



Systematic Mapping Study on Process Mining in Agile Software Development

Sezen Erdem^{1(✉)}, Onur Demirörs^{2,3}, and Fethi Rabhi³

¹ ASELSAN INC, Ankara, Turkey
erdem@aselsan.com.tr

² Computer Engineering Department, Izmir Institute of Technology,
Izmir, Turkey
demirors@metu.edu.tr

³ School of Computer Science and Engineering,
University of New South Wales, Sydney, Australia
f.rabhi@unsw.edu.au

Abstract. Process mining is a process management technique that allows for the analysis of business processes based on the event logs and its aim is to discover, monitor and improve executed processes by extracting knowledge from event logs readily available in information systems. The popularity of agile software development methods has been increasing in the software development field over the last two decades and many software organizations develop software using agile methods. Process mining can provide complementary tools to Agile organizations for process management. Process mining can be used to discover agile processes followed by agile teams to establish the baselines and to determine the fidelity or they can be used to obtain feedback to improve agility. Despite the potential benefit of using process mining for agile software development, there is a lack of research that systematically analyzes the usage of process mining in agile software development. This paper presents a systematic mapping study on usage of process mining in agile software development approaches. The aim is to find out the usage areas of process mining in agile software development, explore commonly used algorithms, data sources, data collection mechanisms, analysis techniques and tools. The study has shown us that process mining is used in Agile software development especially for the purpose of process discovery from task tracking applications. We also observed that source code repositories are main data sources for process mining, a diversity of algorithms are used for analysis of collected data and ProM is the most widely used analysis tool for process mining.

Keywords: Process mining · Agile software development · Process discovery

1 Introduction

Process mining is a relatively young research discipline that sits between computational intelligence and data mining on the one hand, and process modeling and analysis on the other hand [1]. The aim of process mining is to discover, monitor and improve executing processes, by extracting knowledge from event logs readily available in

information systems. Although it has received great attention during the last few years, the idea of reconstructing processes using the stakeholder footprints is not new. Many groups have been working on techniques on process mining over the last two decades. Earlier studies are around application of process mining in the context of workflow management systems. The studies of Cook and Wolf [2, 3] are the pioneers of the works on application of process mining techniques in the field of software development process. There are various methods and algorithms in the field of process mining proposed for different purposes such as rediscovering business processes, conformance checking, process enhancement, software development and social networks analysis [4–10]. Akman and Demirors [11] studied the applicability of process discovery algorithms for software organizations. As the capabilities of information systems and features of CASE tools are improved, it has become possible to record the footprints of each stakeholder in software development processes which leads to increase in the maturity of process mining techniques.

As the benefits of large scale process centric improvement approaches questioned more [12], Agile software development methods have increased their popularity in the software development field. Today prominent percentage of software organizations develop software using agile methods [13]. Agile software projects are generally developed by small teams and in short iterations. Agile methods are more lightweight, more people centric and leave less traces when compared with the traditional approaches such as waterfall [14]. Agile approaches such as Scrum and XP also suggest a set of practices and rules for developers. However, their application in the field is left to the agile teams. Process conformance validation is one of the key challenges in agile software development [15] together with the agile maturity of their practices [16]. Process mining techniques have the potential to be used to discover agile processes followed by agile teams to determine the conformity. They can also establish the necessary evidence for assessing or measuring the agility of organizations.

There are case studies, researches on applicability, systematic literature reviews and mappings on process mining on software engineering in the literature [44–46]. However these studies do not provide evidence and a general understanding on their usability in the Agile software development context.

In order to understand the applicability and usage of process mining in agile software development, we have searched the literature systematically with the aim for determining the state of the art on process mining in agile software development. Our goal is to find out usage areas of process mining in agile software development, explore commonly used methods, algorithms and techniques, data sources, data collection mechanisms, analysis techniques and tools.

Both process mining and Agile Software Development are hot topics. Usage of process mining in agile software development is an interesting topic to research. The aim of this paper is to find out the studies on application of process mining in agile software development process. The contributions of this study will be creating awareness about researches on the subject and highlight the usage areas process mining in agile software development. Also we can create a base for our further research activities and prepare a road map for our studies.

The paper is structured as follows. The design of our research and the method that we followed for the systematic mapping are described in Sect. 2. Findings obtained

from the analysis of selected publications that we performed to answer our research questions are summarized in Sect. 3. Finally, our conclusions are in Sect. 4.

2 Research Method

A systematic literature mapping is a mechanism used to contextualize a particular area of interest through identification, assessment and interpretation of the set of research works which describe such an area [17]. Systematic mapping studies adopt rigorous planning, follow repeatable and well-defined processes, and produce unbiased and evidence-based outcomes [18]. We have conducted a systematic mapping study to achieve our goals for identifying the answers to the research questions we set in advance. The questions are derived based on a preliminary research on process mining in software engineering and also in agile software development. Also the aim of process mining and the fundamentals explained in Process Mining Manifesto [1] lead us to generate these questions.

The guidelines for performing a systematic literature mapping mentioned by Kitchenham [17] are followed in our study quality assessment.

Research questions to address in this mapping are as follows:

- Q1. What are the different categories of research areas concerned with process mining associated with agile software development?
- Q2. What are the different purposes of using process mining in Agile Software Development processes?
- Q3. Which agile teams' footprints are utilized by proposed methods in the agile process mining context?
- Q4. What are the techniques and methods of process mining used for Agile Software Development processes?
- Q5. How do current process mining techniques use these footprints?

Springer Link, IEEE Xplore, ACM Digital Library, Google Scholar and Science Direct repositories are selected as data sources. Search terms are determined, and the same search terms are used for all data sources. The searches are full-text searches and the terms are searched in the title, abstract and body of the paper.

Search Terms:

- “Process mining” AND “Agile”
- “Process mining” AND “Agile Software”
- “Process mining” AND “Agile Software Development”
- “Process mining” AND “Agile Software Lifecycle”
- “Process mining” AND “Scrum”
- “Process mining” AND “Extreme Programming”
- “Mining” AND “Agile event logs”
- “Mining” AND “Scrum event logs”

We performed the study selection in two phases. In the first phase, the search results are evaluated via reading the title and abstract part. The relevance of studies

with process mining in software development, especially agile methodologies, is the main concern for study selection. Relevant studies are selected for final evaluation. In the final evaluation part, entire paper is read and evaluated based on a control list containing study selection criteria. Inclusion/Exclusion criteria for paper selection is as follows:

Inclusion Criteria:

- Article reporting software process mining related study in agile software development context.
- Written in English
- Published in a journal, conference or workshop,
- Full-text is available.

Exclusion Criteria:

- Article that is not utilizing process mining in agile software development,
- Article that is not entirely in English,
- Partially available or unreachable articles
- Article that is published as a short study and not as a full study.

References of the selected papers are also checked to find the related papers to increase the results set size before the selection process is completed.

The list of papers are cross checked by the authors to validate paper selection criteria. Papers which are found related by all authors are selected for data extraction and analysis.

Data about the methods applied in the field of process mining, the tools used, and the results of the applied methods are collected. Collected data is analyzed to generate classifications to extend the analysis.

3 Results

Process mining is a growing research area and there are many works on algorithms, methods, tools and applications in business process management field. There are also special studies on the field of software development which are referred to as software process mining. The application of process mining algorithms and methods in the field of software development is also a rapidly growing research area but overall, there are not many studies on the application of process mining in agile software development.

Although there are more search keywords in the design of search process, the results drive us to similar paper sets. So search keywords are analyzed and only four of them which produce most relevant results are selected for reporting. The numbers of publications associated with the keywords in the target data sources are given in Table 1.

After applying the inclusion and exclusion criteria to the search results, the number of publications related with process mining in the field of agile software development decreases considerably. The different numbers of publications after the study selection process are given in Table 2.

Table 1. Search results

Search Term/Databases	Springer Link	IEEE Xplore	ACM Digital Library	Google Scholar	Science Direct
“Process mining” AND “Agile”	196	2	3	1750	49
“Process mining” AND “Agile software development”	12	14	1	220	3
“Process mining” AND “SCRUM”	17	9	0	209	0
“Process mining” AND “Extreme programming”	17	6	1	151	3

Table 2. Selected search results

Search Term/Databases	Springer Link	IEEE Xplore	ACM Digital Library	Google Scholar	Science Direct
“Process mining” AND “Agile”	6 [19–24]	1 [25]	1 [31]	15 [19, 21, 25, 27, 28, 31–40]	1 [3]
“Process mining” AND “Agile software development”	2 [19, 20]	5 [25–29]	1 [31]	9 [19, 27, 32, 34–39]	0
“Process mining” AND “SCRUM”	1 [19]	3 [25–27]	0	5 [19, 27, 31, 32, 38]	0
“Process mining” AND “Extreme programming”	1 [20]	1 [30]	1 [30]	4 [27, 28, 39, 40]	0

Search results of Google Scholar database include some of the papers from our other search databases. So the number of publications obtained from Google Scholar database is higher when compared with other databases. However, some of the papers appear only in the search results of Google Scholar database.

Selected publications are analyzed to search for answers to our research questions.

Q1. What are the different categories of research areas concerned with process mining associated with agile software development?

Selected publications can be categorized into three subgroups according to the application area of process mining in agile software development.

- Application of process mining in agile business processes: The studies [22–24, 27, 33] in this classification are not directly related with agile software development. Agile business processes are able to act immediately to changes in real time. Application of process mining in agile business processes may lead us to generate methods of application of process mining in agile software development area.
- Usage of process mining in agile software development: These articles focus on developing software using process mining techniques. The aim is to characterize

user interaction with the software, to understand which features are used and to find out sequence of operations. Event logs generated during usage of applications are analyzed via process mining algorithms and the findings are used for developing software in an agile manner [31, 34, 38].

- Usage of process mining for discovering/conformance checking of agile software development processes: Process mining is used for discovering the application of agile software development processes by agile teams/organizations. The aim is not only process discovery but also conformance checking and enhancement are in the scope [19–21, 25, 26, 28–30, 32, 35–37, 39, 40]. In the second research questioned we decomposed the different purposes.

Q2. What are the different purposes of using process mining in Agile Software Development processes?

Classification of publications according to purpose of usage is given in Table 3.

Table 3. Purpose of usage

Purpose of usage	Publications
Process discovery	[19, 21, 25–30, 32, 33, 35, 39]
Process conformance check	[19, 32, 36, 37, 40]
Process improvement/enhancement	[19, 21, 25, 36, 37, 40]

Process discovery is the most commonly used area of process mining [1, 8]. Our work also has shown that process mining is used mostly for process discovery to reveal the processes that were actually executed in organizations. Event logs generated during process executions are analyzed to extract real processes.

Process mining has also usage in process conformance checking. Agile methodologies such as Scrum, Extreme Programming (XP), Agile Unified Process (AUP), have prescriptions to follow. Also the artifacts defined in agile methodologies (product backlog items, tasks, bugs etc.) have states and workflow of these artifacts are pre-defined by agile methodologies. After process discovery, the results are compared with the predefined models to check process conformance.

Another usage area is process improvement and enhancement. Actual executed processes are analyzed to find out bottlenecks and delays in actual executions. Best practices of actual executions are put in evidence for the benefit of organizations.

Q3. Which agile teams' footprints are utilized by proposed methods in the agile process mining context?

The data sources that contains footprints of agile teams can be classified into three groups. The classification is given in Table 4.

Table 4. Process Mining Data Sources

#	Data source	Publications
1	Issue tracking applications such as Microsoft TFS and JIRA are valuable data sources. State changes of work items generate data for analyzing the real process. The order of events, timestamps, and team member changing the state can be queried to generate data for analysis	[19, 32, 39, 40]
2	Software repositories such as version control systems, source code configuration control systems and bug tracking systems. Mining software repositories reveal data about the real process executions	[26, 29, 36, 37, 39]
3	Communication channels between agile team members are another potential source for collecting data about the executed process. Analysis of e-mails between the agile team members is a research topic for extracting process data. The aim is to build a set of workflow models that represent the processes laying behind the agile teams' activities	[21, 25]

Q4. What are the techniques and methods of process mining used for Agile Software Development processes?

Process mining algorithms used in analysis in selected papers are given in Table 5.

Q5. How do current process mining techniques use these footprints?

Our work has shown us that analysis are conducted by using noninvasive techniques to collect the data. Data collected in event logs are queried for transforming into the format that can be fed to process mining tools. Collected data should be transformed into a format that process mining tools can understand to conduct the analysis. Most commonly used format is eXtensible Event Stream (XES) [43]. There exist many tools with process mining capability. ProM is a popular and powerful tool for process mining. It has many plug-ins and serves a high number of alternative to run analysis. Majority of the selected studies make use of ProM or extension of ProM for analysis [19, 21, 22, 25, 29, 32, 35, 38–40]. Disco is another process mining tool used in analysis [19, 39].

Table 5. Techniques and methods

Algorithm	Publications
MINERfull algorithm [41]	[21, 25]
Fuzzy miner [6]	[29, 38, 39]
Genetic mining algorithm [5]	[19]
Heuristic mining algorithm [42]	[19, 22, 38, 40]
Alpha algorithm [8]	[19, 32, 38]

4 Conclusions

There are numerous research studies on process mining algorithms, methods and tools but our study has shown that application of process mining in agile software development processes is a research area that requires more work to be done. Usage of process mining in agile software development have significant potential for agile teams and organizations to increase their success and agility. However the mapping study has shown us that there is not so much work on application of process mining in agile software development context.

We were able to identify 25 papers in this study. Most of these papers are published in the last decade. This shows that software process mining in agile software development has an increasing research trend in recent years.

We observe that process mining is mainly used for discovering actual processes. Organizations want to see what is going in real life and what the bottleneck in their processes are. But this does not mean that process discovery is the sole interest area of process mining. Process conformance checking and process enhancement are other types of process mining having significant usage areas.

We have also observed significant challenges reported for applying process mining in agile contexts. Data collection and event log creation are non-trivial issues in agile software development processes. Due to the nature of agile software development, finding structured event logs to mine the process is a challenging problem. Concept drift is another challenging issue since process improvement is a continuous activity in agile approaches through the iterations.

Agile approaches value individuals and interactions over processes and tools which frequently lead to development processes which are not formalized. Agile approaches are lightweight, more people centric and leave less traces behind when compared with traditional methods. Since development process is often not formalized, agile teams feel themselves freer to determine the sequence of events and the techniques which may result in inconsistency, instability, and unpredictability. Interpretation of agile method rules differently by the teams in an organization may lead to interoperability problems between the projects of the organization. Agile methodologies such as Scrum, Extreme Programming (XP), Agile Unified Process (AUP) have prescriptions to follow in order to achieve real agility. Process mining can be the right tool to extract the actual processes followed by agile teams. It can be valuable and help to visualize consistency, stability, interoperability and repeatability problems.

References

1. van der Aalst, W.: Process mining manifesto. In: Daniel, F., Barkaoui, K., Dustdar, S. (eds.) BPM 2011. LNBIP, vol. 99, pp. 169–194. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-28108-2_19
2. Cook, J.E., Wolf, A.L.: Automating process discovery through event-data analysis. In: Proceedings of 17th International Conference on Software Engineering, pp. 73–82 (1995)
3. Cook, J.E., Wolf, A.L.: Discovering models of software processes from event-based data. *ACM Trans. Softw. Eng. Methodol. (TOSEM)* 7(3), 215–249 (1998)

4. Weijters, A.J., van der Aalst, W.M.P.: Rediscovering workflow models from event-based data using little thumb. *Integr. Comput. -Aided Eng.* **10**(2), 151–162 (2003)
5. de Medeiros, A.K., Weijters, A.J., van der Aalst, W.M.P.: Genetic process mining: an experimental evaluation. *Data Min. Knowl. Disc.* **14**(2), 245–304 (2007)
6. Günther, Christian W., van der Aalst, Wil M.P.: Fuzzy mining – adaptive process simplification based on multi-perspective metrics. In: Alonso, G., Dadam, P., Rosemann, M. (eds.) *BPM 2007*. LNCS, vol. 4714, pp. 328–343. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-75183-0_24
7. Schimm, G.: Mining exact models of concurrent workflows. *Comput. Ind.* **53**(3), 265–281 (2004)
8. van der Aalst, W.M.P.: *Process Mining: Data Science in Action*. Springer, Heidelberg (2016). <https://doi.org/10.1007/978-3-662-49851-4>
9. Hindle, A.: Software process recovery: recovering process from artifacts. In: 2010 17th Working Conference on Reverse Engineering, Beverly, MA, pp. 305–308 (2010)
10. van der Aalst, W.M.P., Song, M.: Discovering Social Networks from Event Logs. BETA Working Paper Series, WP 116, Eindhoven University of Technology, The Netherlands (2004)
11. Akman, B., Demirörs, O.: Applicability of process discovery algorithms for software organizations. In: 35th Euromicro Conference on Software Engineering and Advanced Applications, SEAA 2009, pp. 195–202. IEEE (2009)
12. Uskarı, A., Demirörs, O.: Do staged maturity models result in organization-wide continuous process improvement? Insight from employees. *Comput. Stand. Interfaces* **52**, 25–40 (2017)
13. Garousi, V., Coşkunçay, A., Can, A.B., Demirörs, O.: A survey of software engineering practices in Turkey. *J. Syst. Softw.* **108**, 148–177 (2015)
14. Beck, K., et al.: *Manifesto for Agile Software Development* (2001)
15. Brhel, M., Meth, H., Maedche, A., Werder, K.: Exploring principles of user-centered agile software development. *Inf. Softw. Technol.* **61**(2), 163–181 (2015)
16. Ozcan-Top, O., Demirörs, O.: Assessment of agile maturity models: a multiple case study. In: Woronowicz, T., Rout, T., O'Connor, Rory V., Dorling, A. (eds.) *SPICE 2013*. CCIS, vol. 349, pp. 130–141. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-38833-0_12
17. Kitchenham, B.: *Guidelines for performing systematic literature reviews in software engineering*. EBSE Technical Report EBSE-2007-01. Keele University (2007)
18. Barn, B., Barat, S., Clark, T.: Conducting systematic literature reviews and systematic mapping studies. In: *Proceedings of the 10th Innovations in Software Engineering Conference (ISEC 2017)*, pp. 212–213. ACM, New York (2017). <https://doi.org/10.1145/3021460.3021489>
19. Erdem, S., Demirörs, O.: An exploratory study on usage of process mining in agile software development. In: Mas, A., Mesquida, A., O'Connor, Rory V., Rout, T., Dorling, A. (eds.) *SPICE 2017*. CCIS, vol. 770, pp. 187–196. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-67383-7_14
20. Capodiceci, A., Mainetti, L., Manco, L.: A case study to enable and monitor real IT companies migrating from waterfall to agile. In: Murgante, B. (ed.) *ICCSA 2014*. LNCS, vol. 8583, pp. 119–134. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-09156-3_9
21. Di Ciccio, C., Mecella, M., Scannapieco, M., Zardetto, D., Catarci, T.: MailOfMine – analyzing mail messages for mining artful collaborative processes. In: Aberer, K., Damiani, E., Dillon, T. (eds.) *SIMPDA 2011*. LNBIP, vol. 116, pp. 55–81. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-34044-4_4

22. Montani, S., Leonardi, G., Quaglini, S., Cavallini, A., Micieli, G.: Mining and retrieving medical processes to assess the quality of care. In: Delany, S.J., Ontañón, S. (eds.) ICCBR 2013. LNCS (LNAI), vol. 7969, pp. 233–240. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-39056-2_17
23. Brander, S., et al.: Refining process models through the analysis of informal work practice. In: Rinderle-Ma, S., Toumani, F., Wolf, K. (eds.) BPM 2011. LNCS, vol. 6896, pp. 116–131. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-23059-2_12
24. Schönig, S., Gillitzer, F., Zeising, M., Jablonski, S.: Supporting rule-based process mining by user-guided discovery of resource-aware frequent patterns. In: Toumani, F. (ed.) ICSOC 2014. LNCS, vol. 8954, pp. 108–119. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-22885-3_10
25. Di Ciccio, C., Mecella, M.: Mining artful processes from knowledge workers' emails. *IEEE Internet Comput.* **17**(5), 10–20 (2013)
26. Jankovic, M., Bajec, M.: Comparison of software repositories for their usability in software process reconstruction. In: The International Conference on Research Challenges in Information Science, pp. 298–308 (2015)
27. Caldeira, J., e Abreu, F.B.: Software development process mining: discovery, conformance checking and enhancement. In: 10th International Conference on the Quality of Information and Communications Technology (QUATIC), pp. 254–259. IEEE (2016)
28. Astromskis, S., Janes, A., Mahdiraji, A.R.: Egidio: a non-invasive approach for synthesizing organizational models. In: Glinz, M., Murphy, G.C., Pezzè, M. (eds.) Proceedings of the International Conference on Software Engineering (ICSE), Zürich. IEEE (2012)
29. Poncin, W., Serebrenik, A., van den Brand, M.G.J.: Process mining software repositories. In: CSMR, pp. 5–14. IEEE (2011)
30. Ceravolo, P. et al.: An ontology-based process modelling for XP. In: Software Engineering Conference, Tenth Asia-Pacific Conference, pp. 236–242 (2003)
31. Rubin, V., Lomazova, I., van der Aalst, W.M.P.: Agile development with software process mining. In: Proceedings of the 2014 International Conference on Software and System Process (ICSSP 2014), pp. 70–74. ACM, New York (2014)
32. Zayed, M.A., Farid, A.B.: The discovery of the implemented software engineering process using process mining techniques. *Int. J. Adv. Comput. Sci. Appl.* **1**(7), 279–286 (2016)
33. Schönig, S., Cabanillas, C., Jablonski, S., Mendling, J.: Mining the organisational perspective in agile business processes. In: Gaaloul, K., Schmidt, R., Nurcan, S., Guerreiro, S., Ma, Q. (eds.) CAISE 2015. LNBIP, vol. 214, pp. 37–52. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-19237-6_3
34. Astromskis, S., Janes, A., Mairegger, M.: A process mining approach to measure how users interact with software: an industrial case study. In: Proceedings of the 2015 International Conference on Software and System Process (ICSSP 2015), pp. 137–141. ACM, New York (2015). <http://dx.doi.org/10.1145/2785592.2785612>
35. Sunindyo, W.D., Moser, T., Winkler, D., Biffel, S.: Foundations for event-based process analysis in heterogeneous software engineering environments. In: 2010 36th EUROMICRO Conference on Software Engineering and Advanced Applications, Lille, pp. 313–322 (2010). <https://doi.org/10.1109/seaa.2010.52>
36. Chen, N., Hoi, S.C.H., Xiao, X.: Software process evaluation: A machine learning approach. In: 2011 26th IEEE/ACM International Conference on Automated Software Engineering (ASE 2011), Lawrence, KS, pp. 333–342 (2011). <https://doi.org/10.1109/ase.2011.6100070>
37. Chen, N., Hoi, S.C.H., Xiao, X.: Software process evaluation: a machine learning framework with application to defect management process. *Empir. Softw. Eng.* **19**, 1531 (2014). <https://doi.org/10.1007/s10664-013-9254-z>

38. Olson, K.: Process Mining Concepts for Discovering User Behavioral Patterns in Instrumented Software. All Regis University Theses. 842 (2017). <https://publications.regis.edu/theses/842>
39. Mittal, M., Sureka, A.: 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), pp. 344–353. ACM (2014). <http://dx.doi.org/10.1145/2591062.2591152>
40. Thomson, C., Gheorghe, M.: Using process mining metrics to measure noisy process fidelity. In: Budgen, D., Turner, M., Niazi, M. (eds.) Proceedings of the 13th International Conference on Evaluation and Assessment in Software Engineering (EASE 2009), pp. 132–135. BCS Learning & Development Ltd., Swindon (2009)
41. Di Ciccio, C., Mecella, M.: Mining constraints for artful processes. In: Abramowicz, W., Kriksciuniene, D., Sakalauskas, V. (eds.) BIS 2012. LNBIP, vol. 117, pp. 11–23. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-30359-3_2
42. Weijters, A.J.M.M., van der Aalst, W.M.P., Alves de Medeiros, A.K.: Process mining with the HeuristicsMiner-algorithm. In: BETA Working Paper Series WP 166. Eindhoven University of Technology, Eindhoven (2006)
43. 1849-2016 - IEEE Standard for eXtensible Event Stream (XES) for Achieving Interoperability in Event Logs and Event Streams. <https://standards.ieee.org/findstds/standard/1849-2016.html>
44. Chavada, V.N., Kumar, P.: A survey paper on process mining. *Int. J. Adv. Res. Comput. Sci. Softw. Eng.* **5**(2) (2015). ISSN: 2277 128X
45. Dong, L., Liu, B., Li, Z., Wu, O., Babar, M.A., Xue, B.: A mapping study on mining software process. In: 2017 24th Asia-Pacific Software Engineering Conference (APSEC), Nanjing, pp. 51–60 (2017). <https://doi.org/10.1109/apsec.2017.11>
46. Maita, A.R.C.: A systematic mapping study of process mining. *Enterp. Inf. Syst.* (2017). <https://doi.org/10.1080/17517575.2017.1402371>