TDDC88/TDDC93: Software Engineering
Lab 6
SOFTWARE METRICS

Objectives:

- To give you a fundamental understanding and practical experience with different code metrics

- To get familiar how code metrics works

- To understand why code metrics are important for quality products

IntelliJ Installation:

On Terminal: module add prog/idea and then start it with idea.sh or ./idea.sh
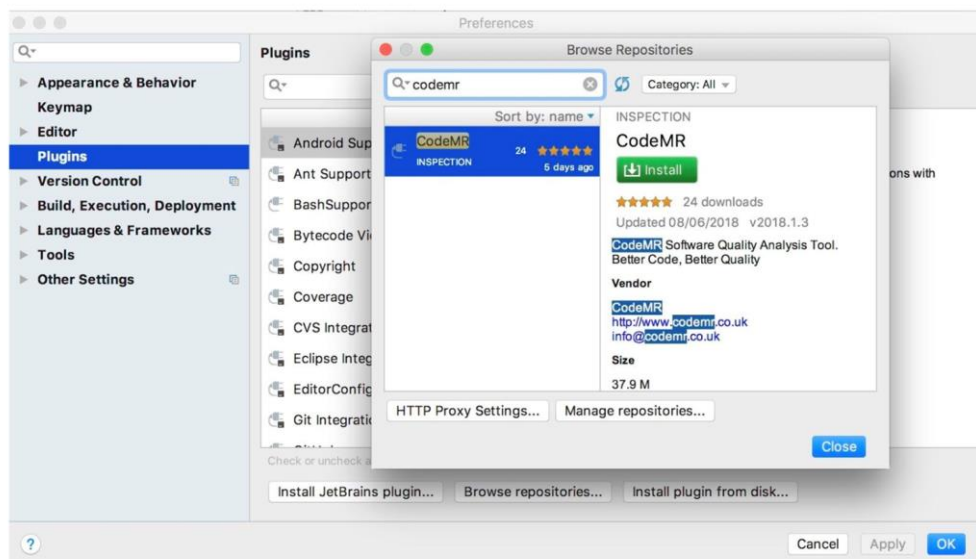
CODEMR: The Plugin

CodeMR is a static code analyser for Java, Kotlin and Scala projects. It helps software developers and team to develop better code thus leading to increase software quality. CodeMR visualizes code metrics and high level quality attributes (coupling, complexity, cohesion and size) in different views, such as Package Structure, TreeMap, Sunburst, Dependency and Graph Views.

Installation:

Manual install:

Download CodeMR plugin to your local file system from CodeMR website (https://www.codemr.co.uk/) or from IntelliJ Plugin repository site (https://plugins.jet-brains.com/).

●Select IntelliJ File Menu→Settings→Plugins→Install Plugin from disk

NOTE: Restart the IDE after installing the CODEMR plugin. On successful installation, you would be able to see the "CodeMR" as an option/menu in your IDE.
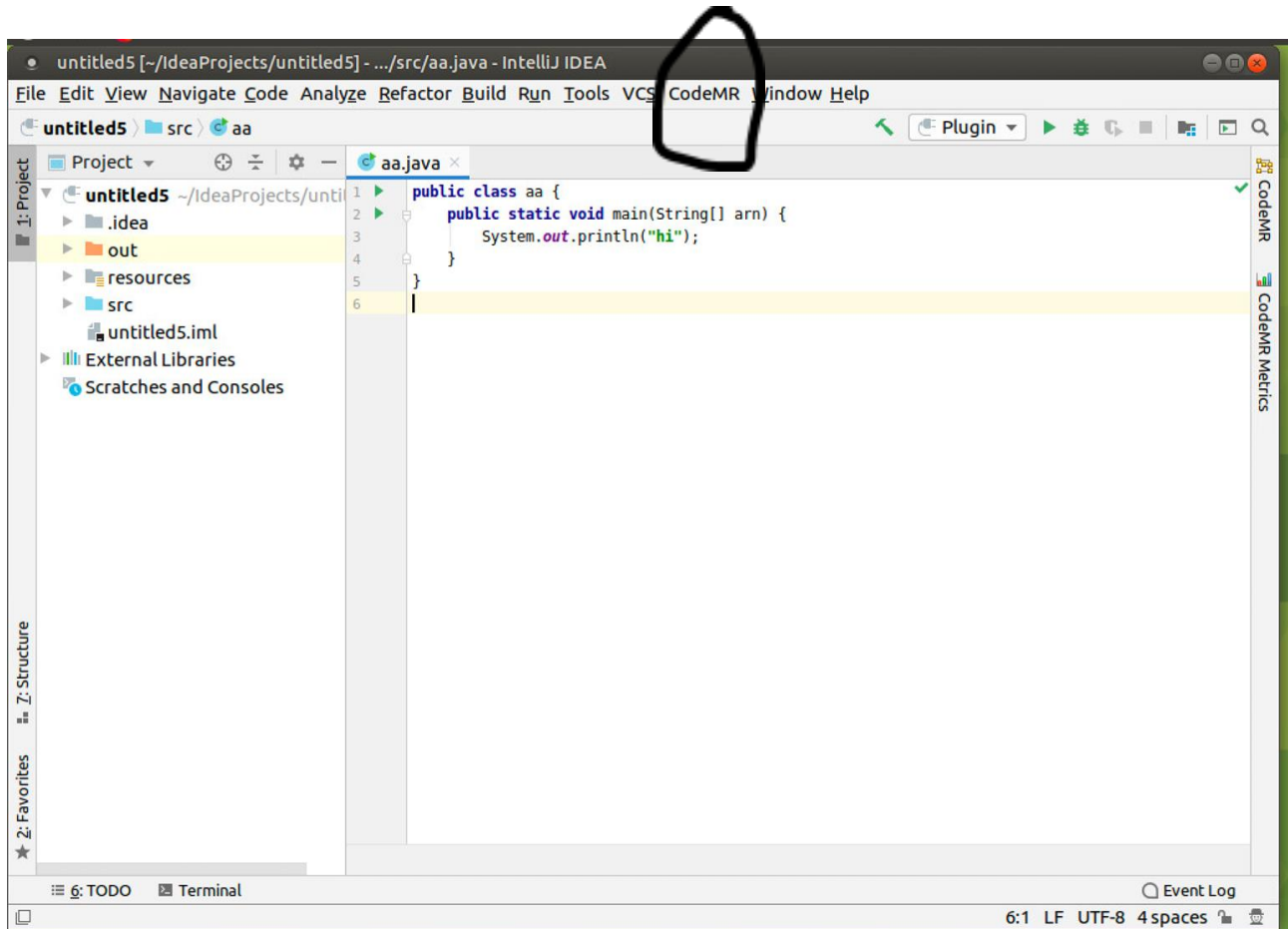
License Update for CodeMR:
We need a trial license for CODEMR because current downloaded version only accommodates 50 classes. Go to https://www.codemr.co.uk/downloads/ and download the trial version. Once the license is downloaded,
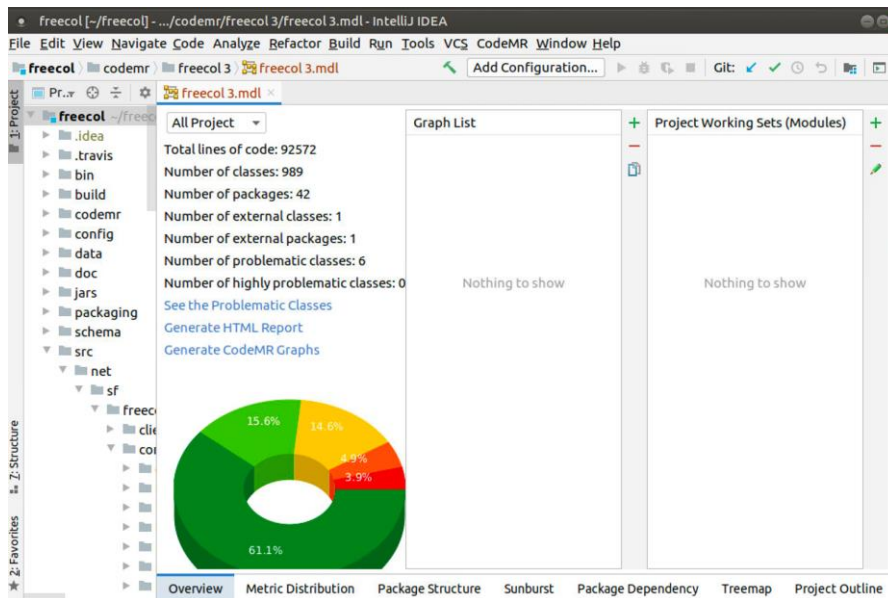
CODEMR menu at IntelliJ → License Management → provide the license, you downloaded and click Activate

NOTE: If the above license does not work or it says it has expired, so I created, for you, link below to access all the metrics on FreeCol. You can visit the following link and can proceed with your tasks. This is the same link, which would have been generated by "codemr" but it is online now.
(https://www.ida.liu.se/~TDDC88/labs/Labs2020Uploaded/Lab6/codemr/freecol%203/html/main_report/htmlx/lbd/dashboard.html).

Download Freecol:
1. Clone the free col repositories from https://github.com/FreeCol/freecol.git
2. Open the freecol project in IntelliJ

3. There are 3 ways to install CodeMR:
   a. Eclipse
   b. IntelliJ
   c. Using this link directly: (https://www.ida.liu.se/~TDDC88/labs/Labs2020Up-loaded/Lab6/codemr/freecol%203/html/main_report/htmlx/lbd/dashboard.html).

CODEMR Model Editor:

We have cloned the freecol project and installed the CodeMr plugin.

The first step is to create a model with CodeMR. In simple words, this model will go through the files of freecol and create different metrics. It displays different quality views of projects.

1. Select a folder (freecol on the right side in IntelliJ) in the project, click CODEMR menu and select "Extract Model for Java"

2. You can specify a name for the model and also select basic options for the model extraction.

After you click extract model, it will take few second to generate the results. You will see the following page:

In addition to above following pages, you can access the web page of all information about all metrics on the following folder:

freecol/codemr/freecol/html/main_report/htmlx/lbd/dashboard.html
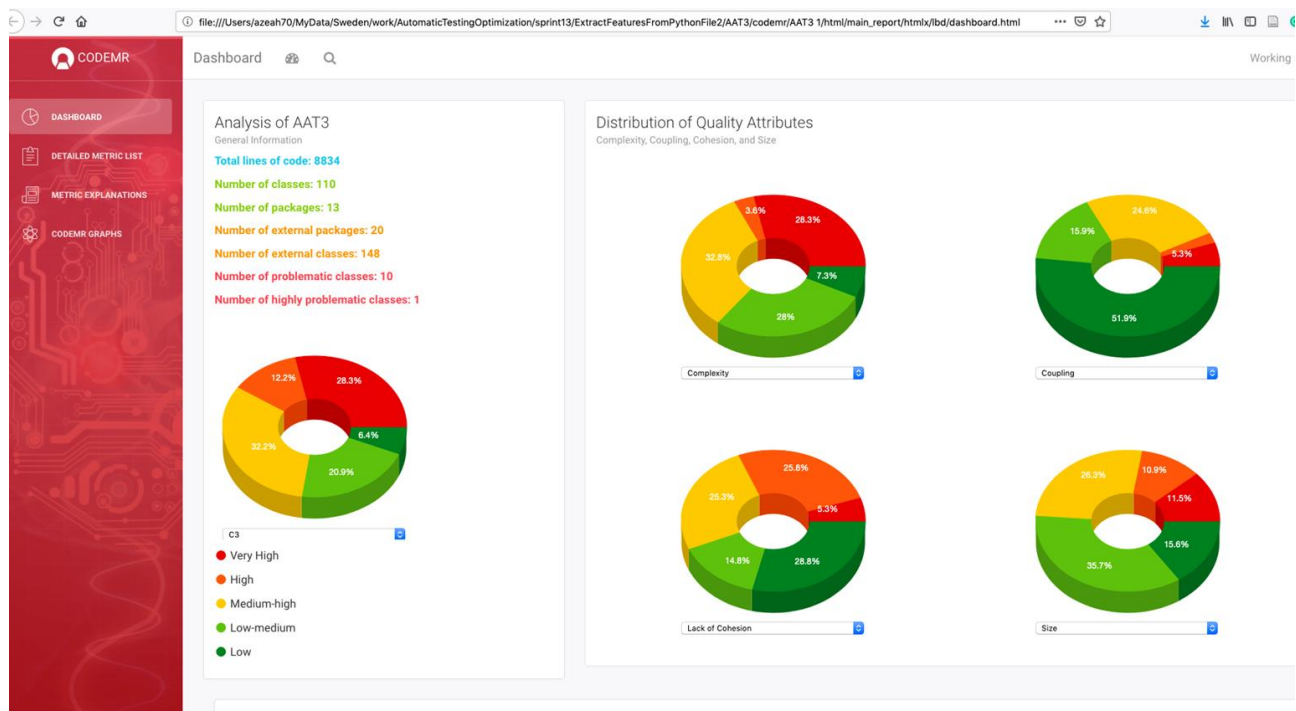
Exercises:
Task 1:
Explore the website of metrics, generated by CODEMR and explain the following questions:

1. Do you think that website provides some help in terms of software quality?
2. What type of information, the website provides and how can you use this information to increase software quality?

3. Can you name 5 – 6 different metrics, available in the website? And what does they do?

Task 2:

Which school of thought, from the following, you belong to? Motivate and justify your choice by writing a half page. You may compare both choices with pros and cons, if you want.

1) If system is working according to requirements, no matter, how we wrote a software code, it is a good software.
2) A good quality software does not only perform desired functionality but also written with care and good coding conventions and styles.

Task 3:

Your task is to perform a study of software quality factors of the open source project FreeCol, and from that write a short report (about 2 pages). The study should be done as follows:

- Select **ONE** software quality factor of your choice from the following list as a basis for your study.
    - Changeability
    - Testability
    - Understandability
    - Maintainability
    - Security
    - Availability
    - Scalability
    - Replicability

- Decide and describe **three** software metrics, (**You can find which metric does what on left side of website named "Metric Explanation"**) from the output of CODEMR which, you will use to evaluate the quality factor, identified above, and which measurements you will need to perform on the code base to get the metric.

- You need to describe why your chosen metrics will give you a good estimation of the quality factor, and also how the metric is calculated.

- Perform measurements on FreeCol using the CODEMR and find locations in the code where your metrics has especially high or low values.

Task 4:

Using your study and based on the extreme points you found, you will then write a discussion where you discuss the following questions:

- What is causing the metric to have a particularly high/low value in the code locations indicated by your extreme points?

- Are the areas of very high or very low software quality or are the values caused by something else?

- Are you able to draw any conclusions regarding software quality factors in FreeCol? Why/why not?

- Do you know of any other metrics handling the same software quality factor as your chosen metric? How are they different from the metric you used?

- Do you think other metrics are required in order to get a good estimation of the software quality factor of the product?

- Do you have any suggestions for improvement in the locations indicated by your study? This could be either on a method or class level or a larger structural change.

- Do you think your study gave you a better understanding of the code base? Why/why not?

Examination: Write a report by justifying your answers/choices send the report to your instructor's urkund <u>address</u>. Include all the answers of all the tasks above.