SENG 371 – Spring 2024
Assignment 01 – Small Changes

(Initial Draft of the Assignment submitted on Jan 27, to be improved until Jan 31)

This group assignment explores the evolution of the JabRef open source software system via private group repositories on GitHub. By the end of this assignment, you will:

- Clone a full replica of a software system from a private repository in GitHub;
- Explore the repository in your working directory using an IDE;
- Locate concepts in the source code of a software project;
- Analyze the impact of a change request to help estimate its cost;
- Code simple small changes to a software system using an IDE;
- Submit the result of your changes by means of pull requests to a private group repository;
- Review and accept changes in a private group repository.

Clone your JabRef private group repository

In this assignment, you will first clone your private group repository, which is called `jabref-seng371-lXgY`, where X is the number of your lab section (1, 2 or 3) and Y is the number of your group (1, 2, 3, 4 or 5). This will be your work repository for all three assignments in this course.

To clone your private group JabRef repository, please use the following command (or use the URL below that starts with git@github.com to clone it from your IDE):

```
git clone --mirror git@github.com:uvic-seng371/jabref-seng371-lXgY.git
```

Then, you can import the working directory from the main branch by using your IDE importing features as you did on Lab 2. Finally, run the `gradlew` scripts as you also did on Lab 2 to have a working project.

(Obs.: If you are not sure of your group number, please use the Microsoft Teams course team and lab section channel to recover this information. You must have provided your GitHub user name to your lab TA in order to be added to both your private group channel and to your private group repository.)

Three JabRef issues (change requests)

Here are the JabRef issues (change requests) your group will solve and code during this assignment:

Enable proper pasting of HTML code into fields (by converting to Markdown) #10558

https://github.com/JabRef/jabref/issues/10558

Entry creation from DOI does not recognize URL encoding, but "get bibliographical data" does #10648

https://github.com/JabRef/jabref/issues/10648

Export to MS Office 2007 (*.xml) of @electronic and @online records *mistakenly duplicates* BibTeX 'Title' to MS Office 'Publicationtitle' field #10807

https://github.com/JabRef/jabref/issues/10807

You will code your changes in your private group repository, `jabref-seng371-lXgY`. But, of course, you may interact with the JabRef team by using the URLs above to ask questions and clear doubts about each issue (the JabRef team is made of volunteers that are generally busy persons that may answer or not to your questions and comments, so please do not rely on this as a strategy for problem solving; but as you may see from some issues, they sometimes answer questions).

Your group must split the three issues to be worked by three pairs (or two pairs and one person, in case your group has five members). Each pair will be responsible for coding and submitting the code for their assigned issue. Of course, the other group members from a different pair may help another pair from the same group, it is up to your group to decide how you will work. However, we will look at code commits to assess individual contributions. Please, let your Lab TA know which pairs have been formed during this assignment.

Concept Location and Impact Analysis

Before you solve your assigned issue, try to locate the concepts from that issue. Using a search strategy of your choice, identify the class(es) where the concept is located.

Yet before you solve your assigned issue, try to analyze the impact of that issue. Your concept location must have produced an initial impact set, i.e., a small set of classes where the concept in the change request is located in the source code. Starting from this initial impact set, analyze the potential impact of the change as if you were performing the change, and compute the final impact set of the change, i.e., a final set of classes that you believe will be affected by the change. Briefly record your steps to later show the TA the process you followed during impact analysis.

Actualization

Before you start changing code, create a branch and checkout the new branch, so that you do not work on the main branch during your issue solving.

Your pair must perform the changes in the source code in order to solve the issue. Other group members may help with suggestions. We suggest that your pair works with pair programming (either local or remote pair programming) because the type of communication that pair programming requires usually leads to deeper learning and better coding choices.

You may commit parts of your change into your branch, before you send the full change to the main branch. This may be useful because you may split your change into smaller steps.

Submit a pull request to your private repo

After your pair finishes coding the solution for the issue, submit a pull request back into the main branch of your private group repository.

Other members of your group should review your pull request and either accept it (if they feel it is correct and complete) or comment on it and suggest changes to make sure your pair solved the issue.

After the reviewer(s) accept your pull request, let the TA know that one of the issues has been solved, so they may start grading.

Assessment

Your grade in this assignment will have two components:

- A grade based on the changes performed on your private group repository;
- An individual grade based on your group's assessment of your contributions to this assignment.

The first grade component will be assessed by your lab instructor. They will analyze your group changes in your private group repository. Accepted changes will be their main source of analysis.

The second grade component depends on your group's assessment of each group member's work. The default assumption is that everyone carries equal eight in this project. However, grade differentials may be assigned to members of the same group if contributions are imbalanced. Each group member will submit (via Brightspace until

the assignment deadline) an assessment of themselves and the other group members. This simple submission only needs to state which percentage of the group work in this assignment each group member did. Your Lab TA, then, will use the information from all group members' submissions to decide on a weighting factor (less than or equal to one) that will multiply your first grade component to lead to a final individual grade. You Lab TA may also assign bonus points to one or more students according to their contributions to the assignment.

Can your group submit a pull request back into the original JabRef repo?

Well, that is our wish, and we hope it is your wish as well: to become an open source contributor. But there is a catch here. You can only submit a pull request into the original JabRef repo after the assignment deadline. This is necessary in order to avoid plagiarizing. **Please know that if any of your group members submits a pull request back into JabRef before the assignment deadline, that particular issue will be graded with ZERO for all group members**.

But, then, after the assignment deadline, both our teaching team and the JabRef team will be happy to see your contributions on the original JabRef repo. The JabRef team will have to both analyze students' pull requests and choose only one of the pull requests (since we have various groups working on the same issues), so we cannot assure that your pull request will become part of JabRef source code. We hope, though, that some (or most) students will see their code there after the three assignments of this course finish.