

# *Time Performance Evaluation of Agile Software Development*

Asadul Haq Muhammad Shani

Informatics  
Institut Teknologi Sepuluh Nopember  
Surabaya, Indonesia  
Shani15@mhs.if.its.ac.id

Riyanarto Sarno

Informatics  
Institut Teknologi Sepuluh Nopember  
Surabaya, Indonesia  
riyanarto@if.its.ac.id

Kelly Rossa Sungkono

Informatics  
Institut Teknologi Sepuluh Nopember  
Surabaya, Indonesia  
kelly@its.ac.id

Cahyaningtyas Sekar Wahyuni

Informatics  
Institut Teknologi Sepuluh Nopember  
Surabaya, Indonesia  
cahyaningtyas.18051@mhs.its.ac.id

**Abstract**—Agile is a method in software development that demand the development teams to work fast and adapt to changes that happens along the way. However, in the fact, there are activities in software development process that take longer time to complete than others, which is called bottleneck. Bottlenecks are difficult to find if there is no actual event log of the process. To provide the event log this research focus on creating an agile process model which can be executed and generate the event log. Besides the agile process model, this research also discover bottleneck not only from the whole processes but also from processes that are executed frequently. This research creates the agile process model using a workflow management systems. Then the bottleneck from the whole process and processes that are executed frequently are obtained by using Alpha ++ Algorithm and Heuristic Miner Algorithm respectively. The occurrence of activities over time is also analyzed in this research by using dotted chart analysis. This experiment proves that this research can create event log automatically and detect bottlenecks in whole process and processes that are executed frequently.

**Keywords**-- *process mining; agile; bottleneck; alpha ++; heuristic miner algorithm*

## I. INTRODUCTION

A business process is made up of activities that corresponds with one another. During execution of a business process, some activities may take longer than others to be done and cause bottleneck. The activity that creates the bottleneck sometimes cannot be seen clearly, therefore it needs to be extracted from the event log of the business process. The activity of extracting knowledge from an event log is known as Process Mining [1], [2]. By doing Process Mining on a business process the problems of a business process can be identified precisely and therefore better decision can be made for the business process.

Agile is a method used in software development. Agile software development is demanded to move fast and able to adapt to changes that happen during the development process. Agile development work in iterations that last for weeks to months. At the end of every iterations a

functioning product is presented to the stakeholders of the software thus request for changes are likely to exist [3].

This research analyzes the performance of an agile project by creating a process model of agile software development and determine the performance based on the event log of the process model [4].

The result shows that some activities took longer time than others which indicates that changes need to be made regarding these activities.

## II. PRELIMINARY RESEARCH

### A. Agile

Agile development process lets project managers and employees to adapt to circumstances that are always changing and not imposing rigid control [4]. In agile development process, customers are frequently delivered working software for them to test. The delivery of working software is expected even from the early stage of development. Therefore there is a big chance that changes will happen along the development process, changes can even be made at late stages of development therefore customers can be more satisfied. The primary measurement for agile development progress is the working software that are delivered to customers. In agile development process the stakeholders are encouraged to do face-to-face meeting to efficiently and effectively transfer information between themselves [3].

### B. Process mining

Process mining is a technique used to extract information from event logs that nowadays widely available across information systems. Process mining can help business stakeholders with discovery, monitoring, and improvement for their business processes. Process mining is derived by two things. First is that event being recorded are growing into a large amount which means that detailed information about a process can be gathered. The second is in this rapidly changing environment there is a need from business stakeholders to improve their business to stay in competition with others.

Process mining consisted of discovery, conformance checking, social network or organizational mining, automated construction of simulation models, model extension, model repair, case prediction, and history-based recommendations [5]. To get information about performance of a model, the event log of the model is analyzed using process mining [6].

### C. Alpha ++ algorithm

Alpha ++ algorithm is one of process mining algorithms, it is known as the basic algorithm for doing process mining, it is used to generate and analyze activity pattern of a process. Alpha++ algorithm is user-friendly even for the users. Alpha++ algorithm is suitable for mining processes that has “short loops”. However, this has some downsides, some of which are its inability to calculate frequency of activity relation and also its inability to detect noise in the process [7]–[9].

### D. Heuristic miner

Heuristic miner is one of process mining algorithms, this algorithm is stable and already proven to be implemented well in various cases [10], [11]. It mines from the control flow perspective of a process [12]. Heuristic Miner has advantage over alpha ++ algorithm, that is it can calculate frequency of activity relation [13], [14].

### E. Petri Net

Petri net is a process modeling technique with a formal semantics [9]. Petri net is a directed graph consists of two types of nodes, i.e. place and transition. The two nodes are connected by a directed edge called arcs. On each place node, a non-negative integer number of tokens is assigned. On each arc, a non-zero positive integer is assigned as its weight. A place node is represented by circles, the tokens placed inside it are represented by smaller circles, transition nodes are represented by boxes, arcs are represented by arrows [15]. It is commonly used to represent a process that is utilized by a process mining algorithm, it also allows explicit concurrency [14].

### F. Dotted Chart Analysis

Dotted chart is a chart showing spread of events of an event log over time. The basic idea is plotting dots according to time. A dot in the chart represents an event from the event log. The chart consists of two dimensions: Time and Component Type. Measurement of time is shown in the x axis. This way the log can be visually examined and patterns can be identified easier than when trying to identify it with examining text format of the event log [5], [16].

### G. Performance Analysis

Performance analysis is very important for a software, as performance is an important attribute in measuring the quality of a software. Bad performance can result in broken customer relation, decrease in productivity of user, and loss of revenue [17].

## III. RESEARCH METHOD

In this research a process model of agile software development was made, the model was made with a workflow management system. The process in the model is divided into six parts: Requirements, Design and Architecture, Development and Coding, Quality Assurance

and Software Testing, Implementation, and Maintenance and Support.

The process model was run several times with each iteration done through different path, the event log of this process was recorded in database and later extracted.

After the mining is complete the next step is the analysis. In this research Alpha ++ algorithm and Heuristic Miner Algorithm is used to mine the event log. This research used Performance Analysis with Petri net for the analysis. Performance Analysis with Petri net divide the activities in the event logs into colors that corresponds with three categories: high, medium, and low based on the average time needed to complete the activity.

This research do evaluation of performance based on time. Process Mining allows us to get insight from the event logs. The output of Process Mining can be used for improving the performance of a development process [5].

One of the research in this field is done in [18]. In that research the researchers use the event log from a European touristic software system development of a company, the development using Agile software development. The system used to book planes, hotels, trains, etc. From the first iteration they already found out that the users of the system did not use the system as it is supposed to be used and the system had a performance problem. From the research they discovered a model of user behavior and software runtime behavior. The research has successfully integrated process mining with software development using Agile.

## IV. RESULT AND ANALYSIS

### A. Used Data

Data used in this research is event log of a process model of an agile software development. The process model was made using a workflow management system called ProcessMaker. The ProcessMaker used in this research, ProceMaker Community version, is a free version thus it cannot perform process mining. To overcome this another software is used, it is a process mining tool named ProM 5.2. The model consisted of thirty three activities and was run thirty three times for generating event log. The event log which was in csv format consisted of 4 columns: Case Id, Activity Name, Timestamp, and Resource which means the actor that did the activity. Snippet of the event log can be seen in TABLE I. The running of the process model was done manually and the timestamp recorded is the actual timestamp when it was run, so it doesn't really represent the actual condition in real world. Because of this we made a python script to randomly generate time and then assign it to the timestamp column of the event log.

### B. Data Processing

The event log extracted from database is in csv format, where the tool used for analysis, ProM 5.2, is compatible with various file types but csv is not one of them. One of the file types compatible with prom 5.2 is mxml file. Therefore the csv formatted event log file was converted to mxml file using Fluxicon Disco, an application for process mining. Here we choose ProM 5.2 over Fluxicon Disco because Prom 5.2 has more features to it. Then the event log is imported to ProM 5.2 to be analyzed using Alpha++ algorithm and also heuristic miner.

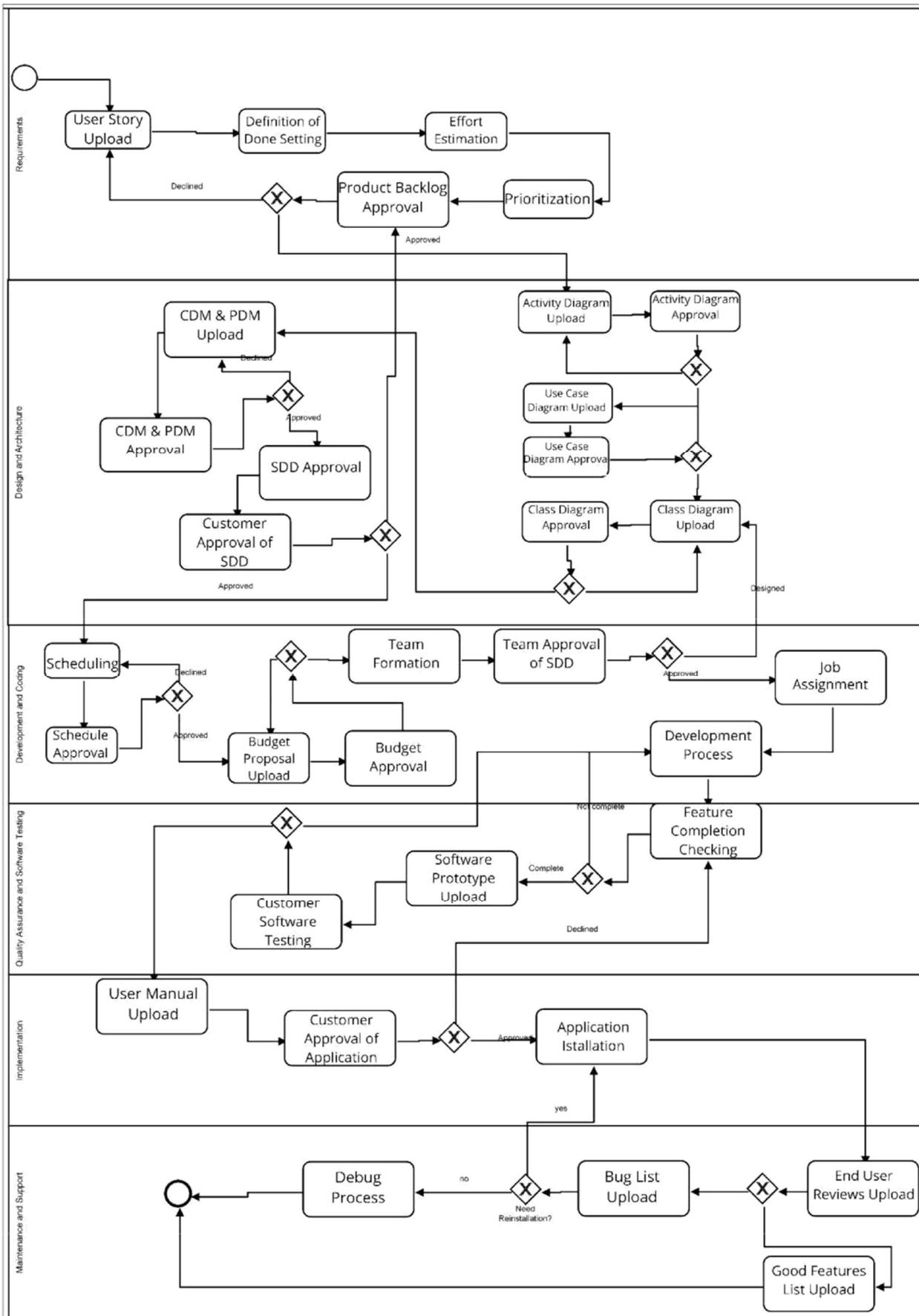


Fig. 1. Agile SDLC model made with ProcessMaker Community Edition

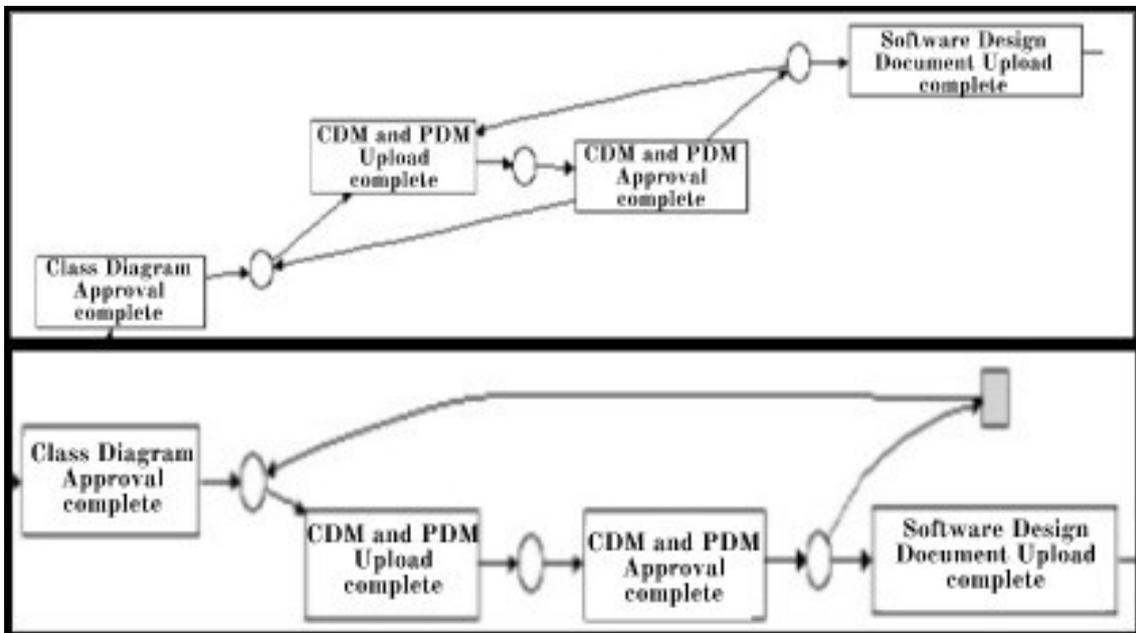


Fig. 2. Result of Alpha ++ and Heuristic Miner analysis

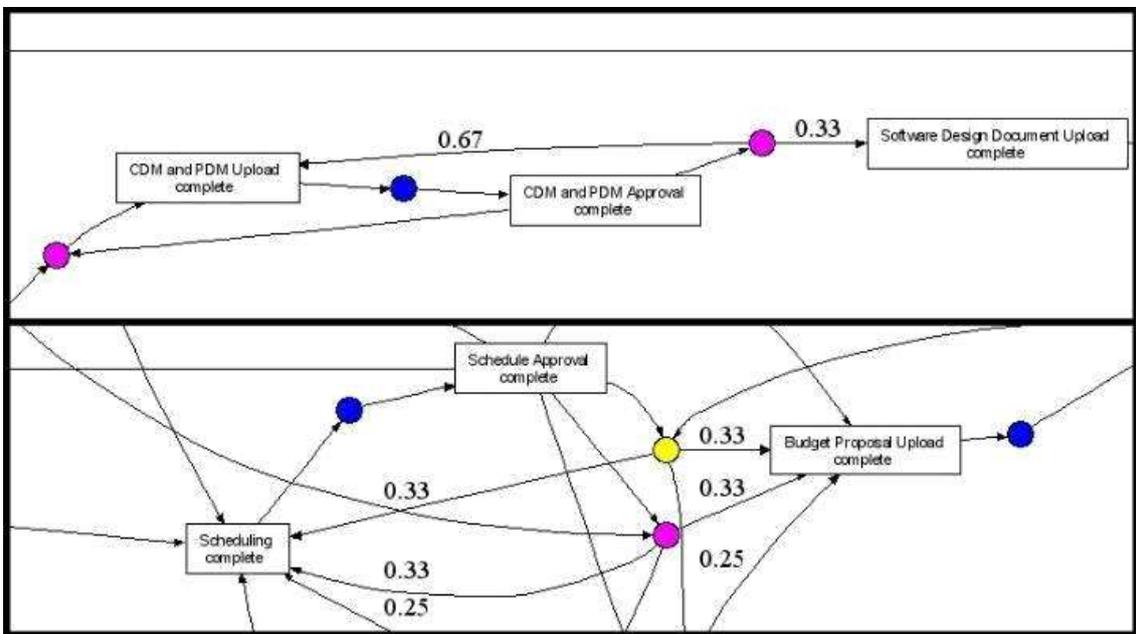


Fig. 3 Result of performance analysis with petri net on Alpha ++ Algorithm

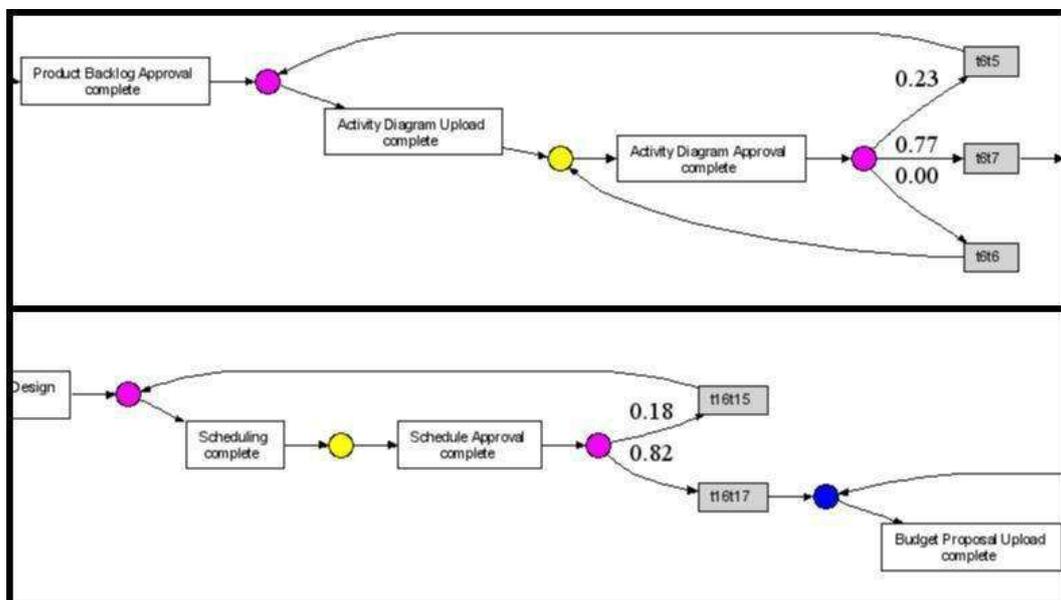


Fig. 4 Result of Performance analysis with petri net on Heuristic Miner

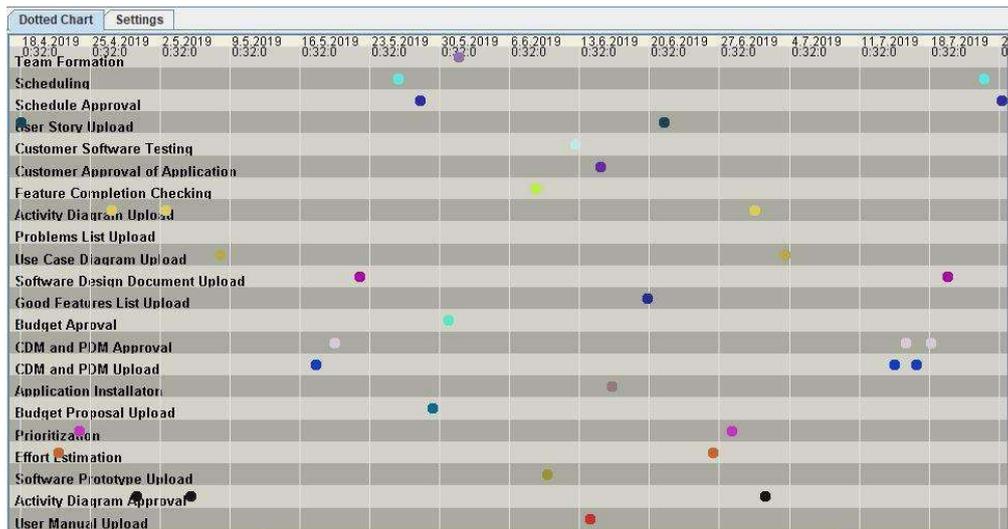


Fig. 5 Result of Dotted Chart Analysis

### C. Analysis

Fig. 1 shows the process model made with ProcessMaker. From the event log analysis can be found which activity is causing bottleneck. After event log is extracted from the process then mining is done using ProM 5.2. Fig. 2 shows snippet of the result of mining with Alpha ++ Algorithm and Heuristic Miner. Then performance analysis with petri net is done, the result is a diagram of activities from the event log with labels regarding time spent by each activity. The threshold for each time class is automatically determined by ProM 5.2. Result of performance analysis with petri net on each mining technique can be seen in Fig. 3 and Fig. 4, it only shows the snippet due to the large size of the actual image. ProM 5.2 also shows an information table of the event log including average, minimum, and maximum time spent per activity

In this research, dotted chart analysis is also done to show the spread of activity over time. This way it can be seen at what time most activity is done, and what activity is done the most on a particular time. Result of dotted chart analysis can be seen in Fig. 5. From Dotted Chart Analysis it can be discovered, if exist, a particular time where activities are slower to get done.

From Fig. 3 it can be seen that the transition between “CDM and PDM Upload” and “CDM and PDM Approval” has blue as the color which indicate that the activity “CDM and PDM Upload” is categorized as low in term of time of completion, and it can be concluded that this activity is not a problem to the process. On the other hand the transition from “CDM and PDM Approval” to “Software Design Document Upload” has purple as the color which indicates that there might be a problem in “CDM and PDM approval” that cause the transition between the two activities took a long time. Because of the different thresholds, results from Alpha ++ Algorithm and heuristic miner is also different for some activities. These visualizations can really help the stakeholders of the process to identify problems in the process and solve them.

### V. CONCLUSION

Agile software development is widely used in the software industry. With the growing number of agile implementation, there need to be a way of evaluating the performance to keep the development in track. In this research, a model of agile development process was made using a workflow management system called ProcessMaker. The event log then converted from csv format to mxl format using Fluxicon Disco. The mxl file then opened with ProM 5.2 and mining process was done to it using Alpha ++ Algorithm and Heuristic Miner algorithm. After that performance analysis with petri net was done to get visualization of time needed to complete each activity in the event log.

The result of performance analysis with petri net shows a diagram of the whole processes with details of time spent per activity and which activity causes bottleneck in the process. On the diagram from Alpha ++ algorithm, it can be seen that ten out of thirty three activities are categorized as bottleneck with eight of them happened in early stage of development. This indicates that in the process model made for this research there might be problem particularly in the early stages of development. On the other hand, the diagram resulted from performance analysis using Petrinet on event log that was mined with Heuristic Miner shows different result. This difference occurs because the thresholds are different and the petri net is also different.. The threshold for each time class were automatically defined by ProM 5.2. The petri net from heuristic diagram can be different from the actual event log, to check the alignment between those conformance checking needs to be done. The result of conformance checking on the petri net from heuristic miner shows that the Fitness Score is 0.98. Fitness Score shows how many of the events in the event log actually captured in the petri net (model) [12]. Alpha ++ Algorithm Shows the whole process while Heuristic Miner algorithm only processes that done many time depending on the threshold. Here in this research the threshold is manually set by Prom 5.2. The findings of the bottlenecks can be used by business stakeholders as a basis for making decision and further improve their software development. Dotted chart analysis was also done to check the spread of the activities over time of running of the process Further research of this

topic can be done in area of real time process mining. With event log streamed to the system, activities that cause bottlenecks can be immediately identified. This will speed up the whole development process and products can be delivered to customers more quickly.

TABLE I. Event Logs

Caseid	NameActivity	Timestamp	Resource
104313515 5c9f584318 f5d1077721 420	User Story Upload	18-04-19 00:32	Customer
104313515 5c9f584318 f5d1077721 420	Product Backlog Approval	25-04-19 07:41	Project
.	.	.	.
.	.	.	.
.	.	.	.
906415055 5c9faf1a42 040504231 5506	End User Reviews Upload	20-04-24 19:04	Cusomer

## REFERENCES

- [1] K. R. Sungkono and R. Sarno, "CHMM for discovering intentional process model from event logs by considering sequence of activities," in *2017 4th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI)*, 2017, pp. 1–6. <http://doi.org/10.1109/EECSI.2017.8239194>.
- [2] R. Sarno and K. R. Sungkono, "Coupled Hidden Markov Model for Process Discovery of Non-Free Choice and Invisible Prime Tasks," *Procedia Computer Science*, vol. 124, pp. 134–141, 2018. <http://doi.org/10.1016/j.procs.2017.12.139>.
- [3] K. Beck, M. Beedle, A. van Bennekum, A. Cockburn, W. Cunningham, M. Fowler, J. Grenning, J. Highsmith, A. Hunt, et al., *Manifesto for agile software development*. 2011.
- [4] S. Augustine, "Agile project management: steering from the edges," *Communications of the ACM*, vol. 48, no. 12, 2005.
- [5] K. R. Sungkono and R. Sarno, "Constructing Control-Flow Patterns Containing Invisible Task and Non-Free Choice Based on Declarative Model," *International Journal of Innovative Computing, Information and Control (IJICIC)*, vol. 14, no. 4, 2018.
- [6] S. J. J. Leemans, D. Fahland, and W. M. P. van der Aalst, "Discovering Block-Structured Process Models from Incomplete Event Logs," *International Conference on Applications and Theory of Petri Nets and Concurrency*, vol. 9698, pp. 91–110, 2014. [http://doi.org/10.1007/978-3-319-07734-5\\_6](http://doi.org/10.1007/978-3-319-07734-5_6).
- [7] L. Hakim, R. Sarno, and K. R. Sungkono, "Modified Alpha++ Algorithm for Discovering the Hybrid of Non-Free Choice and Invisible Task of Business Processes," *International Journal of Intelligent Engineering and Systems (IJIES)*, vol. 12, no. 3, pp. 31–40, 2019.
- [8] R. Sarno and K. R. Sungkono, "Coupled Hidden Markov Model for Process Mining of Invisible Prime Tasks," *International Review on Computers and Software (IRECOS)*, vol. 11, no. 6, pp. 539–547, 2016. <http://doi.org/10.15866/irecos.v11i6.9555>.
- [9] A. K. A. De Medeiros, B. F. Van Dongen, W. M. P. van der Aalst, and A. J. M. M. Weijters, "Process mining: Extending the  $\alpha$ -algorithm to mine short loops," *Eindhoven University of Technology Eindhoven*, pp. 1–25, 2004. <http://doi.org/10.1016/j.is.2011.01.003>.
- [10] A. Burattin, A. Sperduti, and W. M. van der Aalst, "Heuristics miners for streaming event data," arXiv, 2012.
- [11] W. Song, X. Xia, H.-A. Jacobsen, and H. Hu, "Heuristic Recovery of Missing Events in Process Logs," 2015. <http://doi.org/https://doi.org/10.1109/ICWS.2015.24>.
- [12] S. K. L. M. vanden Broucke and J. De Weerd, "Fodina: A robust and flexible heuristic process discovery technique," *Decision Support Systems*, vol. 100, pp. 109–118, 2017. <http://doi.org/10.1016/j.dss.2017.04.005>.
- [13] D. Rahmawati, M. A. Yaqin, and R. Sarno, "Fraud detection on event logs of goods and services procurement business process using Heuristics Miner algorithm," in *2016 International Conference on Information Communication Technology and Systems (ICTS)*, 2016, pp. 249–254. <http://doi.org/10.1109/ICTS.2016.7910307>.
- [14] Y. Caesarita, R. Sarno, and K. R. Sungkono, "Identifying bottlenecks and fraud of business process using alpha ++ and heuristic miner algorithms (Case study: CV. Wicaksana Artha)," *Proceedings of the 11th International Conference on Information and Communication Technology and System, ICTS 2017*, vol. 2018-Janua, pp. 143–148, 2018. <http://doi.org/10.1109/ICTS.2017.8265660>.
- [15] L. Wells, "Performance analysis using CPN tools," *Proceedings of the 1st international conference on Performance evaluation methodolgies and tools - valuetools '06*, p. 59, 2006. <http://doi.org/10.1145/1190095.1190171>.
- [16] R. Sarno and K. R. Sungkono, "Hidden Markov Model for Process Mining of Parallel Business Processes," *International Review on Computers and Software (IRECOS)*, vol. 11, no. 4, pp. 290–300, 2016. <http://doi.org/10.15866/irecos.v11i4.8700>.
- [17] C. Klinkmüller and I. Weber, "Analyzing control flow information to improve the effectiveness of process model matching techniques," *Decision Support Systems*, vol. 100, pp. 6–14, 2017. <http://doi.org/10.1016/j.dss.2017.06.002>.
- [18] V. Rubin, I. Lomazova, and W. M. P. van der Aalst, "Agile development with software process mining," in *Proceedings of the 2014 international conference on software and system process*, 2014.