Analyzing Software Engineering Courses with Process Mining and Business Intelligence*

Dennis Schmitz, Matthias Feldmann, and Daniel Moldt

University of Hamburg, Department of Informatics
{schmitz,4feldman,moldt}@informatik.uni-hamburg.de

Abstract. Analysis of student performance and resulting support during practical courses is an important topic in the education of students. This contribution illustrates our ongoing work to analyze practical courses with process mining based on Petri nets and business intelligence tools.

Introduction The data-driven analysis of students behavior and performance in university courses potentially provides valuable insights. The tutors get a better overview of the course and of the individual students in order to support weaker students in time and improve appropriate learning materials. The students on the other hand get the opportunity to evaluate their own performance in comparison with the requirements of the teachers. Among other things, they can analytically review their learning strategies.

This work is located in the Educational Data & Process Mining and Learning Analytics domains. The aim of our research is to determine the effects of real-time process mining and business intelligence on students and teachers.

During a one semester practical course we teach the comprehensive Petri net-based, agent- and organization-oriented software development approach (PAOSE) [3,2]. In order to analyze the practical course the technical learning environment used so far was extended by data collection capabilities. This contribution presents our ongoing work on these extensions as well as an extract of the analyzes already carried out.

Data Collection Setup Interesting questions to collect data for are e.g. *How long do the students need for a task?*, *Does the required time diverged from the targeted time?*, *How do the students proceed to solve a task?* or *Does the method of proceeding influence the result?*.

The practical teaching of the PAOSE bases on *process-oriented* worksheets [4]. The worksheets' processes are modeled as workflow nets [1]. On the one hand, a project management tool was set up to collect student data to answer questions such as those mentioned above. In the last years *Jira* and *Redmine* were used. Both provide an issue tracker plus time logging (among a lot of other features)

and store data in a relational database. On the other hand, a Git repository was used to track the results of the students' tasks and additional work behavior.

For each task of the worksheets, tutors provide predefined tickets in the mentioned issue trackers. Students should update the status of the tickets and log their working time. They should also check their work into the Git repository, while the commit message should contain the related ticket ID and the names of cooperating students.

Data Analysis For the data analysis the process mining tools *Disco* and *ProM* and the business intelligence tool *PowerBI* are used.

Analysis of the tickets and the Git data in regard of the students processes shows that the students differ greatly in their behavior when working on a worksheet. While some students tend to learn quickly to work with these tools systematically and organized, others have problems (to accept) to follow the instructions. For the tutors, this is a valuable observation that helps to find out early which students need further assistance.

Checking the conformance of the mined processes against the provided workflow nets of the worksheets, the students can review their behavior and identify where they did not worked systemically and why problems may have arisen.

Comparing the estimated times with the students' logged median time per task with PowerBI shows which of the tasks are over-/underestimated by the tutors. Based on these results the underestimated tasks were enriched by helpful information to ease the solution of the tasks while the estimated time was reduced for overestimated tasks.

The analytical methods that we are currently developing can be used by others to evaluate the behavior and performance of their students in similar learning environments.

References

1. van der Aalst, W.M.P.: Verification of Workflow Nets. In: Azéma, P., Balbo, G. (eds.) Application and Theory of Petri Nets 1997. pp. 407–426. Springer Berlin Heidelberg, Berlin, Heidelberg (1997)
2. Cabac, L.: Modeling Petri Net-Based Multi-Agent Applications, Agent Technology – Theory and Applications, vol. 5. Logos Verlag, Berlin (2010), `http://www.logos-verlag.de/cgi-bin/engbuchmid?isbn=2673&lng=eng&id=`
3. Moldt, D.: PAOSE: A way to develop distributed software systems based on Petri nets and agents. In: Barjis, J., Ultes-Nitsche, U., Augusto, J.C. (eds.) Proceedings of The Fourth International Workshop on Modelling, Simulation, Verification and Validation of Enterprise Information Systems (MSVVEIS'06), May 23-24, 2006 – Paphos, Cyprus 2006. pp. 1–2 (2006)
4. Schmitz, D., Moldt, D., Cabac, L., Mosteller, D., Haustermann, M.: Utilizing Petri Nets for Teaching in Practical Courses on Collaborative Software Engineering. In: 16th International Conference on Application of Concurrency to System Design, ACSD 2016, Toruń, Poland, June 19-24, 2016. pp. 74–83. IEEE Computer Society (2016). https://doi.org/10.1109/ACSD.2016.21