Software Engineering

The S-Team

Project Phase 2 - Merged Document

Link to demo video: *https://youtu.be/05hD_UCXna0*

Team members:

1. *Miguel Real (55677)*

2. *Gonçalo Virgínia (56773)*

3. *João Vieira (56971)*

4. *Guilherme Pereira (57066)*

5. *Gabriela Costa (58625)*

Task - User Stories 1

This user story was done by team members 56971 and 57066.

User Stories

- As a user, I want to search for a specific topic in Google Scholar through JabRef;

Use Case:

Name: SearchGoogleScholar

Id: US1

Description: A user can search for any topic in Google Scholar through JabRef.

Actors:

Main - User

Task - User Stories 1.1

This user story was done by team member 56773.

User Stories

- As a user, I want to add a Google Scholar article as a filled in entry into JabRef;

Use Case:

Name: NewEntryFromGoogleScholar

Id: US1.1

Description: A user can add a new filled in entry by specifying inputting a Google Scholar article name.

Actors:

Main - User

Limitations:

1.0 and 1.1:

The search windows give an error when clicking "Cancel".

1.1:

The GUI button doesn't appear greyed out when a library isn't available.

The search gives an error if the parser used doesn't find one of the fields (e.g., authors).

Only a limited number of fields gets filled in.

Formatting is limited to what Google Scholar shows, which makes parsing for information like the year of the article very difficult.

Articles, books (etc.) are not differentiated.

Feature 2

This user story was developed and implemented by team members 55677 (Miguel Real) and 58625 (Gabriela Costa).

Feature description

Our feature allows users to mention entries in the comments of other entries. Upon typing a designated character (we chose an @), all the characters that follow (and are not a space) will be interpreted as an entry's citation key. If an entry with that citation key exists in the currently selected library, the String containing the @ as well as the citation key will be turned into a hyperlink, which can be clicked to open the tagged entry in edit entry mode.

User Story

As a user, I want to be able to type an entry's citation key in the comment of another entry, so that I can click it to access the tagged entry in edit mode.

Use Case

Name: LinkEntryInComment
Id: US2
Description: The user types an @ followed by an entry citation key, creating a link to the referenced entry automatically.
Actors:
    Main: User
    Secondary: None
Pre-conditions: the citation key belongs to an entry in the current library and the format is as follows: @citationKey
Main flow:
1. The user types in an @ followed by the citation key of an existing entry
2. The @ and the citation key become a hyperlink
3. The user clicks the hyperlink
4. The referenced entry is opened in the entry editing tab

Alternative flows:
1. The user types in an @ followed by a space character or a sequence of characters that does not represent a citation key belonging to an existing entry
2. The sequence is not recognized and an hyperlink isn't formed

Disclaimer

While User Stories and Use Cases were provided for this feature, it was not implemented in its entirety for several reasons, which will be elaborated upon in this disclaimer. This was not a formal requirement, however we felt that an explanation would help contextualize our situation and shed some light on all the work done, even if that work was not deliverable or included in the code.

The JabRef developers extended the JavaFX TextField and TextArea classes to make their text input boxes, with the Comments box being a TextArea (or rather, a subclass of it - EditorTextArea, but for the purposes of this disclaimer that's irrelevant). The purpose of a TextArea in JavaFX is to provide an area for a user to write and edit text that takes up more than a single line. This means that it does not provide support or integration for Rich Text, which is a prerequisite for the usage of hyperlinks, the simplest way to make text clickable.

This presented a fundamental problem in the suggested feature, as there is no simple way to implement it without changing the comments Text Area to something else, namely something that provides support for RTF and hyperlinks. In JavaFX there are several text display components that support hypertext, as well as several components that support text editing, however there is none that does both, in the way we needed it to. We were left with several choices: we could try to find another third-party library that provided a rich text editor with hyperlink capability and then try to integrate it into JabRef, replacing the TextArea; we could try to find a workaround for our problem that allowed us to simulate our feature as best as possible with the included libraries; or we could try to change our feature requirements to something more feasible. Ultimately all of these were considered and most of them attempted, yet none yielded satisfactory results.

First, a workaround was attempted where a VBox (read only JavaFX component that supports hyperlinks) was put over the Text Area displaying the hyperlink as soon as the citation key was recognized. This proved to be very challenging under time constraints and very limited JavaFX knowledge. In order to access the editor tab where the comment area is, from a class with enough information to access entries across a library, necessitated other workarounds, many of them being less than ideal, introducing code smells like Feature Envy, Message Chain, and others.

Then, an attempt to replace the Text Area with a Rich Text editor was made, and while the chosen replacement was JavaFX based, it did not have the required features and functionality to work as we intended, as the JabRef code depended on missing features it did not include. To mend this problem by extending the editor or changing JabRef would be a monumental task for a very small team under severe time constraints and other issues, as well as leading to Code Smells. This approach was therefore abandoned as well.

Finally, a change to the feature was considered, however with very little time remaining until the deadline and many tasks still left to be completed, the team could not agree on a suitable replacement nor would the change in requirements be approved in time for research, implementation, testing and debugging. Therefore, the team agreed to deliver what they had, to complete the other deliverables as best as possible and to explain the situation.

To further complicate the entire situation, two of the five team members - both responsible for the development of this feature - were struck by COVID-19 shortly before the 24th of December, being left to care for infected family members and eventually falling ill themselves. This drastically affected the workflow, morale, and productivity of the team.

Location

To implement this feature, changes were made in the following location:

Starting in line 884 and ending in line 927: src/main/java/org/jabref/gui/LibraryTab.java