

Bachelor's Thesis

A C# Tool to Post-process and Format Batches of Exam Submissions

Dipl.-Ing. Markus Weninger
Institute for System Software

Student: Wolfgang Irrgeher

Advisor: Dipl.-Ing. Markus Weninger, BSc.

Start date: February 2021

P +43-732-2468-4361

F +43-732-2468-4345

markus.weninger@jku.at

The coronavirus outbreak in 2020 strongly changed the way how teaching is performed at universities. Most classes switched to an online teaching mode – including digital exams. While digital exams may sometimes even prove more useful than “paper exams”, for example by providing syntax highlight when writing source code, they also have certain downsides.

The Institute for System Software has developed the software Xaminer that can be used to perform online exams. Unfortunately, exam submissions are only stored as text files, which results in two problems: Including graphical submissions (such as drawings) is currently not possible, and printing the submission “as is” often results in bad-placed page breaks.

The goal of this thesis is to develop a C#-based post-processing that can be used to produce formatted and well-aligned submission files that can be printed out for correction. The tool should be able to load a batch of exam submissions, i.e., text files that contain answers in the form of *free text*, *SVG graphics*, or *choices*. Once such a set of submissions has been loaded, they should be sorted based on the students' last names, and a filter functionality should allow to filter the submissions based on a given text. The user should be able to step through the different submissions one-by-one (i.e., the tool should have a “Next” and “Previous” button). The formatted and aligned “print preview” of the current submission should be shown. Following a few guidelines and corner cases that have to be tested:

- A page break should never appear within an exercise (except for the exercise being longer than 1 page).
- Long text and images must not be cut off (neither horizontally nor vertically)
- Choice answers should be displayed as checkboxes (also check possible cut offs)
- Each exam should start with the student's name, student ID, and study code, nicely formatted.
- Each exercise should start with a heading such as “Exercise #3:”, i.e., the printed version of the exam should have a “nice look”.

The tool should provide features to (a) save a single formatted submission, (b) print a single formatted submission, (c) save all loaded submissions as formatted, (d) print all loaded submissions as formatted. We suggest to produce an HTML file from each submission's text file. There, it is possible to specify `style="page-break-inside: avoid"` on a block, for example on a `div` region, to prevent page breaks with that block during printing. Such a block could be placed around every answer. This HTML file can be used as a source for the print preview within the tool, as well as the printing and file saving. Nice-to-have features would also be to find suitable page-break points within a single exercise (for example, between two methods of a coding exercise) or to also allow to place page breaks manually.

Modalities:

The progress of the project should be discussed at least every two weeks with the advisor. A time schedule and a milestone plan must be set up within the first 3 weeks. It should be continuously refined and monitored to make sure that the thesis will be completed in time. The final version of the tool has to be finished before start of June 2021 (to make it available for the exams) and the thesis must be submitted not later than 15.08.2021.