JOHANNES KEPLER
UNIVERSITY LINZ

Master's Thesis

Workflow enhancements for the online examination tool Xaminer

Student: Alexander Kainzinger, BSc
Advisor: Dipl.-Ing. Dr. Markus Weninger, BSc
Start date: October 2023

Dipl.-Ing. Dr. Markus Weninger, Bsc
Institute for System Software
T +43-732-2468-4361
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 teaching mostly went back to teaching in presence, a number of exams are still performed digitally. For this, the online examination tool Xaminer has been developed at the Institute for System Software at the JKU. Since its first version, the tool has been expanded into various directions but still has potential to provide more quality-of-life features, for users as well as for the developers.

Thus, the goal of this thesis is to target these quality-of-life features as well as the system's architecture to increase its easy-of-use as well as its simplicity of hosting the system. Overall, the enhancements can be divided in the following categories, which should be tackled in the given order:

- *Modernize Frontend Stack*: Even though Xaminer constantly received new feature, only little time has been spent on keeping its tech stack up-to-date. For example, Xaminer was built using the frontend framework Vue.js 2. Yet, since September 2020 Vue.js 3 is available. Thus, old libraries and frameworks should be brought up-to-date and should be checked for compatibility.
- *Deployment and Hosting*: Since Xaminer was first released, its frontend and backend have been strongly coupled. It is thus not possible to restart the one without the other. Part of this thesis is to de-couple both entities, setting up Docker and Docker Compose settings for both, and to run automatic tests using GitHub Actions upon repository commits („CI pipeline"). A setup guide has to be written that describes how to host Xaminer on a server and how to restart it if necessary.
- *User-side QOL improvements*: Over the years, various „nice-to-have" feature requests have accumulated which should be targeted in this thesis. These include:
  - *PDF export in a single file*: Currently, exam results can be exported in a .zip file containing one .pdf file per student submission. In the future, it should also be possible to export all submission in a single .pdf file that can be easly printed double-sided (one sheet of paper should never contain the submission of more than one student).
  - *Preliminary exam submissions*: Currently, students can only submit the exam once. To prevent data loss, it should be possible to also save exams preliminary.
  - *Questions with subquestions*: One of the most requested features by lecturers is to support questions with sub-questions, e.g., multiple questions sharing the same descripiton and summing up their points to a common end score. This involves a nice editor to define such questions as well as nice formatting during exam participation.
- *Bi-directional server-client communication*: Currently, certain pages have to be manually refreshed to show the most current data, first and foremost the exam admin page. This page is used to monitor which students are still working on their exam and which students have already submitted. In this thesis, this page should be reworked to automatically update when a new submission has been perfomed. The exam supervisor should also be informed with a small popup / notification that a new submission has been performed.

Modalities:
The progress of the project should be discussed at least every three to four weeks with the advisor. A time schedule and a milestone plan must be set up within the first two weeks and discussed with the advisor and the supervisor. It should be continuously refined and monitored to make sure that the thesis will be completed in time. The final version of the thesis is expected to be finished before the 31.10.2024.