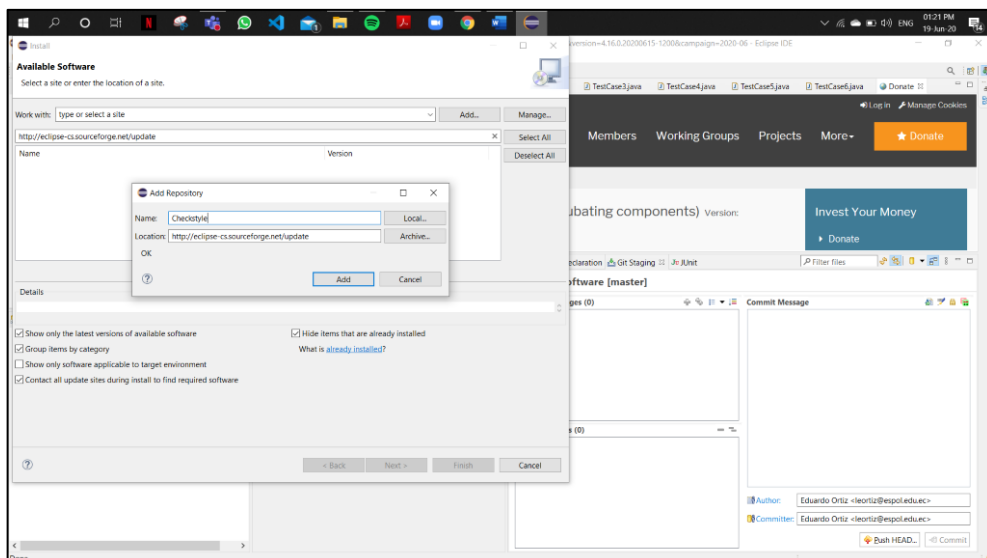
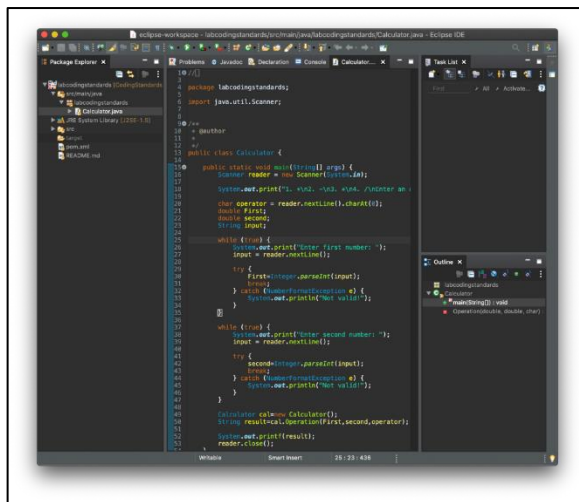
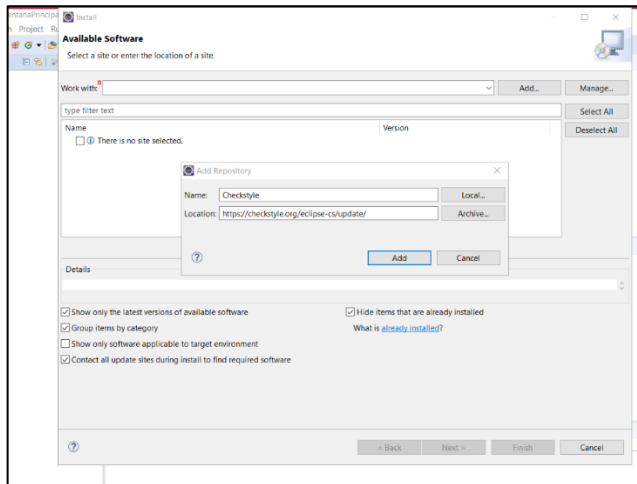
There are several ways to use this tool, including the command line and build automation tools. In this lab, Eclipse will be used in conjunction with Maven to facilitate the use of Checkstyle.

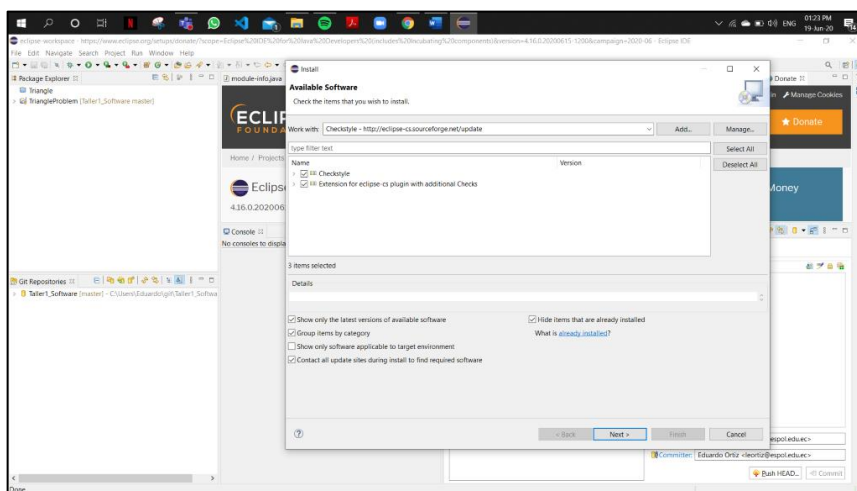
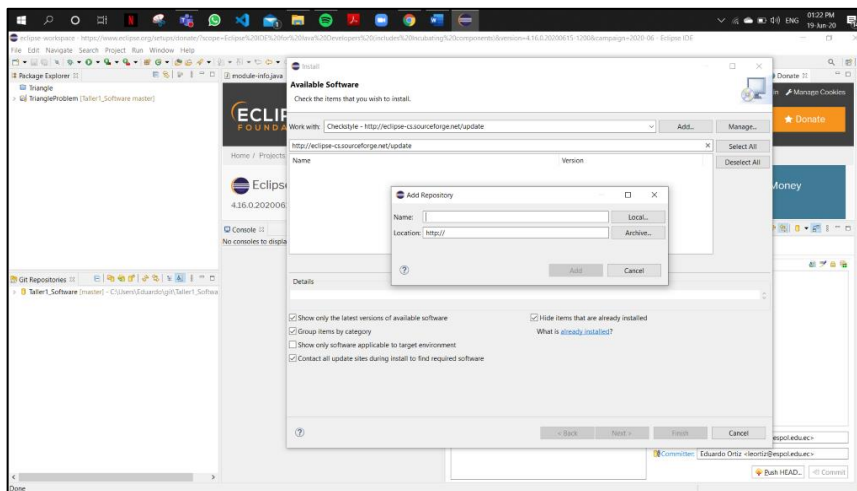
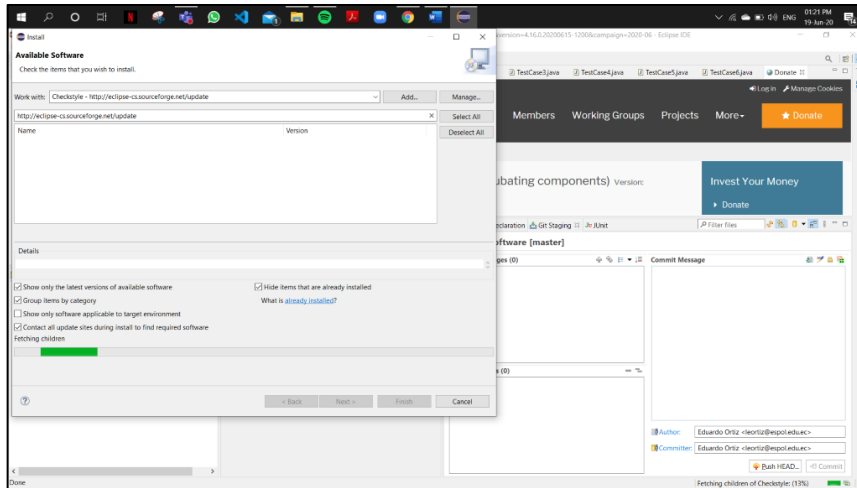
3. Activity Screenshots

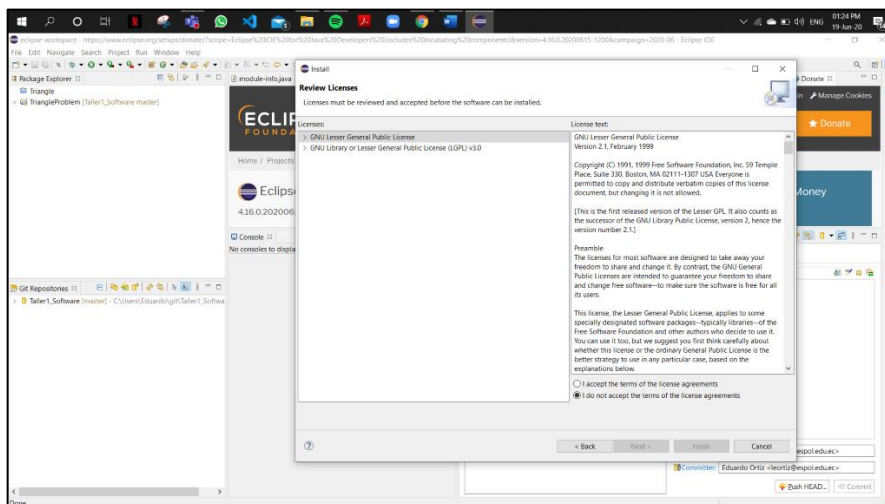
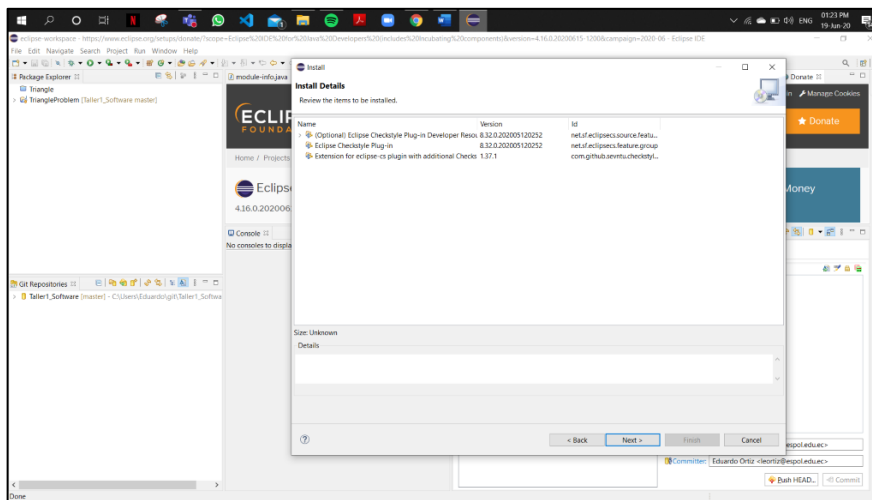
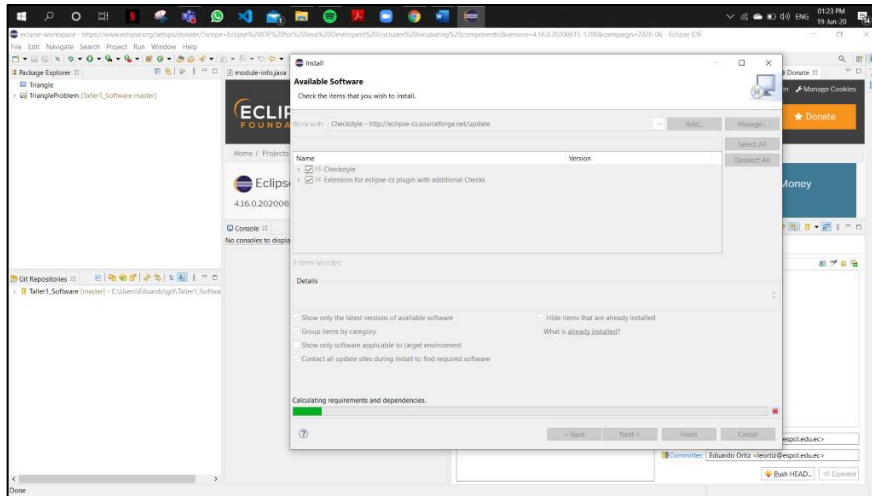
3.1. Part 1: Install Maven for Eclipse.

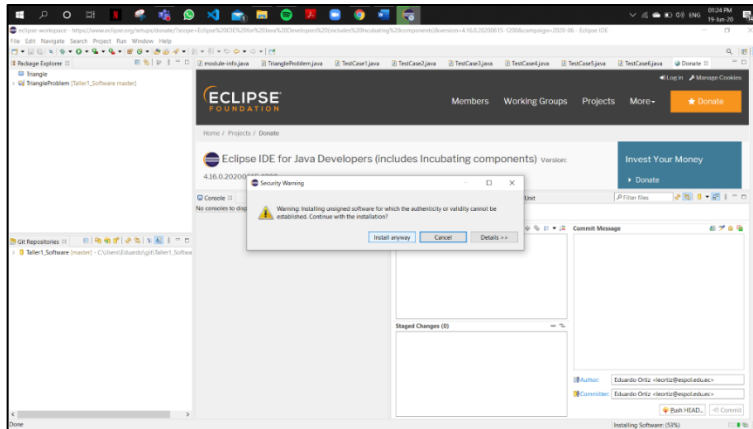


3.2. Part 2: Install the Eclipse Checkstyle plug-in









3.3. Implementation

```

1 team4_checks.xml
2 <?xml version="1.0" encoding="UTF-8"?>
3 <!DOCTYPE module PUBLIC
4   "-//Checkstyle//DTD Checkstyle Configuration 1.3//EN"
5   "https://checkstyle.org/dtds/configuration_1_3.dtd">
6 <module name="Checker">
7   <module name="Header">
8     <property name="header" value="// Copyright (C) 2020\n// ALL rights reserved"/>
9   </module>
10  <module name="TreeWalker">
11    <module name="LocalVariableName">
12      <property name="format" value="^[a-z]([a-zA-Z0-9]+)*$" />
13    </module>
14    <module name="MissingJavadocMethod">
15      <property name="scope" value="private"/>
16      <property name="allowMissingPropertyJavadoc" value="true" />
17    </module>
18    <module name="JavadocType">
19      <property name="authorFormat" value="\s" />
20    </module>
21  </module>
22 </module>
23 </module>

```

```

1 /**
2  *
3  *
4  * package labcodingstandards;
5  *
6  * import java.util.Scanner;
7  *
8  *
9  * /**
10  *  * @author
11  *  *
12  *  */
13 public class Calculator {
14   //CHECKSTYLE:OFF
15   public static void main(String[] args) {
16     //CHECKSTYLE:ON
17     Scanner reader = new Scanner(System.in);
18
19     System.out.print("1. +\n2. -\n3. *\n4. /\n\nEnter an operator: ");
20
21     char operator = reader.nextLine().charAt(0);
22     double first;
23     double second;
24     String input;
25
26     while (true) {
27       System.out.print("Enter first number: ");
28       input = reader.nextLine();
29
30       try {
31         first = Integer.parseInt(input);
32         break;
33       } catch (NumberFormatException e) {

```

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <!DOCTYPE module PUBLIC
3   "-//Checkstyle//DTD Checkstyle Configuration 1.3//EN"
4   "https://checkstyle.org/dtds/configuration_1_3.dtd">
5 <module name="Checker">
6   <module name="Header">
7     <property name="header" value="// Copyright (C) 2020\n// All rights reserved"/>
8   </module>
9   <module name="TreeWalker">
10    <module name="LocalVariableName">
11      <property name="format" value="[a-z]([a-zA-Z0-9]+)*$"/>
12    </module>
13    <module name="MissingJavadocMethod">
14      <property name="scope" value="private"/>
15      <property name="allowMissingPropertyJavadoc" value="true"/>
16    </module>
17    <module name="JavadocType">
18      <property name="authorFormat" value="|S"/>
19    </module>
20    <module name="MethodName">
21      <property name="format" value="^[a-z]([a-zA-Z0-9]+)*$"/>
22    </module>
23    <module name="WhitespaceAfter">
24      <property name="tokens" value="COMMA, SEMI"/>
25    </module>
26  </module>
27 </module>
28
29 </module>

```

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3   xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
4   <groupId>com.espol.euw</groupId>
5   <artifactId>labcodingstandards</artifactId>
6   <version>0.0.1-SNAPSHOT</version>
7   <name>LabCodingStandards</name>
8   <description>Lab Coding Standards, Software Engineering II</description>
9   <build>
10    <plugins>
11      <plugin>
12        <groupId>org.apache.maven.plugins</groupId>
13        <artifactId>maven-site-plugin</artifactId>
14        <version>3.7.1</version>
15      </plugin>
16      <plugin>
17        <groupId>org.apache.maven.plugins</groupId>
18        <artifactId>maven-project-info-reports-plugin</artifactId>
19        <version>3.0.0</version>
20      </plugin>
21    </plugins>
22  </build>
23  <reporting>
24    <plugins>
25      <plugin>
26        <artifactId>maven-checkstyle-plugin</artifactId>
27        <configuration>
28          <configLocation>team4_checks.xml</configLocation>
29        </configuration>
30      </plugin>
31    </plugins>
32  </reporting>
33 </project>
34
35

```

Overview Dependencies Dependency Hierarchy Effective POM pom.xml

Problems Javadoc Declaration Console

```

<terminated> CodingStandardsS12 [Maven Build] C:\Program Files\Java\jre1.8.0_211\bin\javaw.exe (Jun 19, 2020, 10:04:52 PM)
[INFO] BUILD SUCCESS
[INFO] Total time: 12.853 s
[INFO] Finished at: 2020-06-19T22:05:07-05:00

```

4. Conclusions

- This plug-in is a usefull tool to the implementation.
- Checklist improve the process of error identification.
- In case of reboot of eclipse the checklist should be activated again.