Process Mining for Project Management

Jeni Joe
Ex-Student
Computer Engineering Department
St. Francis Institute of Technology
Mumbai, India
jeni_1993@rediffmail.com

Yasha Ballal
Ex-Student
Computer Engineering Department
St. Francis Institute of Technology
Mumbai, India
yashaballal@gmail.com

Tanya Emmatty
Ex-Student
Computer Engineering Department
St. Francis Institute of Technology
Mumbai, India
et.tanya@gmail.com

Snehal Kulkarni
Assistant Professor
Computer Engineering Department
St. Francis Institute of Technology
Mumbai, India
snehalkulkarni@sfitengg.org

***Abstract*—Business process mining or process mining is the intersection between data mining and business process modelling that extracts business patterns from event logs. Event logs are freely available in any organization. Business logs are a potential source of useful information. By the various patterns that are present in the logs, a lot can be estimated about the type of procedures that should be incorporated into the organization for better performance. Event logs store information about time and event data of business processes. Process mining algorithms are used to mine business process models using event logs. Generating automated business models out of this could provide valuable insight to a firm eventually leading to customer satisfaction. Process Mining works by three phases: discovery, conformation and alteration. By using process mining, many kinds of information can be collected about the process, such as control-flow, performance, organizational information and decision patterns. A process model could be represented as Petri nets which is a formal graphical representation of the workflow diagram or it can be represented as Business Process Modelling Notation. This project aims to develop a user friendly platform which is capable of generating petri net like models by process mining. By using various process mining algorithms we will develop software which would mine the event logs of a particular firm. It would provide a data or workflow analysis scheme. This would optimize business process intelligence and thus provide alternative and superior work strategies. In this project, we are mainly targeting project management using process mining. There are many projects that are undertaken by an IT company that all follow the same procedure. The concept of business process mining can be used in order to improve the performance of a company by optimizing its Software Development Life Cycle. By feeding the previous logs of a similar project of the company, the software would give a flowgraph. This flowgraph can help to identify the sequence of the activities, roles in the organization as well as various efficiency parameters. The algorithm being used is the Heuristic Miner Algorithm for process mining.**

Keywords-process mining; event logs

I. INTRODUCTION

Process Mining is done through rigorous data collection and then application of appropriate mining techniques depending on the nature of the data. The relationship of Project Management with Process Mining is very close since for the various stages of project management, it is necessary to follow a workflow that is optimal, in order to generate best possible results. However, customized process mining tools for Project Management are unavailable. The project aims at tackling this problem and creating a tool that would be specially used for management of projects. This tool can be used by IT organizations in order to find out the optimal techniques for mining the project procedure related information.

A. Motivation

Many tools (e.g., EMiT, Little Thumb, MiSoN) have been developed for serving the purpose of Business Activity Monitoring and Business Process Intelligence These tools work by extracting knowledge from event logs for process mining. The formats used by different tools for reading and storing log files are different and these tools also present their results in different ways. This, unfortunately, makes it difficult to use different tools on one set of data and further to compare the mining results. Some of these tools also put into practice concepts that can be very useful in the other available tools, however it is, in most cases difficult to combine tools efficiently. We aim to build a framework that will enable entry of logs directly into the software. The logs will be converted into a lucid, user-friendly Process Model (flow graphs) that caters specifically to project management.

B. Problem formulation

Project management is both a process and an activity consisting of planning, motivating, organizing, and controlling protocols, procedures and resources to achieve specific goals in both scientific and daily problems. A project can be defined as a temporary venture designed to produce a unique result with a well-defined beginning and end (usually constrained by funding, time and deliverables), undertaken to meet specific goals and objectives, typically to bring about added value or useful change. Project management faces two constraints: the primary or basic challenge is to meet the project goals and cater to its objectives while also satisfying primary constraints like scope, funding and deliverables within the stipulated time frame. The secondary challenge is intelligent allocation of inputs so as to achieve optimum results. What we advance is a model, tailored to the characteristics of projects, which identifies and relates common project success factors and measures using process mining.

II. RELATED WORK

Business processes are an integral part of various fields. It has been an increasing concern about monitoring and controlling them in order to verify if they are functioning properly. Business processes are getting more complicated as well as the numbers of processes are increasing. Hence manually modeling these processes is not possible and therefore, there is a need to automate this process .There is also a need to mine the previous process information and generate patterns so as to give valuable insight for future projects. “Business process mining, or just process mining, is a field which is the intersection of data mining and business process modeling” that extracts useful information based on execution logs. Since many systems record their transactions in log files, process mining techniques can be used for a large class of business processes. By using process mining, many kinds of information can be collected about the process, such as control-flow, performance, organizational information, role discovery and decision patterns [1].

The idea of process mining is to discover, monitor and improve real processes by extracting knowledge from event logs readily available in today's systems [2]. Process discovery is extracting process models from the event logs and conformance checking is comparing the previous model with the new set of event logs. Starting point for process mining is an event log. All process mining techniques require sequential and regular updation of event logs which contains information of the respective event. All event logs need to store two items of mandatory information: timestamp and the event name. It can also store other additional information like an individual’s role in the activity, activity’s start date, activity’s end date and other efficiency parameters [3].

The basic guiding principles for process mining are:

1. Event data should be treated as first-class citizens
2. Log extraction should be driven by questions
3. Concurrency, choice and other basic control-flow constructs should be supported
4. Events should be related to model elements
5. Models should be treated as purposeful abstractions of reality
6. Process mining should be a continuous process [2].

A. Event logs

To be able to apply process mining techniques it is essential to extract event logs from data sources like databases, transaction logs or audit trails. Apache POI is a useful software for this extraction process. Once the relevant data has been located, the extraction and conversion is fairly straightforward. The challenge is to select event data related to the questions an organization has [4].

III. PROPOSED SOLUTION

We intend to use the previous event logs of a similar project conducted by the company so as to develop a workflow scheme and analysis which could optimize the current projects efficiency in terms of time and resources. The event logs need to be stored on Microsoft Excel sheets with mandatory fields of timestamp and activity name as well as additional fields like role and other work parameters. These Excel sheets are imported into the software by using Apache POI (Poor Obfuscation Implementation). The entire application will be coded in Java. There are various mining algorithms for process mining. We will be implementing the Heuristic Mining Algorithm. This algorithm is tolerant to noise and can be applied regardless of whether the input set of execution traces (logs) are complete, because it can handle incomplete logs. The algorithm will also be coded in Java. As displayed in Fig. 1, once the logs are fed into the Mining algorithm, it generates a customized process diagram for business project management. We will be displaying the output as a simple, user friendly Process Model. Thus we can display the optimal sequence of activities which should be followed as well as determine the time consumption for each activity. User will be given an option to edit the logs. The user can view how different activities are dependent on each other as well as the frequency of occurrence of different paths. The user will also be given an option to view case wise statistics as well as the overall statistics for the log. Thus user can analyze the workflow parameters at a lower level of detail as well.

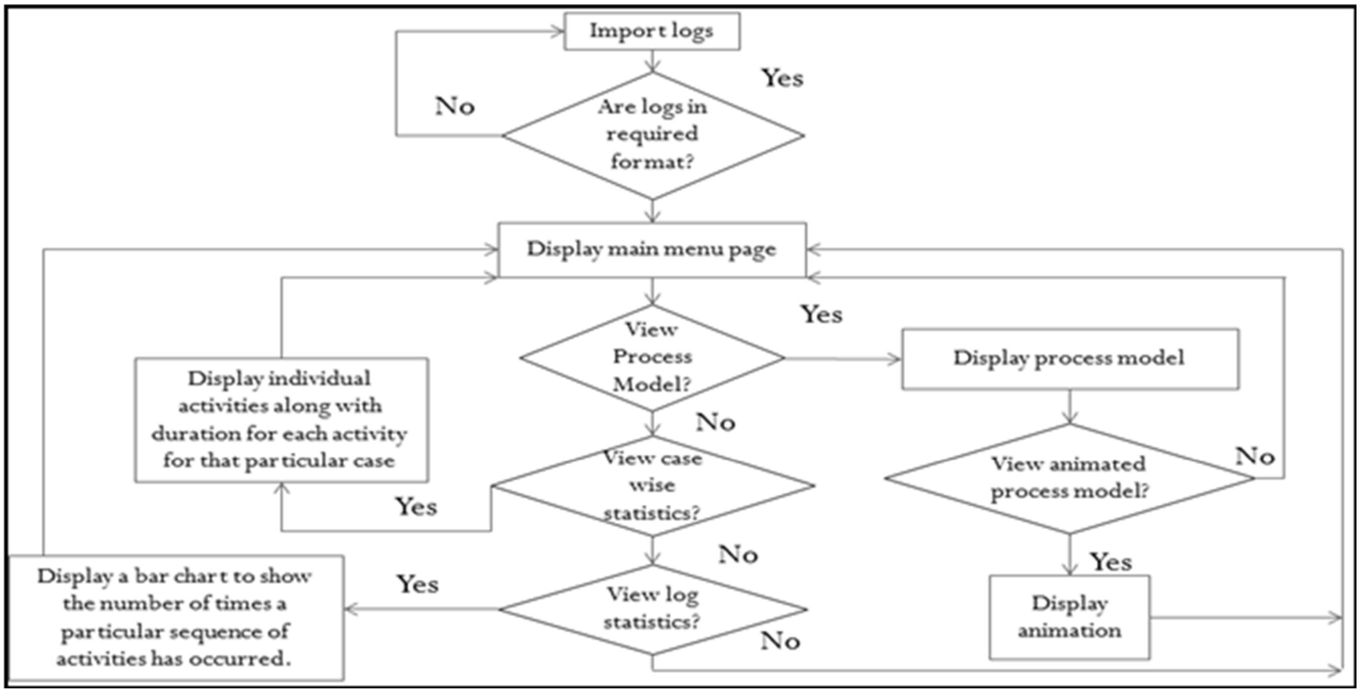

Import logs
Yes
No
Are logs in required format?
Display main menu page
Yes
View Process Model?
Display process model
Display individual activities along with duration for each activity for that particular case
No
View case wise statistics?
Yes
View animated process model?
No
Display a bar chart to show the number of times a particular sequence of activities has occurred.
Yes
View log statistics?
No
No
Yes
Display animation

Figure1. Flow of Proposed work

These diagrams can be scaled to desired complexity using a friendlier user interface.

The goal of process mining is to extract information about processes from transaction logs. It assumes that it is possible to record events such that

- Each event refers to an activity (i.e., a well defined step in the process),
- Each event refers to a case (i.e., a process instance),
- Each event can have a performer also referred to as originator (the actor executing or initiating the activity), and
- Events will have a timestamp and will be completely and sequentially ordered [5].

IV. Implementation

A. Algorithm

The Heuristics Miner Algorithm will be used to perform the mining of the logs. Heuristic miner algorithm takes this information and it creates a flow graph for the events within individual cases. There is no dependency in between the two cases. Each sequence of activities is known as an event trace. Let T be a set of activities, then σ∈ T^* is an event trace. W is an event log, i.e. a multi set of event traces. The motto of Heuristic Miner Algorithm is the construction of a matrix having all the dependency values of the various activities and eliminating the less dependent values on the basis of three thresholds that are discussed in the section.

Consider two activities 'c' and 'd' then we can set the following dependency rules for these two activities:

1. $c >_W d$ iff there is a trace, trace σ = $t_1 t_2 t_{3\dots} t_n$and i ∈ $\{1, \dots, n-1\}$ such that σ ∈ W and $t_i = c$ and $t_{i+1} = d$,
2. $c \rightarrow_W d$ iff $c >_W d$ and $d \ngtr_W c$,
3. $c \neq_W d$ iff $c \ngtr_W d$ and $d \ngtr_W c$,
4. $c \parallel_W d$ iff, $c >_W d$ and $d >_W c$,
5. $c >>_W d$ iff there is a trace σ=$t_1 t_2 t_{3\dots} t_n$and i ∈ $\{1, \dots, n-2\}$ such that σ ∈ W and $t_i = c$ and $t_{i+1} = d$ and $t_{i+2} = c$,
6. $c >>>_W d$ iff there is a trace σ=$t_1 t_2 t_{3\dots} t_n$ and i < j and i,j∈ $\{1, \dots, n-1\}$ such that σ ∈ W and $t_i = c$ and $t_j = d$.

Based on these rules, the value of dependency of each activity with each other activity is found out. Mining of dependency graph between two activities 'c' and 'd' is given in (1).

$$c \Rightarrow_W d = \left(\frac{|c>_W d|-|d>_W c|}{|c>_W d|+|d>_W c|+1}\right) \quad (1)$$

In cases where the same activity is executed multiple times, this refers to a loop in the corresponding model. For loops consider that $c >>_W c$ is the number of times $c >>_W d$ occurs in W. These are represented by (2) and (3) respectively.

$$c \Longrightarrow_W c = \left(\frac{|c >_W c|}{|c >_W c| + 1}\right) \qquad (2)$$

$$c \Longrightarrow_{2W} d = \left(\frac{|c >>_W d| + |d >>_W c|}{|c >>_W d| + |d >>_W c| + 1}\right) \qquad (3)$$

In order to understand the dependency for AND/XOR-split/join and non-observable tasks, consider the following example:

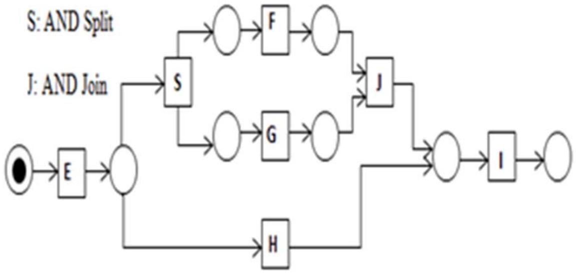


Figure2.Flowgraph for example [6]

As observed in Fig. 2, F and G are activities that can be executed in parallel. Hence, an AND split and AND join has been added to support the same. The above diagram which is a Process model has been developed from a casual matrix which looks like given in Table 1

TABLE I . EXAMPLE [6]

Activity	Input	Output
E	Φ	(F ∨ H) ∧ (G ∨ H)
F	E	I
G	E	I
I	(F ∨ H) ∧ (G ∨ H)	Φ
H	E	I

Each activity will have both an input expression and an output expression. Because E is the start activity the input expression is empty and activity E is enabled. After the firing of E the output (F ∨ H) ∧ (G ∨ H) is activated (i.e. (F ∨ H) is activated and (G ∨ H) is activated). We now look if activity F is enabled. Because the input expression of F is only E, we have to look if all F's are active in the output expression of E. This is the case. But because the ∨ in (F ∨ H) is an exclusive or, the (F _ H) isn't longer activated but the (G ∨ H) expression still is. We now look if activity H is enabled. Because the input expression of H is only E, we have to look if all H's are active in the output expression of E. This is the not case because (F ∨ H) is no more activated. However, activity F is still enabled (all relevant expressions are still in activated state). This is exactly the kind of behavior we would like to model.

Equation (4) is defined to express the above formulated idea. Let W be an event log over T, a, b, c ∈ T, and b and c are in depending relation with a [6]. Then,

$$a \Longrightarrow_W b \wedge c = \left(\frac{|b >_W c| + |c >_W b|}{|a >_W b| + |a >_W c| + 1}\right) (4)\ [6]$$

V. EXPERIMENTAL SETUP AND RESULT

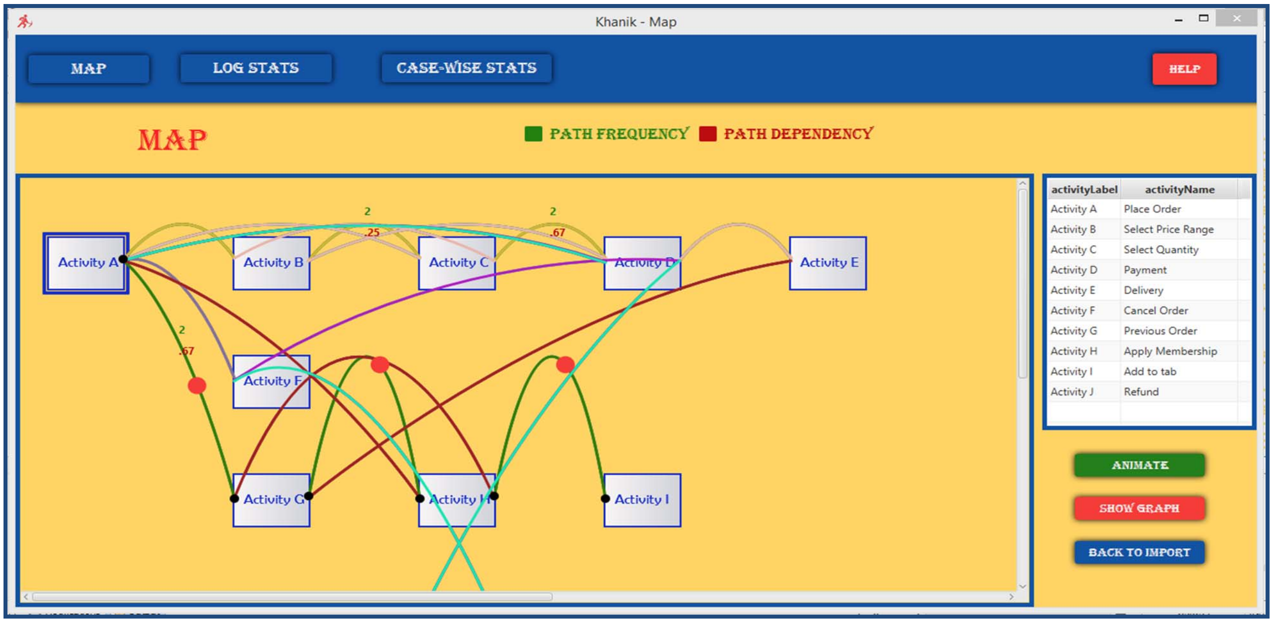
The final result comprises of 3 main pages:

- The animated process model.
- Case-wise statistics
- Log statistics

Fig. 3 displays the process model. The “Show Graph” button generates the process model on the page. The model is displayed using the JavaFX Canvas API. “Animate” button animates the model using tokens which displays the individual sequence of activities in the event log imported. The path frequency and path dependency for each connecting path is also displayed. If a particular path has dependency as 0, it implies that both these activities can be performed simultaneously. Each path is highlighted in a unique color so as to differentiate the different sequences. A table shows the activity label used in the model with its actual name.

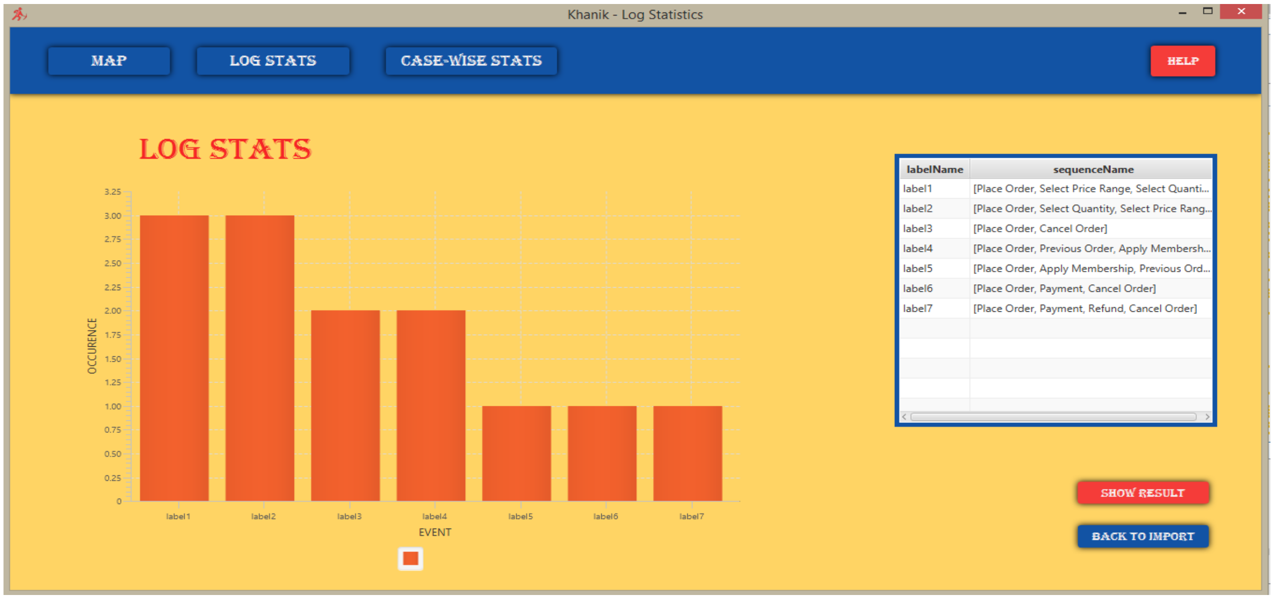
The animated process model thus helps the user to see the ‘big picture’ of what exactly his/her event logs signify. Thus we can predict the sequence of activities. Khanik also computes and displays the event log statistics. That is, for each case present in the event logs, the software computes the frequency of occurrence. On clicking the Show Result button, these values are plotted and presented to the user in the form of an attractive and elucidative bar graph.

The Log Stats help the user in identifying the cases which have the highest/ lowest frequency of occurrence. This is displayed in Fig. 4



activityLabel	activityName
Activity A	Place Order
Activity B	Select Price Range
Activity C	Select Quantity
Activity D	Payment
Activity E	Delivery
Activity F	Cancel Order
Activity G	Previous Order
Activity H	Apply Membership
Activity I	Add to tab
Activity J	Refund

Figure3. Map display



labelName	sequenceName
label1	[Place Order, Select Price Range, Select Quanti...
label2	[Place Order, Select Quantity, Select Price Rang...
label3	[Place Order, Cancel Order]
label4	[Place Order, Previous Order, Apply Membersh...
label5	[Place Order, Apply Membership, Previous Ord...
label6	[Place Order, Payment, Cancel Order]
label7	[Place Order, Payment, Refund, Cancel Order]

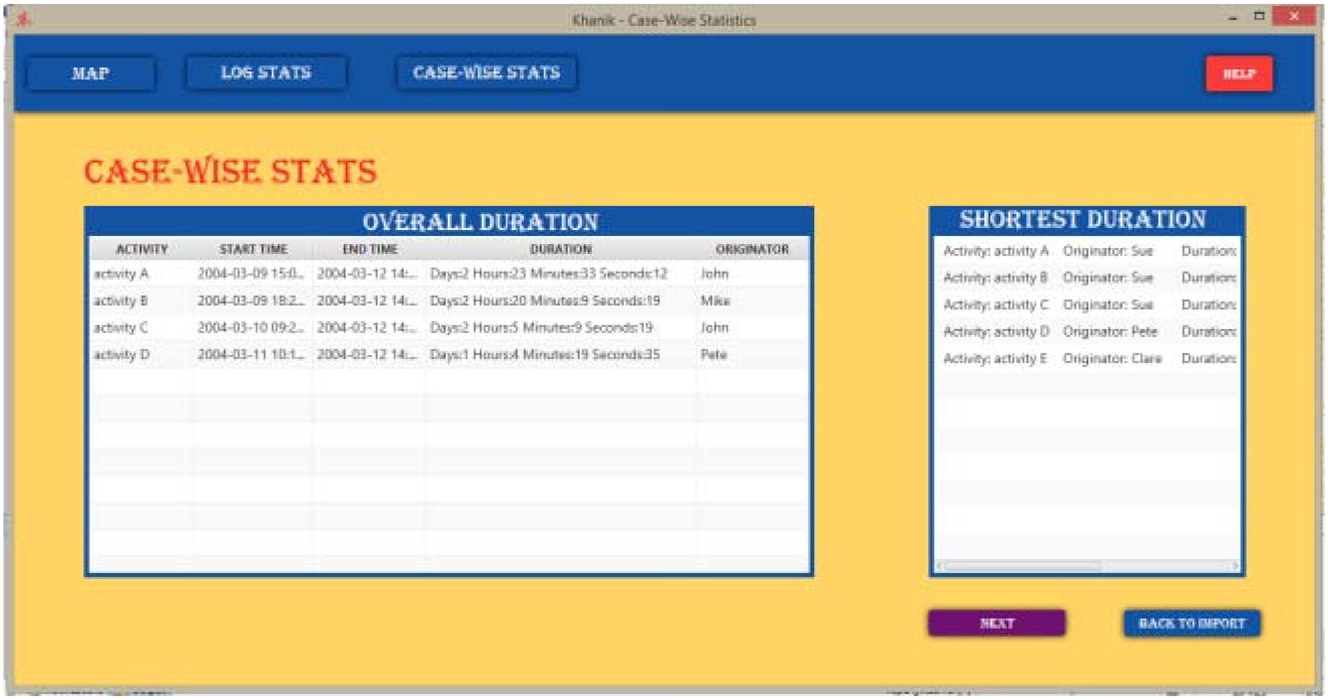
Figure4. Log Statistics

Case-Wise statistics feature provided by Khanik offers the user a whole new perspective on the data mined by the software. It displays each activity present in the logs with its starting and ending time, duration, and also displays the originator who has performed the activity. This helps the user judge the efficiency of different originators. This feature also helps the user to identify activities which are more time consuming, and hence cause the project to be delayed. It also displays for each activity, the originator who has completed it in least time. Fig. 5 displays an example of case-wise statistics.

VI. CONCLUSION AND FUTURE WORK

In this work we have presented the concept of Process Mining and developed an application which implements this concept. The goal of process mining is to find, observe, and improve processes by extracting their execution traces. By using process mining and developing software we can optimize the Software Development Lifecycle of projects done by IT companies. By feeding the event logs of prior conducted projects of similar nature, the software will develop a Business Process Model. The logs are stored in Microsoft Excel sheets. It makes use of Heuristics Miner Algorithm for mining the logs and Apache POI for XML conversion which is used for the importing of the Excel sheets.

The performed work would provide insight and help identify the antecedents needed for the current



OVERALL DURATION

ACTIVITY	START TIME	END TIME	DURATION	ORIGINATOR
activity A	2004-03-09 15:0...	2004-03-12 14:...	Days:2 Hours:23 Minutes:33 Seconds:12	John
activity B	2004-03-09 18:2...	2004-03-12 14:...	Days:2 Hours:20 Minutes:9 Seconds:19	Mike
activity C	2004-03-10 09:2...	2004-03-12 14:...	Days:2 Hours:5 Minutes:9 Seconds:19	John
activity D	2004-03-11 10:1...	2004-03-12 14:...	Days:1 Hours:4 Minutes:19 Seconds:35	Pete

SHORTEST DURATION

Activity: activity A	Originator: Sue	Duration:
Activity: activity B	Originator: Sue	Duration:
Activity: activity C	Originator: Sue	Duration:
Activity: activity D	Originator: Pete	Duration:
Activity: activity E	Originator: Clare	Duration:

Figure5. Case-wise statistics

project. Thus the software could be an asset for project management

So far, process mining has only been applied to operational processes, i.e., knowledge is extracted from event logs (process discovery). As future work, we propose to apply process mining to change logs, i.e. we not only study the operational processes but also the adaptations made at the process type or process instance level.

Using such analysis, we envisage being able to better support flexibility by understanding when and why process changes become necessary.

REFERENCES

[1] A. M. Lemos, C. C. Sabino, R. M. F. Lima, C. A. L. Oliveira," Using Process Mining in Software Development Process Management", Proc. IEEE Int. Conf. System, Man, Cybernatics.,2011.

[2] W.M.P. van der Aalst, A. Adriansyah,"Process Mining Manifesto", Springer-Verlag Business Process Management Workshops ,vol. 99, 2011.

[3] W.M.P. van der Aalst," Service Mining: Using Process Mining to Discover, Check, and Improve Service Behavior",IEEE Transactions on Services Computing, vol. 6, no. 4, October-December 2013.

[4] www.processmining.org

[5] van Dongen, Boudewijn F., et al. "The ProM framework: A new era in process mining tool support." Applications and Theory of Petri Nets 2005. Springer Berlin Heidelberg, 2005. 444-454.

[6] A.J.M.M. Weijters, W.M.P. van der Aalst, and A.K. Alves de Medeiros," Process Mining with the Heuristics Miner Algorithm",Department of Technology Management, Eindhoven University of Technology.