Updated requirements

Radu Vasile, Paul Stan, Rebecca Andrei, Stefan Bud, Calin Georgescu

May 2024

1 Requirements

1.1 Functional Requirements

1.1.1 Must-have Requirements

- Users must be able to upload a PDF to the tool.
- The tool must only accept PDF files, not other types.
- The tool must be able to recognize both single-column and double-column PDF's.
- The tool must convert the PDF text into a machine-readable format[1].
- The machine-readable format must indicate the position of the annotations throughout the PDF, as well as distinctly separate them from the rest of the text.
- The plain text must not contain the title page, references section, images and tabular data.
- The caption of images and tables must be separated from the rest of the text, to avoid confusion.
- The tool must have a Graphical User Interface, to facilitate usage.
- The GUI must display the uploaded PDF, to allow users to make changes to files directly.
- Users must be able to add annotations through the GUI.
- The tool should allow users to delete annotations.
- Users must be able to download the updated version of the PDF, either in machine-readable format or in PDF format.

[1] The machine readable format specified in this document consists of plain text, which clearly separates text, annotations and captions

- The tool must be able to accept multiple files as input, process them all, and create a database containing all results.
- When inputting multiple files, the tool must be able to recognize the format from the Brightspace export tools.
- Users must be able to view a list of codes with predefined feedback, and select them as annotations
- Users must be able to add / edit / delete codes.

1.1.2 Should-have Requirements

- The coordinators should be able to create accounts and log in, to separate workload.
- The tool should allow users to view all the uploaded PDF files simultaneously.
- The tool should allow coordinators to view all their corresponding PDFs.
- The currently displayed PDF file should be selected through a menu.
- The tool should display the current progress when performing costly operations, in the form of a progress bar.
- Users should be able to save their work through the tool.
- Users should be able to choose different formats for new annotations written, such as text size, font, or style.
- The tool should support different annotation types such as annotations on text or free annotations.

1.1.3 Could-have Requirements

- The tool could accept more than two different PDF formats.
- The tool could allow users to copy results to their clipboard, aside from inserting them into a database.
- The tool could allow for quick access of files, based on a recent history.
- The users could indicate the document's current status, such as whether it requires further additions, or if it is marked as completed.
- The tool could be able to render PDF files in real-time.
- The users could organize the uploaded files into multiple sections.
- The tool could process only the files from a specific section.

1.1.4 Won't-have Requirements

- The users should be able to confirm their identity using a verification method.
- The users should be able to choose from multiple themes, to customize the aspect of the application.
- The users could collaborate with other users, on adding annotations to the same file.
- The shared files must be saved in real-time.
- The tool could accept multiple file types, not just PDF files.

1.2 Non-Functional Requirements

- The tool should be written in Java 17.
- The tool should have a minimum of 80% test coverage.
- The tool should have a minimum of 75% mutation coverage.
- The tool should abide by all privacy and copyright laws in effect, as of 25th April 2024.
- The tool's source code should be open-source.
- The tool should have a processing time of a maximum of 30 seconds per PDF.
- The code should be written in such a way that it allows easy extension.