Exploration of Process Mining Opportunities In Educational Software Engineering - The GitLab Analyser

Philipp Dumbach, M. Sc.[1]
philipp.dumbach@fau.de

Alexander Aly, B. Sc.[1]
alexander.aly@fau.de

Markus Zrenner, M. Sc.[1]
markus.zrenner@fau.de

Prof. Dr. Bjoern M. Eskofier, PhD
[1]Machine Learning and Data Analytics Lab
Department of Computer Science
Friedrich-Alexander-Universität Erlangen-Nürnberg
Carl-Thiersch-Str. 2b, 91052 Erlangen
bjoern.eskofier@fau.de

ABSTRACT

The increasing complexity in software development leads to the necessity for a detailed data analysis. Literature illustrates a stronger research focus on Educational Process Mining (EPM) being applied to the fields of e-learning and professional training. In this work, the opportunities of Process Mining (PM) are further examined by the evaluation of software engineering (SE) courses. The methodology follows the five stages of the *L* life cycle model* for PM projects using data from software repositories. The event log data was analyzed with the PM tool Disco to examine the students' work following an agile development process. The new tool *GitLab Analyser* supports supervisors to visualize educational processes and still extracts event logs for the further analysis and application of PM techniques.

Keywords

data mining, educational process mining, software engineering, agile development, Git, GitLab, education, software repositories, Innovation Lab, Scrum

1. INTRODUCTION

Within the last decades an enormous increase of research interest associated to the field of machine learning systems was observed [7]. Especially during the last ten years, the public interest in the impact of applied machine learning and data analysis methods further grew [6]. The massive increase and demand of new software functionality in these fields also lead to higher software complexities. Dealing with these complexities is difficult especially when it comes to innovations. For the development process of innovative software, time-to-market is a relevant factor due to its impact on revenue and business success of companies in comparison to their potential competitors [9].

In order to cope with the raised complexity Version Control Systems (VCS) were deployed in software development. Additionally, time-boxed plans, IT systems like content management systems as well as issue trackers are now widely used [1, 9].

Those systems also become more and more important for the educational domain. In practical SE courses, students learn and use such systems in order to better structure their development process as well as become prepared for their future employments. Research picked up on this development and started to explore PM opportunities regarding the evaluation of educational software development teams in order to improve the learning process. By extracting event logs from the software development projects, critical processes can be identified and improved.

Philipp Dumbach, Alexander Aly, Markus Zrenner and Bjoern M. Eskofier "Exploration of Process Mining Opportunities In Educational Software Engineering - The GitLab Analyser" In: *Proceedings of The 13th International Conference on Educational Data Mining (EDM 2020)*, Anna N. Rafferty, Jacob Whitehill, Violetta Cavalli-Sforza, and Cristobal Romero (eds.) 2020, pp. 601 - 604

Tools for Educational Process Mining:

Different tools for extracting, visualizing and analyzing event logs for educational purposes were introduced in literature. One solution called *SoftLearn* is mentioned in the work of Vázquez-Barreriros et al. [3] and allows the visualization of the students' learning paths by offering a graphical user interface (GUI).

A publicly available solution is the platform *PHIDIAS* presented by Awatef et al. [2]. This tool provides a service for data and process mining to educational experts. It supports the reconstruction of educational processes and the detailed analysis of social networks.

Sokol et al. [11] introduced a web application called *MetricMiner* for mining software repositories and supporting researchers with the data extraction and statistical inference. Another analysis tool in the application area of Git repositories is *Gitinspector*. This tool is not directly defined as a PM tool, but supports creating insight into development processes by analyzing Git logs and delivering details about the author's contribution over time [5].

Despite all those approaches, Bogarin et al. [4] underline the lack of tools supporting educational specialists from various fields in analyzing educational processes by providing an easy to use tool and a generic framework for EPM in the context of SE courses. Many of the tools demand special knowledge in fields which educational specialists lack.

Applications of Educational Process Mining:

Bogarin et al. [4] summarized various application domains of EPM, which are listed in Table 1.

Table 1: Application areas for EPM [4]

Application field	Amount of studies
Massive Open Online Courses, hypermedia learning environments, learning management systems	8
Computer-supported collaborative learning	5
Professional Training	5
Curriculum Mining	3
Computer-based assessment	2
Software repositories	2

In these applications EPM is used to discover learning flows and sequential patterns. Participants' decision-making processes as well as usage of group communication tools are analyzed to detect learning difficulties. Consequently, the quality of education can be improved by adapting the educational software development process based on the analysis results [4].
Mittal et al. [8] introduced a holistic approach to evaluate the complete educational software development process. They present the idea for a research framework for PM using event logs of VCSs, issue tracking systems and team wikis. To the best of our knowledge, there is no tool available extracting all the relevant event logs necessary to feed this research framework.

Contribution
We contribute a tool called *GitLab Analyser*, which visualizes and extracts EPM relevant event logs from the open source software project management framework GitLab. The tool is easy to use not only for experts but also general educational specialists. Besides, it allows a holistic analysis over underlying learning processes by extracting event logs from the git software repository, the GitLab issue tracker and the GitLab documentation Wiki.
The *GitLab Analyser* is publicly available as standalone application under the following link:
https://www.mad.tf.fau.de/research/gitlab-analyser/.

2. METHODOLOGY:

For the development of the tool we aligned with the first three stages in the five stage process of the *L* life-cycle model* for PM projects as described in the PM manifesto: planning and justification, data inspection, event log extraction, analysis execution and result interpretation [12, 13].

Planning and justification: We planned to extract event logs from a SE course called *Innovation Lab for Wearable and Ubiquitous Computing* offered by the Machine Learning and Data Analytics Lab at the Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU) over the last five semesters. The Innovation Lab is offered to various majors of the university's technical faculty. In this course interdisciplinary student groups of size five to eight develop innovative software related prototypes in cooperation with multiple industry partners and public institutions. Over four months, these teams use the agile development process Scrum [10] and perform three Sprints after an on-boarding phase.

Data inspection: As a project management tool, GitLab Community Version 12.8.0 was used because it offered

- planning features (milestones and issue tracker),
- versioning of the source code and
- documentation features (Wiki).

Both the VCS of the source code as well as the Wiki are git repositories. From all those features events describing the development process can be extracted.

Event log extraction: The events of the VCS and the Wiki were extracted using the git native 'git log' command. Events related to the planning features (milestones, issues) were extracted using GitLab's native API and the REST API client postman. After extracting the events, they were converted to a .CSV file, which can be interpreted by commonly used PM software like Disco or Celonis.
Table 2 summarizes the data set out of the Innovation Lab projects at FAU, the tool was developed with.

Table 2: FAU GitLab log data

GitLab General Information	Value
Number of projects	24
GitLab issues	3409
GitLab repositories commits	5332
GitLab wiki commits	8474
Number of project branches	744

3. RESULTS AND DISCUSSION

3.1 Event Logs for Process Mining

Table 3 gives an overview about the majority of events and activities concerning the planning in GitLab, tracking of source code changes in Git as well as the documentation of the project in Wiki.

Table 3: Events and activities considered in project development process

Planning	issues, issue labels, milestones, branches, merge requests, notes, projects
VCS	number of changed files, commits, commit type, inserted and deleted lines of code, days with commits, number of merges and merge requests
Documentation	inserted and deleted lines of code, number of wiki pages, commits, days with commits

Further indirect available events are extracted by mining the issue notes section e.g. *changed milestone*, *assigned to* or *time spent*. By extracting the data the minimum information about the event logs is collected (*instance id*, *activity*, *timestamp*, *actor*).

3.2 The GitLab Analyser

The *GitLab Analyser* is developed for the implementation as easy to use tool for general educational specialists and to visualize event logs for supervisors. The tool offers three different analysis types all aiming to support supervisors in evaluating students and the development process itself:

1. *Single project analysis:* Support for supervisors in evaluating students and the development process itself.

2. *Group project analysis:* Support for supervisors to compare the performance of different teams of the same course (given that all projects are hosted on the same GitLab server).

3. *Cross-project analysis:* Support for different courses to compare the development process by extracting event logs from different GitLab servers.

Within the tool different result views are presented to the user on a dashboard. A user view offers the analysis of project events performed by the individual users, whereas a project view provides insights into the overall project status.

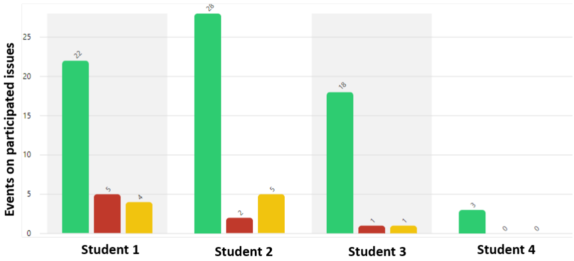

Figure 1: Participation event distribution of users. Color coding: assigned to user (green); unassigned to user but still participated (red) and mentioned user and participated (yellow).

Figure 1 depicts an exemplary graph from the dashboard on the user view. This graph visualizes the number of events as a result of the issues students worked on during the development process and whether they were assigned to those issues. This example shows an even assignment rate with one exception, *Student 4*. Based on this visualization the supervisors can see students with less participated issues and act based on these results by providing additional help to the student in case of lack of background knowledge, breaking big issues down into smaller issues to increase the student's success or motivation.

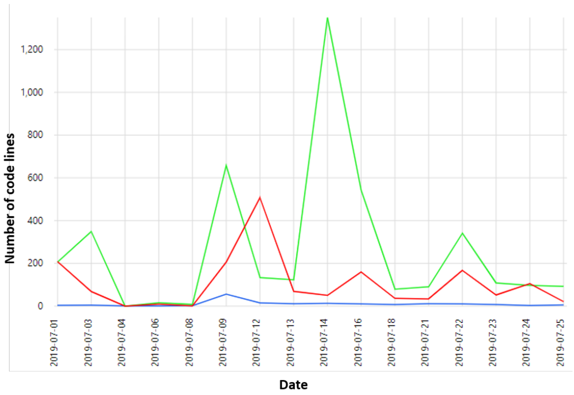

Figure 2: Inserted (green) and deleted (red) code lines in commits and changed files (blue) per day.

Figure 2 shows a graph from the project view. This graph visualizes the number of added, deleted and edited software code lines committed to the VCS of all team members over time. It clearly shows a peak in the middle of the development process, which was due to a Scrum Review at this point in time. By inspecting this graph, supervisors can identify that students do not continuously push their code changes to the repository, which is necessary for other team members to work on a common base. Thus, they can motivate students to improve the development process by continuously committing their new developments to GitLab.

3.3 PM opportunities in university projects

With Celonis and Disco two PM tools were tested. The extracted and transformed event log data (from GitLab raw-log data) was exported and analyzed with Disco to support the identification of correlations in the development process. In the Wiki and Git analysis the commit behaviour and distribution of commit activities was identified. In addition to the visualization of issue states, performance measurements like the average working time on an issue or time until the first user assignment, were determined by the analysis of GitLab features. The participation on issues as well as the information about users carrying out an activity at a specific point in time can be visualized. The time-boxes filter options in Disco enable to use the event logs for precise analysis of activities occurring for example within one Sprint. Furthermore, Disco offered options to analyze specific process parts by filtering the individual and process-relevant activities.

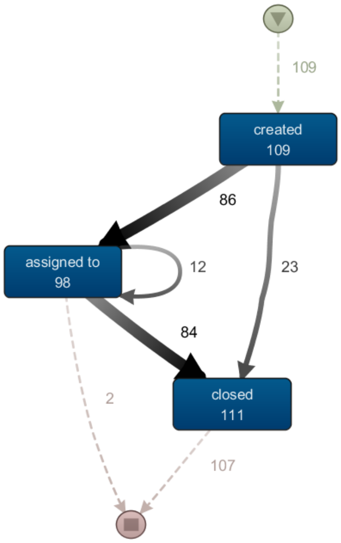

Figure 3: Process Map - Status changes of issues (100 percent activities, 50 percent paths)

Figure 3 illustrates the analysis of assignment activities in form of a process map as an exemplary Disco visualization. The investigation shows the total number of issues being assigned before their closing, i.e. whether a team member was responsible for them. Additionally, the number of assignee changes can be viewed, represented by the self-referencing arrow at the ”assigned to” activity (12 times), and used as indicator for evaluating the team performance.

The event log extraction allowed to gain insight into the students' work following an agile development process. Nevertheless, the analysis and filter configurations in Disco require a certain training period. It was critically questioned whether scientific staff, performing the role as Scrum Master not as a data scientist, need to familiarize themselves with the deeper functionality of PM analysis. Instead, course supervisors should be enabled with a tool to obtain essential analysis results with less effort in a short amount of time.

4. SUMMARY AND OUTLOOK

Due to the increasing complexity in the software development process the application of PM techniques offers valuable opportunities especially in the education domain. Various studies underlined the necessity for tools supporting educational analysis following an agile development process [4]. We introduced a standalone, easy to use tool called *GitLab Analyser* which can be used by supervisors from various fields without significant background in computer science. The tool not only offers the event log extraction for a detailed PM analysis using elaborate PM software (e.g. Disco, Celonis), but also visualizes the individual event logs in clear way for supervisors to evaluate students and the development process quickly. We made the tool publicly available under the following link:
https://www.mad.tf.fau.de/research/gitlab-analyser/.
The *GitLab Analyser* will be used within the upcoming semester of the Innovation Lab by the supervisors of the different development teams for immediate feedback on the development process. Additionally, the first version will be available for supervisors of other universities with similar courses to receive feedback and first bug reports for the next iteration of the tool development process.
Besides, we will use the tool to extract event logs from the last five semesters of the FAU's Innovation Lab and other comparable innovation courses of cooperating universities. By finishing the *L* life-cycle model* for PM projects through performing analysis execution and result interpretation, we will evaluate students' development and learning processes in order to come up with recommendations for improved teaching.

5. ACKNOWLEDGMENTS

The authors would like to thank the Zentrum.Digitalisierung Bayern (ZD.B) for funding the *Innovation Lab for Wearable and Ubiquitous Computing*. We thank Prof. Dr. Gerd Beneken and Martin Kucich from the Rosenheim Technical University of Applied Sciences as well as Prof. Dr. Daniela Nicklas and Simon Steuer from the University of Bamberg for sharing their expertise regarding the conduction of SE courses and supporting the increase of project data for the upcoming steps.
Bjoern M. Eskofier gratefully acknowledges the support of the German Research Foundation (DFG) within the framework of the Heisenberg professorship programme (grant number ES 434/8-1).

6. REFERENCES

[1] H. Awatef, B. GUENI, M. Fhima, A. CAIRNS, and S. David. Process mining in the education domain. *International Journal on Advances in Intelligent Systems*, volume 8 no 1&2:pages 219–232, 2015.

[2] H. Awatef, B. GUENI, M. Fhima, A. CAIRNS, S. David, and N. KHELIFA. Towards custom-designed professional training contents and curriculums through educational process mining: Process mining in the education domain. *IMMM 2014 : The Fourth International Conference on Advances in Information Mining and Management*, 2014.

[3] B. V. Barreiros, M. Lama, M. Mucientes, and J. C. Vidal. Softlearn: A process mining platform for the discovery of learning paths. In *2014 IEEE 14th International Conference on Advanced Learning Technologies*, pages 373–375, 2014.

[4] A. Bogarín, R. Cerezo, and C. Romero. A survey on educational process mining. *WIREs Data Mining and Knowledge Discovery*, 8(1):e1230, 2018.

[5] Gitinspector. https://github.com/ejwa/gitinspector: Accessed on march 9, 2020., 2012.

[6] Iain M. Cockburn, Rebecca Henderson, and Scott Stern. The impact of artificial intelligence on innovation: An exploratory analysis. In Ajay Agrawal, Joshua Gans, and Avi Goldfarb, editors, *The Economics of Artificial Intelligence: An Agenda*, pages 115–146. University of Chicago Press, 2019.

[7] Jiaying Liu, J. Liu, X. Kong, F. Xia, X. Bai, L. Wang, Q. Qing, and I. Lee. Artificial intelligence in the 21st century. volume 6, pages 34403–34421.

[8] M. Mittal and A. Sureka. Process mining software repositories from student projects in an undergraduate software engineering course. In *Companion Proceedings of the 36th International Conference on Software Engineering*, ICSE Companion 2014, pages 344–353, New York, NY, USA, 2014. Association for Computing Machinery.

[9] N. M. Devadiga. Software engineering education: Converging with the startup industry. In *2017 IEEE 30th Conference on Software Engineering Education and Training (CSEE T)*, pages 192–196, 2017.

[10] K. Schwaber. Scrum development process. In J. Sutherland, C. Casanave, J. Miller, P. Patel, and G. Hollowell, editors, *Business Object Design and Implementation*, pages 117–134, London, 1997. Springer London.

[11] F. Sokol, M. Aniche, and M. A. Gerosa. Metricminer: Supporting researchers in mining software repositories. pages 142–146, 2013.

[12] W. van der Aalst, A. Adriansyah, de Medeiros, Ana Karla Alves, M. Westergaard, and M. Wynn. Process mining manifesto. In F. Daniel, K. Barkaoui, and S. Dustdar, editors, *Business Process Management Workshops*, pages 169–194, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[13] M. L. van Eck, X. Lu, S. J. J. Leemans, and W. van der Aalst. Pm^2: A process mining project methodology. In J. Zdravkovic, M. Kirikova, and P. Johannesson, editors, *Advanced Information Systems Engineering*, pages 297–313, Cham, 2015. Springer International Publishing.