Towards a Conceptual Model for Undesired Situation Detection through Process Mining

Matheus Friedhein Flores[1], Denílson Ebling[1], Jonas Bulegon Gassen[2], Vinícius Maran[3] and Alencar Machado[1]
[1]*Colégio Politécnico, Universidade Federal de Santa Maria, Santa Maria, Brazil*
[2]*Antonio Meneguetti Faculdade, Santa Maria, Brazil*
[3]*Laboratory of Ubiquitous, Mobile and Applied Computing, Universidade Federal de Santa Maria, Cachoeira do Sul, Brazil*

Keywords: Process Mining, Situation Detection, Control Charts, Proactive Actions.

Abstract: As technology advances, recent research propose solutions to monitor and control organizational processes, aiming to maximize efficiency and productivity and minimize the loss of resources involved in the execution of processes, whether human or technological, in addition to maintaining a controlled environment so that the objectives of the organizations are achieved, that is, the satisfaction of their customers. For this, historical information contained in the event log is frequently used, related to the execution of processes in the organizational environment. These information serves as a basis for controlling the environment, preventing the occurrence of unwanted situations. In this context, this paper presents a model for detecting situations of interest in the organizational environment through event logs, making it possible to initiate proactive actions in the face of these situations, resulting in a Web application provided by interfaces that validate the purpose of the article. Beyond the scenario, an event log related to the execution of a real process was tested. By means of control charts, it is possible to view (using time parameters) the delay in the execution of the process, which may be related to a situation of interest.

1 INTRODUCTION

In contemporary organizations there might be considerable wastes of financial and human resources. The lack of planning and monitoring of existing processes may have a great impact on them. These factors may influence products or services towards low quality and loss of customers, among other factors that impact the direct form of the organization. To address these problems, organizations are increasingly investing in developing, expanding and improving internal processes through the use of areas such as: log mining, process control and monitoring (Van Der Aalst, 2011) (Saylam and Sahingoz, 2013). Managing processes improves organizations efficiency and allow them to reach their objectives, providing agile interactions with its customers. In addition to improving the quality and development of the work carried out, it minimizes internal problems and waste of resources, whether human or financial. Usually the management of processes is performed by means of software, which helps with versioning and execution (Laudon and Laudon, 2015).

Based on these processes, it is possible to identify desired behavioral parameters. For example, execution time of activities in a process can be used to identify if the process is running within the established quality limits of the organization in the provision of any service. In order to manipulate the information provided by the software that executes the processes, process mining research area arises, which proposes to suggest procedural improvements, discover control flow, performance analysis, compliance and monitoring tasks, characteristics that impact the internal and external environment of an organization (Van Der Aalst, 2011).

According to Van Der Aalst (2011), process mining is a research area that focuses on the extraction of information about the behavior of business processes executions. It uses log of events recorded by information systems available in organizations. Event logs, while not adding much value when viewed individually, when analyzed together, can be useful to understand or even improve and optimize the processes that make up a business environment. By analyzing the logs in sets, it is possible to define execution pat-

Flores, M., Ebling, D., Gassen, J., Maran, V. and Machado, A.
Towards a Conceptual Model for Undesired Situation Detection through Process Mining.
DOI: 10.5220/0009564408090816
In *Proceedings of the 22nd International Conference on Enterprise Information Systems (ICEIS 2020) - Volume 2*, pages 809-816
ISBN: 978-989-758-423-7

terns and consequently situations of interest in the environment in which the user is involved. The model proposed by Machado (2017) aims to manage an environment in which situations occur. Such situations generate events and the detection of patterns related to these events characterizes situations of interest, which can be ignored or not. In case of an unwanted situation, the model performs actions to control the environment and to avoid damage related to the monitored context.

It is worth mentioning that many organizations find difficulties in monitoring and controlling their processes at run time, by the fact that they produced information (logs), based on the execution of their processes, and do not know such as to best use this information. Influencing the final result of the product or service provided. This work presents a model to monitor and control the organizational environment, in order to avoid situations that negatively impact business processes. The proposed model is an extension of the model of proactive actions proposed by Machado (2017), with characteristics of the environment being added to the model so that it can be controlled, minimizing the occurrence of undesired situations. Taking into account the times and performance in the execution of process in the business environment, being able to act proactively or reactive in the face of environmental situations, if necessary.

The article is structured as follows: In Section 2, we present the main concepts found in the literature. The model developed in this work is presented in section 3. Section 4 presents a case study using the model and is followed by section 5 where results and discussions are presented. Finally, in section 6, we draw some conclusions and indicate points for future work.

2 BACKGROUND AND RELATED WORK

This section describes concepts that serve as a theoretical basis for the development of the work, such as: Process Mining, Log, Control Charts and Model to Proactive Identification of Situations of Interest, besides Related Work. According to Van Der Aalst (2011), process mining aims to extract knowledge from data generated by the execution of processes in information systems used by organizations. It seeks to discover, monitor and improve existing real processes. Process mining emerged in the 1990s, with the first works focusing on mining process models in software engineering event logs (Cook and Wolf, 1995).

Process mining is an area of research positioned between the areas of artificial intelligence, data mining, process modeling and analysis; therefore, the growing interest in the area can be justified by the fact that more and more events are recorded. (Van Der Aalst, 2016) (Burattin, 2013). The increasing volume of digital information related to processes in organizations allows the registration and analysis of their events. Any step or operation of a process or system can be seen as an event. (Van Der Aalst, 2012a).

In other words, process mining is important and efficient, as it is capable of converting historical information (logs), related to a process, enabling the organizations specialists to view, monitor and control what is really happening in the execution of processes. An event log consists of the record of events that occurred during the functioning of an organization, and the storage of this record is carried out mainly by information systems (Van Der Aalst, 2011)(Glavan, 2011). Events are considered as a tuple containing the following fields (Van Der Aalst, 2011): (i) **ID**: identification; (ii) **Timestamp**: activity start date and time; (iii) **Activity**: Activity description; (iv) **Resource**: responsible for the activity; (v) **Cost**: activity cost; (vi) **Extra data**: additional information about an activity.

The log data can be located in a distributed and incomplete way, inconsistent with the reality and containing outliers (noise). Additionally to quiet activities that are performed in the company and depend on third parties, therefore not being registered in the system (Van Der Aalst, 2012b).

In order to measure the parameters provided by the logs, control charts can be used. Therefore, it is possible to define upper and lower limits, so that metrics and can identify the occurrence of anomalies in the stored records. In addition to monitoring and controlling the actions involving these records. According to Oliveira (2013), control charts are tools used to monitor the performance of a process, based on characteristics that they call control limits. These limits are known as (i) upper line (upper control limit - LSC) and (ii) lower line (lower control limit - LIC), in addition to (iii) central line (central limit - LC).

According to Oliveira (2013), when all the sample points are within the limits of control, it is considered that the process is "under control". However, if one (or more) points are positioned outside the imposed control limits, there is evidence that the process is "out of control" and that an investigation into the occurrences and corrective actions are needed to detect and eliminate special causes in the process. Therefore, after sample analysis, it is possible to define whether a situation in the environment is of interest, that is, characterizes an uncontrolled environment (Machado, 2017).

The work presented in Machado (2017) deals with the identification and management of situations of interest, whether unwanted or not, in intelligent environments. It seeks to perform actions (using web services) in the face of such situations. The work proposes an approach for systems to act in these situations, trying to eliminate or soften their impact. In addition, intends to expand their ability to manage new unwanted situations that arise in the user's life environment over time.

Machado (2017) defines that situations are composed of events, being monitored by the middleware that provides some type of service. These events take place in a valid time window called a sliding window, and the correlations of these events can highlight a situation. If a situation of interest occurs and it is unwanted, actions can be taken to control the environment, which can be reactive or proactive. For more information about the model and concepts, see (Machado, 2015).

An event has a name, being characterized by a type (internal or external) and has a time within Δ td (flow of events monitored as data time), and a set of semantic relations of the context R. When an event is not produced by a simple entity (raw data from a sensor or simple record of the consumption of a service), it must also contain a standard (p) for its detection (Machado, 2017) (1):

$$Event : (name, type, time, \{R\}, p) \tag{1}$$

As Machado (2017) describes, events evaluation (a standard) can make the system identify the likelihood of an unwanted situation to happen in the future. Other works, such as Forkan (2015) and Coronato (2014), define that in order to understand future changes in user behavior, a system must consider proactive actions and current events. So, an approach can understand the user's behavioral deviation using pattern recognition models. Seeking to validate the objective of this work, as well as to compare it with the literature, arise the work proposed by Mannhardt and Landmark (2019). It aims to apply process mining techniques to the rail traffic control event logs (case study), allowing to investigate the quality of decisions made by those responsible for the railroad.

The work proposed by Tax and van der Aalst (2019), conceptualizes techniques to filter chaotic activities from event logs. The case study used by the authors constitutes a collection of seventeen (17) event logs derived from the execution of real life business processes. Based on the authors' results, they argue that chaotic activities affect the quality of process models.

Polato (2018) proposes a model to predict times and deal with unexpected behaviors based on the additional data present in the event logs. For this, the authors use machine learning techniques to train algorithms and detect unexpected behavior.

Based on the works mentioned above, it is possible to analyze that all works deal with some fact that occurred as a chaotic situation, similar to the unwanted situation, which may be infrequent or frequent, depending on the case. However, none of the aforementioned works contains characteristics that make it possible to take actions in relation to situations of interest, minimizing their impact on the related environment. The present work describes, in the next section, a model that allows controlling and monitoring the processes, detecting situations of interest and, if undesired, executing defined reactive actions.

3 A SITUATION DETECTION MODEL FOR PROCESS MONITORING AND CONTROL

In the work of Machado (2017), the focus of the research refers to assisted living environments, managing situations that occur in that environments, with the help of several sensors to capture contextual information of users with cognitive problems or with advanced age. But, as conceptualized by Machado, the model was proposed for generic environments, that is, it can be used in any context.

In the organizational context, the model from Machado (2017) can be extended for the control and monitoring of processes. Considering how to perform actions on behalf of the organization, based on information generated during the execution of processes, in addition to being able to detect and interpret situations that occur in the environment. Based on the event log (Section 2 - log), which contains the historical information stored about actions that have occurred, it is possible to define parameters indicating when a process is exposed to some situation of interest. These parameters are related to the execution of the processes. In this work, the time and noise parameters are taken into account. To improve understanding of the business and the applied model, Figure 1 illustrates an overview of the business and the context around all actions in the process.

- a) Figure 1 illustrates the environment in which the processes are executed, starting with the client's negotiation with the business manager (team responsible for the organization's negotiations). In this negotiation, some information is generated, in the presence of the customer and without the presence of the customer, such as fill-

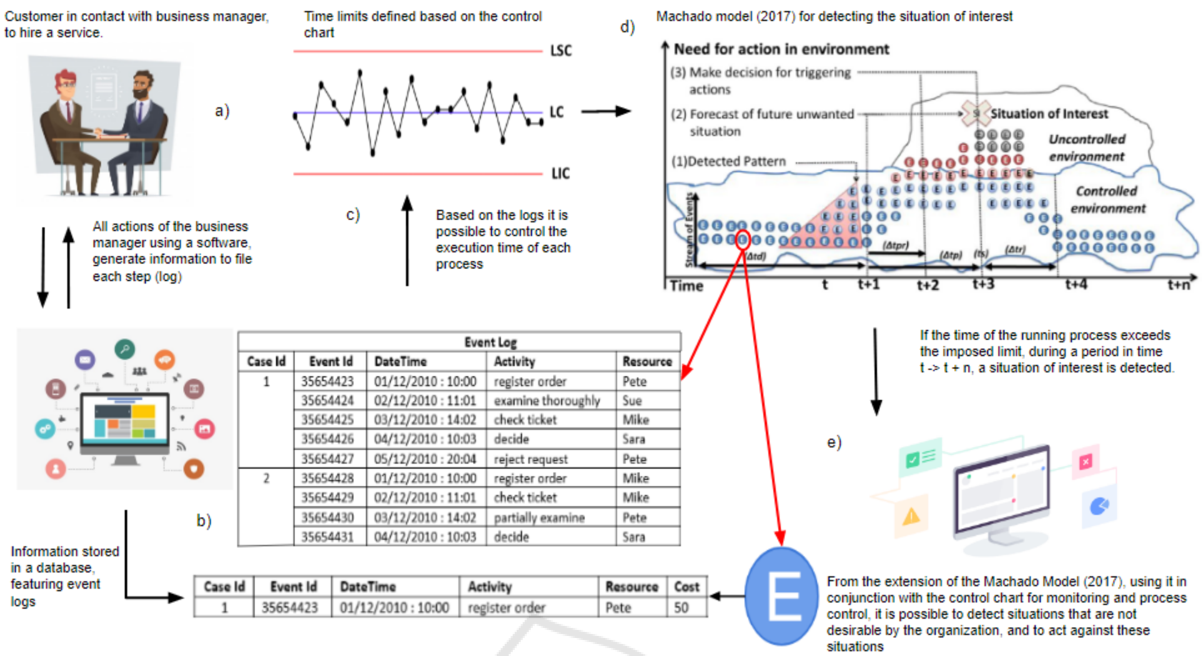

Figure 1: Overview of the organizational context.

ing out records or updating personal information.

- b) The process starts in this negotiation, but it is not limited to this interaction between the client and the business manager, but passes to the internal environment of the organization, being processed between different activity until it is rejected or accepted. In each activity where the process is processed, various information are generated, for example, who was responsible for processing another activity, what time it occurred, such as long it took to process, whether the process was idle or not. All of this information generated are events and each event is saved in the database, resulting in the event log of the process.
- c) These processes are controlled based on the event log, taking into account the time factor of the processes execution.
- d) Using the control chart(Section 2), it is possible to define upper and lower bounds to compare processes with execution times within the expected time and special causes, times that exceed the imposed limits and allow to indicate whether a process is in progress . a controlled or uncontrolled environment due to the detection of event patterns.
- e) Based on the work of Machado (2017), which consists of managing smart environments using middleware and applications, fed by information generated by various sensors embedded in the environment, aiming to control and monitor to minimize or prevent the user involved from putting himself in unwanted situations, however, in this article, the information is provided by the event logs and managed by the middleware provided by an application.

The objective of the present work is similar to that proposed by Machado (2017), however, here the information is provided by log events and not by sensors. The conceptual model proposed by Machado can be applied and extended to the context of this paper. For this, some concepts were added to meet the demands and needs of the organizational environment, illustrated in the figure 2, such as:

- **(a) Legacy systems**, related to third-party systems used for customer service and business process organization. Software to automate routines and bureaucratic procedures, with computerized services that provide the data to be manipulated and managed, aiming to control the organizational environment.
- **(b) Noise**, incorrect information that can negatively influence time calculations, that is, for unforeseen reasons, the information can be misleading, for example, the person responsible for processing the process in a certain sector, went on vacation, but the process was not passed on to anyone, becoming idle, consequently increasing the execution time and negatively impacting the organization's image in relation to the client.

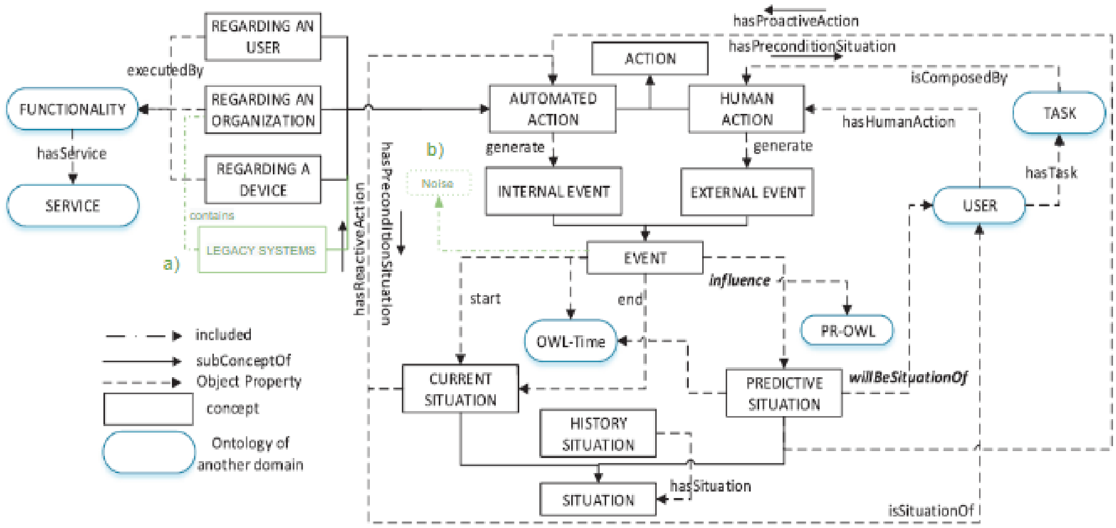

Figure 2: Proactive model for extended business processes.

As described in concept **(b)**, data provided by third-party software saved in logs may present errors, due to logic, commercial rule, technology or other unforeseen factors. To minimize errors when using information from the logs, all data must be filtered and analyzed, so that they can be processed as execution occurs.

For each data in the log, when it is filtered, it is subjected to a base calculation to be categorized as noise or not. The base calculation is equivalent to an accepted time value above the average time defined by the historical data of the process under analysis. If the average time is 10 (ten) minutes for the execution of the process, only events with a time greater than the average time in 3 (three) times the average will be categorized as noise (2), that is, if process Y is related to a situation of interest X, which takes an average of 10 (ten) minutes to execute, information over time of 30 minutes is disregarded and is not taken into account in detecting the situation of interest.

$$Noise = time > (3 * average) \quad (2)$$

The information, categorized as noise, is ignored because, if taken into account, they end up interfering in the final result of the moments when the process is executed, causing errors and inconsistent information in the sample being controlled and monitored. In order to differentiate between with and without noise, the noise attribute was added to the event proposed by Machado (2017) (3), so that it is possible to distinguish events that can be analyzed from those that should be discarded.

$$Event : (name, type, time, noise, \{R\}, p) \quad (3)$$

Given the logs, having the middleware and a system to monitor and control running processes, it is possible to use the time parameter, defined by means of historical information, to audit the processes and discover situations of interest. In addition to determine actions for such situations. For this, the system provides an algorithm with equations and probabilities that manipulate the times of execution of a process. Such as calculation of times by process, in which it interprets individual values to define parameters to be used in comparisons of execution times, saved in the event logs. During execution, the process is monitored and controlled by the diffused system that filters data equivalent to noise and detects patterns in times that exceed the limits defined by the control chart for a certain period, possibly characterizing an unwanted situation. To validate the objective and the extended model, the next section describes the case study with an empirical scenario, which characterizes an organizational environment.

4 CASE STUDY SCENARIO

In order to validate the model described in this work, a scenario that reflects the routine events of a banking organization was elaborated. The scenario consists of a business environment in which data is created, taking into account the interactions of a customer, who requests a product, and an organization, which sells that product. Let us call the situation produced in this scenario as *"Customer dissatisfaction"* initiated by the event *"credit denied"*.

The following scenario was considered to apply the extension of Machado (2017) to another context. *John, 26, is a development analyst and works for a* *private technology company. One day, John decided to buy a vehicle, but he did not have the total amount necessary for the purchase. Given this fact, he chose to go to a bank and apply for a credit to complete the required amount.*

The organization where John is a customer is one of the best known in the banking sector. It has almost all of its processes computerized and well-structured, with the aim of always meet customer needs. Previously to the decision of buying a vehicle, John had already negotiated with the organization. In the prior negotiation, he asked for a construction credit (residential credit line) to finish his apartment and had this credit approved. In such case, the "credit denied" event was not detected, consequently, not causing the situation "customer dissatisfaction".

Satisfied, John returns to the organization in order to request credit for the purchase of the desired vehicle, in a hurry to close the deal with the seller. Historically, the vehicles credit line moves faster than other credit lines in this organization. However, in John's request, there were some flaws in the execution of the process. Due to the delay in the execution of the process, John decided to change the vehicles credit line to the simple credit line, used when users have been customers for some time. But it was different from the first contact with the organization and the usual cases of the same process. When John requested the related credit with the simple credit line, his request was denied, causing the organization to experience an unwanted "customer dissatisfaction" situation.

In order for the unwanted situation to be detected, the "credit denied" event initiates the unwanted situation of "customer dissatisfaction"; however, this situation occurs only when another related event is detected, when Peter (commercial manager) offers another line of credit as an option for John to leave with the approved credit, being "offer another line of credit" the detected event.

The occurrence of these events negatively affects the organization, as it is the opposite of their strategic objective: customer satisfaction. In view of this situation, the organization has defined some measures to manage its processes, seeking to avoid situations like these. One measure was to deploy a middleware and a comprehensive application to monitor and manage the business environment. All information resulting from the customer interactions with the organization is managed by the middleware and the application, as the two together have the ability to assist the organization in the execution of its processes. It helps to control the emergence of situations, as well as to define proactive and reactive actions when such situations occur.

In the case of John, when a situation of interest "credit denied" is detected, the application immediately asks the middleware to send a notification through some communication channel to the organization's manager, taking into account that he takes some action in relation to the event that occurred, in order to minimize the damage for both sides.

The scenario described above demonstrates that a situation of interest was detected, given this fact, extending the Machado (2017) model, a application triggers reactions to control the environment in question, as described in the next section.

4.1 Behavior for Credit Denied Situations

When installed, the comprehensive application informs the middleware based on the event logs, their context and their situation of interest, according to the parameters defined for the comparisons made in the extended model (section 3), disregarding noisy data. If a running process detects that John received the denied credit (the ea1 event was detected), the "customer dissatisfaction" situation is initiated by the application and the application chooses an action to be triggered, with action a1 sending a notification through a communication channel to activate the responsible personnel.

Current Situation:	< (customer dissatisfaction), (ea1), (ea2) >
Event name	ea1
Description	credit denied
Type	Internal
Noise	false
{R}	<John, GoToAgency, RequestCredit> ∧ <RequestCredit, CheckDocuments, ValidDocumentation>∧ < RequestCredit, = , InValidDocumentation > ⇒ <John, creditRequested, denied>
Pattern	SELECT e FROM PATTERN [every e=Event (name= 'ea1')]

With the continuous monitoring of the environment in relation to the events that occur, after sending the notification, the system understands that the situation of "customer dissatisfaction" has ended.

As shown above, if the ea2 event is detected, John

Event name	ea2
Description	credit denied
Type	Internal
Noise	false
{R}	<John ,GoToAgency, RequestCredit> ∧ <RequestCredit, CheckDocuments, ValidDocumentation> ∧<ValidDocumentation, containsMinimumRequirements, RequirementsOk> ∧<RequestCredit, = , creditReleased> =<John, RequestCredit, released>
Pattern	SELECT e FROM PATTERN [every]e=Event (name= 'ea2')]

has his credit released, there is no situation of "customer dissatisfaction". According to the model proposed by Machado (2017) (section 2), the actions are activated corresponding to the events generated, as described in the scenario, action a1 generates event a1, action a2 generates event a2 and so on. The actions for the *"customer dissatisfaction"* situation are described below, being declared as Action (name, Sp, Se, E) (Machado, 2015). Each action generates an event that the application runs to control the environment and the detected situation, in order to avoid losses.

Name:	Notification by whatsapp after ea1;
{Sp}	credit denied;
Services used {Se}	
Se	notifyWpp
{I}	Device, "checkDocumentation";
{O}:	DeviceMobile, SendMessageWhatsApp
Services resulting in action {E}	
Name	a1
Type	Internal
Noise	false
{R}	notifyApp_ ActionAutomated, haveService, notifyService> ∧ notificationService, aTypeOf, interaction_User>
Pattern	SELECT e FROM PATTERN [every e=Event(name ='a1')-> and Event(name ='ea1'))]

5 RESULTS AND DISCUSSION

Based on the scenario described above, where the environment in which the processes are executed is conceptualized, a prototype was developed, which is in the testing phase. The web application intends to assist the monitoring and control of processes. It was implemented in an organization, being tested in the real environment with customers. The data is provided by internal software used by the organization to manage its daily tasks, which are stored in the event log from where the application consumes information.

The application consists of some interfaces, one of which is illustrated in Figure 3, related to the execution schedule of a single process from *"General Credit"* type, with the times and their respective dates as the stored execution occurred in the event log. The application has an interface to perform searches and generate reports related to the execution times of the processes based on the event logs. As shown in figure 3, there are some fields that allow you to specify the period of the start and end dates, the type of process, in addition to the area and the activity in which it is found, information that helps to seek more accurate data about the process.

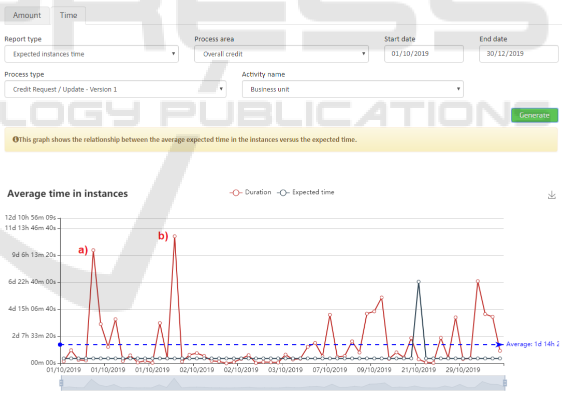

Figure 3: Graphical interface for monitoring and controlling process.

When entering the chosen fields for the search, a graph related to the control charts is displayed, showing the estimated times on the X axis and the date on which the fact occurred in Y, as well as lines and points that characterize the average time (blue dotted line), the time peaks of some processes that occurred longer than expected (red line) and the expected time (black line).

It is possible to observe that most of the points are above the expected schedule and another one even higher than the average, being very extreme cases. In cases of very high times, the analysis of special cases would be made, being calculated with the accepted percentage, that is, in relation to figure 3, cases (a) and (b) would be categorized as noise, greater than 3 times the average of all times in the sample, being removed from the analysis to detect the situation of interest. However, it is worth mentioning that, with the period of execution of the inserted process, with start and end date, it turns out that the times occurred exceed the expected time per process, detecting the undesired situation resulting from the delay in the execution of the process, initiating proactive actions by the application, which can be, as described in the case study, a warning message to the person responsible for the process.

6 CONCLUSION

In this work, a model that extends to that proposed by Machado (2017) was presented. Using the extended model, it was possible to monitor and control the processes, to minimize losses due to unwanted situations.

The contributions of the work can be described as: (i) it is possible to detect an intersection situation based on historical data provided by the software that assist in the execution of organizational processes; (ii) reactive actions in the face of unwanted situations that meet the objectives imposed by the organization and control the organizational environment; (iii) improvement in the flow of process executions, reducing the number of occurrences and unwanted situations, increasing efficiency in the business environment.

Thus, we understand that the expanded model, together with the areas of mining, monitoring and process control, have much to add to organizations concerned with developing and improving their processes. Aiming to gain efficiency, productivity and safety, seeking to provide services or products with added quality to their customers.

REFERENCES

Burattin, A. (2013). *Applicability of process mining techniques in business environments*. Doctoral diss., alma.

Cook and Wolf, A. (1995). *Automating process discovery through event-data analysis*. ICSE '95, pages 7382, New York, NY, USA, 1995. ACM. 22, 27, 34.

Coronato, A.; Paragliola, G. (2014). *Towards a Personal Health Records system for patients with Autism Spectrum Disorders. In: Computational Intelligence in Healthcare and e-Health*. Proceedings Symposium Series on Computational Intelligence.

Forkan, A. R. M. e. a. (2015). *A context-aware approach for long-term behavioural change detection and abnormality prediction in ambient assisted living*. Transactions on Management Information Systems (TMIS), 3(2), 1-17.

Glavan, L. (2011). *Understanding process performance measurement systems*. Business Systems Research, 2(2), 25-38. 30, 35, 39.

Laudon, K. C. and Laudon, J. P. (2015). *Management Information Systems: Managing the Digital Firm Plus MyMISLab with Pearson eText*. Prentice Hall Press, Upper Saddle River, NJ, USA.

Machado, A. (2015). *Sensibilidade à situação em ambientes de vivência assistida: uma abordagem reativa, proativa e extensível*.

Machado, Alencar, e. a. (2017). Reactive, proactive, and extensible situation-awareness in ambient assisted living. Systems with Applications 76 (2017): 21-35.

Mannhardt, F. and Landmark, A. D. (2019). *Mining railway traffic control logs*. Transportation research procedia 37.

Oliveira, C. C., e. a. (2013). *Manual para elaboração de cartas de controle para monitoramento de processos de medição quantitativos em laboratórios de ensaio*. Instituto Adolfo Lutz, São Paulo.

Polato, Mirko, e. a. (2018). *Time and activity sequence prediction of business process instances*. Computing 100.9.

Saylam, R. and Sahingoz, O. K. (2013). *Process mining in business process management: Concepts and challenges*. In International Conference on Electronics, Computer and Computation (ICECCO).

Tax, Niek, N. S. and van der Aalst, W. M. (2019). *Discovering more precise process models from event logs by filtering out chaotic activities*. Journal of Intelligent Information Systems.

Van Der Aalst, W., A. A. D. M. A. K. A. A. F. B. T. B. T. . . B. A. (2012a). *Process mining manifesto*. In 9th International Conference on Business Process Management, BPM 2011P (pp. 169-194). Springer Verlag. 12, 13, 14, 15, 16, 17,18, 19, 20.

Van Der Aalst, W. (2011). *Process mining: discovery, conformance and enhancement of business processes Vol(2)*. Heidelberg: Springer.

Van Der Aalst, W. (2012b). *Process mining: Overview and opportunities*. ACM Transactions on Management Information Systems (TMIS), 3(2), 1-17.

Van Der Aalst, W. (2016). *Data science in action. In Process mining*. Springer, Berlin, Heidelberg.