European Journal of Computer Science and Information Technology, 13(51),119-137, 2025
Print ISSN: 2054-0957 (Print)
Online ISSN: 2054-0965 (Online)
Website: https://www.eajournals.org/
Publication of the European Centre for Research Training and Development -UK

Analyzing Administrative Process Sequences Using PM4Py: A Case Study in an Albanian Municipality

Anxhela Kosta
Department of Computer Sciences, University of Tirana, Albania

Ilma Lili
Department of Computer Sciences, University of Tirana, Albania

Endrit Xhina
Department of Computer Sciences, University of Tirana, Albania

doi: https://doi.org/10.37745/ejcsit.2013/vol13n51119137 Published August 26 2025

Citation: Kosta A., Lili I. and Xhina E. (2025) Analyzing Administrative Process Sequences Using PM4Py: A Case Study in an Albanian Municipality, *European Journal of Computer Science and Information Technology,* 13(51),119-137

Abstract: *Inefficiency, delays, and procedural redundancies are often the hindering elements in administrative processes in local government institutions, especially in municipalities in Albania. When citizens make a specific request to the municipality for a specific process, there are many cases when they never receive a response and do not know what happens to their request, at what step it is, what is missing, when it will be approved, etc. For this reason, this paper presents a focused approach to analyze the sequences of work processes that occur on the Albania administrative side with the aim of identifying inefficiencies and opportunities for improvement through Process Mining. Initially, it was necessary to construct a real dataset of events capturing the execution of municipal services. We developed a SQL Server database structured in five main tables to generate event logs, comprising approximately 366,000 cases, 620,205 events, and 33 unique activities across multiple municipal workflows. Through Pm4Py we analyze and visualize all work process sequences. The goal was to identify inefficiencies and understand how we can optimize the workflow sequences. Through Heuristic and Alpha miner algorithms we analyse and model the processes to build a structured model in the form of a graph such as Petri Net and Heuristic, providing a clear and consistent visualization. Petri facilitates the identification of delays, interruptions and sequences of repeated actions. Pm4py plays a crucial role in accelerating and increasing the efficiency of model building at this stage of the process. Future work, the study will focus on clustering the SNA(social network analysis) to identify relevant anomalies for each process. This clustering analysis aims to identify and classify potential anomalies in the structure and performance of local administrative workflows, providing insights for further optimization efforts.*

Keywords: Process Mining, Workflow Optimization, PM4Py, Public Administration, Petri Net, Heuristic Miner, Transparency

INTRODUCTION

At a time when public administration faces major and ongoing challenges to be more efficient, transparent and accountable to citizens, the role of municipalities in Albania becomes more important than ever. As among the key public bodies at the local level, local self-government institutions have the main role in providing a wide range of services that facilitate the daily lives of citizens, so that any of their requests or procedural actions are administered with transparency, efficiency and justice. However, the daily functioning of public administration is affected by numerous problems, such as long and unnecessary bureaucracies, repetition of processes, procedural delays and interruptions during the handling of requests, which negatively affect the performance of these bodies and the quality of the services they provide. As a result, these problems significantly reduce the operational efficiency of local institutions, but above all they significantly undermine the public's trust in the administration, thus creating the perception that public services are slow, non-transparent and difficult to access. One of the main causes of these difficulties is related to the fact that a significant part of administrative processes continue to be managed in a traditional way and still rely on outdated methods, based on manual documentation, theoretical descriptions of work processes and subjective reporting that do not reflect the daily operational reality in an accurate and measurable way. To overcome this gap, new modern techniques offer good opportunities for transformation. One such technique is process mining - an innovative approach that enables the analysis of real data traces (event logs) that are automatically generated by existing information systems. Such a method enables the identification of the current flow of processes, the identification of deviations, problems and ineffective steps. Through this approach, it is possible to visualize administrative work processes, helping local institutions in making more effective decisions.

This study focuses on the practical application of process mining techniques in an Albanian municipality, using a real database structured in SQL. The processing of these data was carried out through the PM4Py library in the Python language, which offers a wide range of functions for building process models, performance analysis and identifying anomalies. The purpose of the study is to provide a clear presentation of the current functioning of administrative processes (366,000 cases, 620,205 events, and 33 unique activities across multiple municipal workflows), on the identification of critical points that require improvement and to draft concrete, structured recommendations for increasing institutional efficiency. In conclusion, the main goal is to contribute to building a local administration with a more digitalized approach, with higher sustainability and more citizen-oriented — an administration where decision-making is based on real data and where transparency and accountability are achievable standards of public functioning and not just general theoretical principles.

Literature Review

During our research to understand which services are most frequently requested by citizens and which need to be handled in a timely and transparent manner, we identified a specific set of administrative processes carried out in a municipality in Albania. We chose the municipality as the unit of study, as it represents the closest level of local government to the citizen and aims to provide clear and structured responses to any request submitted. In order to analyze how these processes are carried out in practice

— and to understand whether they are implemented in accordance with procedures, without delays or deviations — we conducted a preliminary study on the most appropriate techniques and technologies for this type of analysis. As a result, we identified process mining as the most appropriate approach to analyze the sequences of processes that occur in practice within the municipal administration.

Why Process Mining?

Process mining represents a dynamic interdisciplinary field that brings together data science, software engineering and business process management, with the aim of analyzing, understanding and improving operational flows in public and private organizations. Increased availability of event data from information systems over the past few years has opened up new opportunities for process analysis and optimization of organizational procedures. Process mining has proven to be a valuable technique standing at the intersection of data mining and business process modeling by enabling the derivation of actionable knowledge from real execution logs of business processes (van der Aalst et al., 2012). In contrast to traditional techniques employing static models or process documentation, process mining allows one to discover, monitor, and improve real processes on the basis of stored data, resulting in a fact-based foundation for decisions. The application of process mining in the context of public administration is particularly valuable given the increasing demands for transparency, accountability, and efficiency in service delivery. (Van der Aalst et al., 2007), demonstrated that process mining has the potential to reveal control-flow patterns, organizational bottlenecks, and deviations or rework in actual workflows. These competencies are especially crucial in the public sector, where inefficiency and procedural opacity can have a grave effect on citizen trust and administrative effectiveness. Considering these strengths, this paper applies process mining as the analytical framework to evaluate workflows in Albanian municipalities. Using real event logs and PM4Py—a process mining library for Python—we aim to uncover the real behavior of administrative processes, identify points of inefficiency, and provide concrete recommendations for data-driven governance.

Tools and Technologies in Process Mining

The most advanced software tools have had a great development and progress especially in the last 10 years, this has made automated processes have a development and applicability, especially in scientific research and in the academic industry. There are several platforms that are most used such as Prom, Disco, Celonis and Pm4py. Each of the tools, depending on the cases they are used, has specific functions and according to the research and the audience required and especially according to the level of analytical flexibility required to better understand how these tools change or adapt in practice we will see in the table below. This is a summary of each of these tools with their strengths and weaknesses from previous knowledge.

Table 1. Comparative overview of Process Mining Tools

Tool	Description	Pros	Cros
ProM	A large open-source process mining library written in Java with educational intentions primarily in focus. It encompasses a wide range of process mining techniques including discovery, conformance checking, enrichment, and social network analysis through over 1,000 plug-ins. Well-suited for prototyping and designing algorithms. (van der Aalst 2009)	- Wide algorithm support - Customizable plug-ins - Extensive research history	- Outdated interface - Requires Java setup - Steep learning curve
Disco	A commercial desktop application developed by Fluxicon used for simple process discovery and performance measurement. It automatically creates process maps and KPIs from event logs and is widely practiced since it's quick, simple, and graphical. (Palangsantikul, 2024)	- User-friendly GUI - Real-time metrics - Fast analysis for business users	- Proprietary license - Limited algorithmic depth - No plugin support
Celonis	A cloud-based business-grade platform that combines process mining with automation, AI, and business intelligence. Used by large-scale organizations for end-to-end process visibility, tracking of KPIs, anomaly detection, and integration with ERP systems like SAP and Oracle. (Badakhsha, 2020)	- Powerful enterprise integration - AI-driven insights - Real-time dashboards	- High cost - Not suitable for research customization - Closed architecture
Pm4Py	A modern, open-source Python library for process mining for researchers and data scientists. It supports discovery algorithms like Alpha Miner, Heuristic Miner, and Inductive Miner, in addition to Petri net modeling, performance analysis, and conformance checking. It can be easily integrated into data pipelines with Pandas, NumPy, and Jupyter (Berti, 2023).	- Full Python compatibility - Open-source and extensible - Suitable for real datasets and experimentation	- Requires coding experience - No visual GUI - Less accessible for non-programmers

Applications of Process Mining in the Public Sector

Process mining has increasingly been applied across various public sector domains, such as municipal services, healthcare, education, and social welfare, in recent years. These areas are typically characterized by complex processes, decentralized information systems, and manual reporting, which lower the productivity of services along with citizen confidence. Process mining offers a data-centric approach to counter such problems by exposing the actual implementation of administrative processes, allowing bottlenecks, delays, and non-compliant behavior to be traceable (Rawiro et al., 2022). Early adoptions were reported in China and the Netherlands in 2009, when process mining was applied on government data logs and institutional financial collection systems (Li & Deng, 2009; Măruşter & Van Beest, 2009). The Netherlands once more took the forefront in this field by implementing process mining to government institutions in the Netherlands and building permit applications (Bozkaya et al., 2009; Park & Kang, 2016). In Korea, the technique was employed to control traceability systems of beef imported into the country, while in France, it helped in Emergency Medical Assistance Centre processes (Kang et al., 2013; Lamine et al., 2015). By 2016–2018, applications had gone global: the USA used process mining for military and health projects (Bentley et al., 2017; Haq et al., 2016), India and Hungary employed it in public administration and civil registration (Shrivastava & Pal, 2017; Molnár, 2017), whereas Romania, Greece, and Russia implemented it to energy monitoring, university library systems, and e-government portals respectively (Repta et al., 2018; Kouzari & Stamelos, 2018; Kalenkova et al., 2018). In Germany, it has been used by the requirements that Germany has from the European Union for the purchase of agricultural products (Santoro et al., 2020) and Norway's request for the analysis of child labor (Larsson, 2021). The studies mentioned above highlight the importance of increasing process mining in the reengineering of the functions of the sector, how process mining affects it and why other countries, such as Albania, should use it.

AI and Process Mining for Workflow Optimization

Traditional process mining is focused on process pattern discovery and deviation, and does not focus on automating the processes. This is where Artificial Intelligence comes to the rescue. The advantages of such methods enable systems to learn from previous behaviors and generate proactive insights. Integrating Artificial Intelligence (AI) into process mining helps identify complex patterns, predict future outcomes, and automate decision-making in dynamic environments. However, AI-powered methods go beyond this basic function, allowing learning from historical data, identifying hidden patterns, and generating proactive recommendations for process improvement (Berti et al., 2023). Advances in prompt engineering have also facilitated the integration of large language models, such as ChatGPT, into the context of process mining, enabling linguistic analysis of logs, automation of analytical query generation, and more intuitive interpretation of results for non-technical decision makers (Jessen et al., 2023). One such example is the use of AI technologies to analyze scientific workflows in High-Performance Computing environments, where process mining helps optimize task execution and reduce response times (Jessen et al., 2023). Beyond language processing, AI has also been used to fragment and distribute process models across decentralized execution, enabling parallel management of activities in distributed systems and improving overall performance (Zeng et al., 2011). These developments show that combining AI with process mining not only provides accurate predictions on potential delays or deviations, but also allows for real-time adaptive optimization. Clustering algorithms group process instances with similar characteristics or paths, allowing administrators to identify behavior patterns that are linked to specific inefficiencies. Anomaly detection algorithms identify process executions that deviate from common patterns, flagging potential instances of fraud, error, or non-compliance (Tax et al., 2017). Additionally, predictive monitoring uses past logs and machine learning algorithms (i.e., LSTM networks) to forecast performance measures such as cycle time, delayed completions, or resource overload (Teinemaa et al., 2017). Social Network Analysis (SNA), another AI-supported technique, maps interaction patterns between departments or groups of people involved in a process, revealing communication bottlenecks and misaligned responsibilities. Recent advancements enabled the integration of these techniques in open-source software like PM4Py, which now includes clustering and SNA analysis modules, taking its applicability beyond discovery. In our work, future research will tackle the clustering of event log data for discovering abnormal process behavior and understanding variability among similar cases, particularly in high-volume municipal processes like permit requests and social aid requests.

Methodological Framework

This study follows an empirical and practice-oriented approach, with the aim of analyzing and optimizing the administrative processes of an Albanian municipality using advanced process mining techniques. The methodology followed consists of several consecutive phases: identification of work processes, data collection, database construction, preparation of event logs, process flow analysis and performance evaluation.

Process Identification and Data Collection

Before the practical work of the analysis began, a key step was to precisely define the work processes that we would analyze. The aim was to select processes that were mainly performed manually, had numerous human interventions, and were sensitive to delays or deviations. For this reason, it was decided to focus on the administrative processes of the municipality, which have a direct impact on citizens.
To ensure a complete overview of the services provided, physical interviews and working meetings were held with the persons responsible for the services in the municipality, including the information office and functional directorates. From these meetings, information was collected on the type of services, the procedures followed, the actors involved, and the documentation required for each process. This data served as the basis for building a database that reflected the main services.

Categorization of Services

To enable a complete and consistent analysis of administrative processes at the local level, an essential step was the identification, categorization and functional description of the services provided by the municipality. This analysis was carried out in collaboration with the Unit Information Office (Zi1N in one municipality of Albania), where official documentation and real data from the citizen requests and documents management system were provided. The aim was to understand not only the type of services, but also the practical way in which these processes flow at the operational level. After collecting information, the municipality's services were categorized into four main functional groups:

a) Social and Economic Services – e.g. applications for economic assistance, certificates of special abilities, etc
b) Urban and Property Services – e.g. verification of ownership, construction permits, approval of urban plans.
c) Transport Permits and Licenses – e.g. vehicle authorizations, public transport licenses.
d) General Administrative Requests – e.g. address verifications, copies of documents, information services.

The figure 1 show all the services in this municipality:

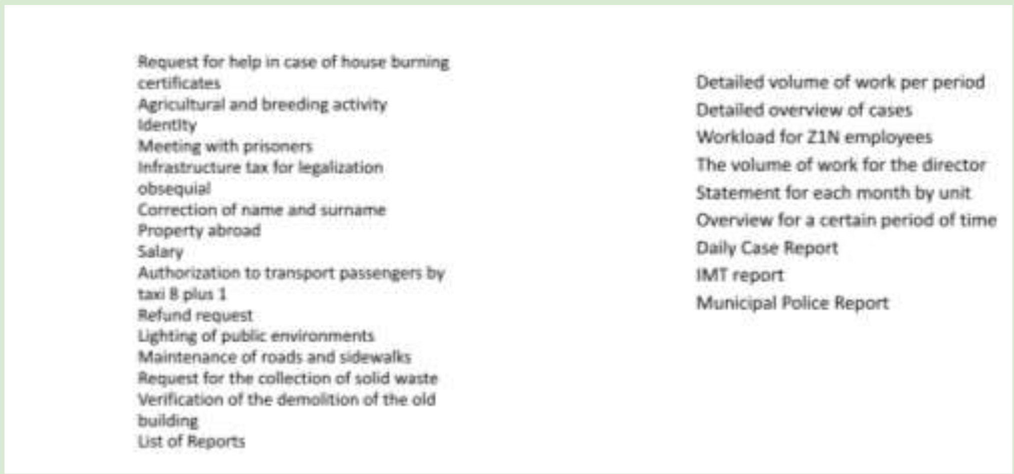

Figure 1. All service in One Albania Municipality

For each category, the practical workflow from the moment of application by the citizen to the submission of the final documentation was documented. This analysis was carried out by following the real steps in the process, not just those defined in administrative manuals. The categorization served as an initial filter to identify the most suitable processes for in-depth analysis through process mining. Priority was given to those processes that had:

- more sensitivity to delays,
- multiple manual interventions,
- branches or backtracking in the workflow.

In this way, the categorization did not only have an organizational purpose, but served as the first filter to determine the ideal candidates for in-depth analysis and optimization through technology. It also helped create a process map of municipal services that can be used for strategic and managerial purposes in the future.

Database Construction

A dedicated database structure was built to support subsequent analysis through process mining. The structure consisted of five main tables as is shown in Figure 1. :

- Process – stores the identifier and characteristics of each process (name, start and end date),
- Cases – represents individual citizen requests as separate instances of the process.

Print ISSN: 2054-0957 (Print)
Online ISSN: 2054-0965 (Online)
Website: https://www.eajournals.org/
Publication of the European Centre for Research Training and Development -UK

- ProcessStatus – documents the current status of each case.
- ProcessSteps – tracks the sequence of steps followed for each case.
- ProcessWorkflow – describes in detail the flow of actions, including the responsible actor and the time of execution.

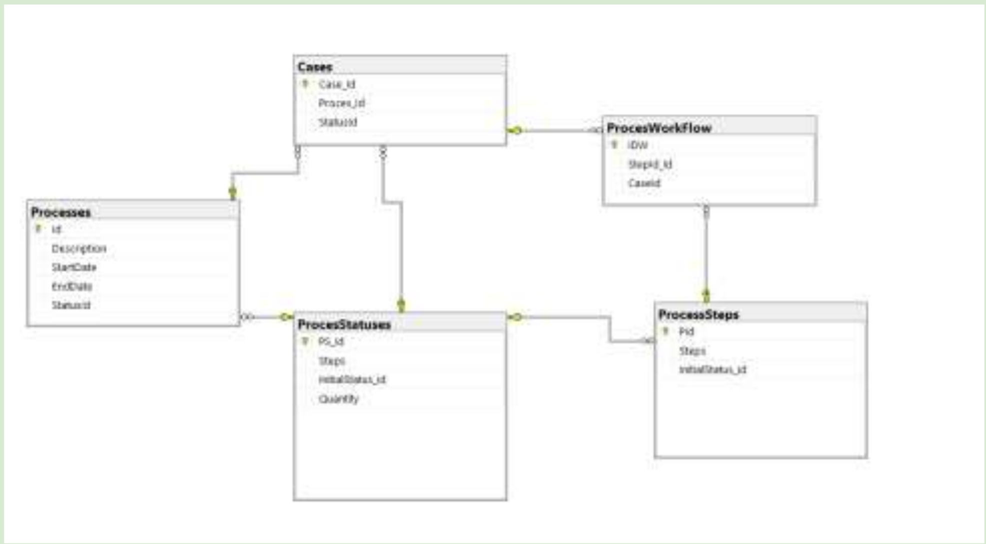

Figure 2. Star Sql Diagram created for Proces mining

This structure enabled a complete and measurable overview of the institution's internal operations.

Preparing Logs for Process Mining

The data stored in SQL was exported to CSV format and then transformed into formats suitable for process mining (event log) using the PM4PY library in Python. Each event log contained the essential fields for analysis:

- Case ID – unique identifier for each case (citizen request)
- Activity – name of the step taken (e.g., "verification", "submission", "response")
- Timestamp – exact time the activity was performed
- Actor – employee or department responsible for the activity (for confidentiality reasons, the specific assignee is not displayed in Figure 3 below).

Print ISSN: 2054-0957 (Print)
Online ISSN: 2054-0965 (Online)
Website: https://www.eajournals.org/
Publication of the European Centre for Research Training and Development -UK

Results | Messages

	CaseId	Activity	Timestamp	NameE	Asst
1	40003	Send for Liability Verification	2015-12-14 13:10:51.340	Goods certificate for own account	HAS
2	46648	accept the request	2016-01-07 08:32:06.980	Certificate that is treated with payment for different abilities	an
3	46648	Send for treatment	2016-01-07 08:33:19.670	Certificate that is treated with payment for different abilities	an
4	47791	accept the request	2016-01-11 12:42:05.773	Certificate that is treated with payment for different abilities	—
5	47791	Send for treatment	2016-01-11 12:43:58.143	Certificate that is treated with payment for different abilities	—
6	47798	accept the request	2016-01-11 14:03:43.687	Certificate that is treated with payment for different abilities	evi
7	47798	Send for treatment	2016-01-11 14:05:57.427	Certificate that is treated with payment for different abilities	evi
8	47825	accept the request	2016-01-12 10:07:25.470	Certificate that is treated with payment for different abilities	an
9	47825	Send for treatment	2016-01-12 10:08:53.323	Certificate that is treated with payment for different abilities	an
10	47841	For Treatment	2016-01-12 11:56:24.727	Request for information	—
11	47848	accept the request	2016-01-12 13:26:18.757	Certificate that is treated with payment for different abilities	an
12	47848	Send for treatment	2016-01-12 13:27:45.673	Certificate that is treated with payment for different abilities	an
13	47850	accept the request	2016-01-12 13:49:41.587	Certificate that is treated with payment for different abilities	an
14	47850	Send for treatment	2016-01-12 13:50:41.627	Certificate that is treated with payment for different abilities	an
15	47873	The citizen withdrew the certificate	2016-01-13 13:24:18.083	Certificate for treatment with economic assistance	err
16	47898	accept the request	2016-01-13 13:29:31.277	Certificate that is treated with payment for different abilities	an
17	47898	Send for treatment	2016-01-13 13:30:48.720	Certificate of being treated with respect for different abilities	an

Figure 3. Event Logs – Source: Extracted from the SQL database, with content translated into English using Google Translate.

Using PM4Py for Analyzing Process Sequences

During this study regarding the analysis of process flows, the PM4Py library in Python was used, which has helped us a lot in the wide range of algorithms that it has, especially in the discovery and control of performance analysis (Berti et al., 2023).

This approach includes several stages: selection of appropriate algorithms, generation of visual models, performance analysis and determination of future research steps. After analyzing all these algorithms that we had available for process analysis, four of these elements were selected which offer the most accuracy and stability in the generated models.

1. Alpha Miner serves as the basic algorithm for discovering process models from event logs, identifying sequence, parallelism (AND) and conditional branching (XOR) relationships between activities (van der Aalst et al., 2004).
2. Alpha++ Miner is an advanced version that addresses the challenges of incomplete logs or rare actions, building more robust and closer to reality models
3. Heuristic Miner uses statistical analysis on frequencies and probabilities to create a Heuristic Net, eliminating irrelevant connections and focusing on the strongest flows;
4. Petri Net that enables detailed analysis of delays, repetitions and interruptions
5. Finally, Inductive Miner provides manageable and well-structured models, guaranteeing complete soundness and being particularly effective for complex logs (Leemans et al., 2013).

To perform the analysis, we used Python language, first importing the Pandas library for reading and manipulating CSV files, where the data exported from the database was stored. Then, PM4Py was imported, a specialized process mining library that provides support for standard event log formats and process models, as well as functions for their visualization. Using these tools, the event logs necessary for analyzing each process independently were created, enabling the application of workflow discovery and visualization algorithms.

```
def import_and_read_csv(file_path, limit=None):
    event_log = pandas.read_csv(file_path, sep=',')
    if limit:
        event_log = event_log[:limit + 1]

    return refactor_event_log_for_nets(event_log)

if __name__ == "__main__":
    event_log = import_and_read_csv("Csvs/ResultFinalStatus100k.csv", 1000)
```

Figure 4. Reading the 100.000 records from cvs files

In the context of process mining, an event log always contains a set of core elements that enable the construction and analysis of process models. First, a case identifier, which links each event to a specific execution instance of the process. Second, an activity, which describes the name of the step or action performed in the process. Third, a timestamp, which records the exact moment when the event occurred, allowing the analysis of the order and duration of activities. Formally, an event log □ L can be defined as a set of traces □ = { □ □ ∣ □ ∈ □ } L={σ c ∣ c∈C}, where each event appears at most once in the entire log, i.e., ∀ □ 1 , □ 2 ∈ □ , □ 1 ≠ □ 2 : □ □ □ (□ 1) ∩ □ □ □ (□ 2) = ∅ ∀σ 1 ,σ 2 ∈L,σ 1 =σ 2 :set(σ 1)∩set(σ 2)=∅ . Using the corresponding Python code with PM4Py, the event logs were created from the exported data, storing these three key elements for each event, to be later used in the process discovery and visualization phase.

```
def refactor_event_log_for_nets(event_log):
    event_log = event_log.rename(columns={
        'CaseId': 'case:concept:name',
        'Activity': 'concept:name',
        'Timestamp': 'time:timestamp',
        'NameE': 'Emertimi'
    })

    event_log['case:concept:name'] = event_log['case:concept:name'].astype(str)
    event_log['Emertimi'] = event_log['Emertimi'].astype(str)
    event_log['concept:name'] = [activity.capitalize() for activity in event_log['concept:name']]
    event_log['time:timestamp'] = pandas.to_datetime(event_log['time:timestamp'], format='%Y-%m-%d %H:%M:%S.%f', errors='coerce')
```

Figure 5. Create Events logs

After explaining the event log structure, the analysis proceeded to the Process Discovery phase, whose primary goal is to identify a suitable process model that accurately represents the order of events and activities executed during a process instance. Various discovery algorithms were applied, including Alpha Miner, Alpha+, Heuristic Miner, and Inductive Miner. The output of the Heuristic Miner is a Heuristics Net, an object that contains the set of activities along with the relationships and dependencies between them. From this Heuristics Net, it is possible to derive a Petri Net representation, which provides a more formal and analyzable model of the process flow. Using PM4Py, the Heuristic Miner algorithm was implemented in Python to automatically discover these relationships from the event logs and generate both the

Heuristics Net and the corresponding Petri Net for further performance and conformance analysis.

```
def generate_and_view_heuristic_net(event_log, view_net=False):
    heu_net = heuristics_miner.apply_heu(event_log)
    gviz = hn_visualizer.apply(heu_net)
    hn_visualizer.save(gviz, "Htmls/heuristic_net.png")
    print("Successfully generated Htmls/heuristic_net.png!")

    if view_net:
        hn_visualizer.view(gviz)
```

Figure 6. Code for Heuristic Net

The resulting Heuristic Net is presented in Figure 6. As observed, the model is cleaner and more comprehensible, focusing on the most relevant process flows and filtering out infrequent or insignificant paths.

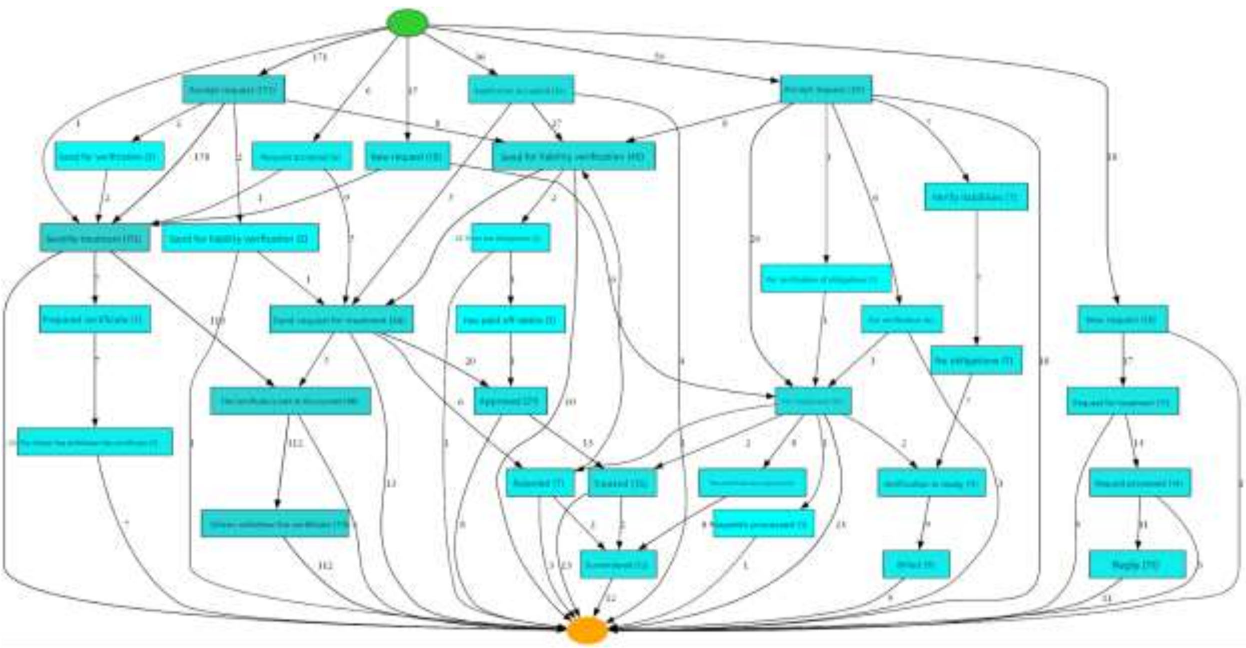

Figure 7. Heuristic graphs

After generating the Heuristics Net, the process was further transformed into a Petri Net representation using PM4Py. This step aimed to compare the two models Heuristics Net and Petri Net, in terms of their effectiveness for analyzing process sequences and identifying workflow patterns.

The Petri Net model consists of places, transitions, and arcs, and is accompanied by two key components: the initial marking and the final marking. The initial marking represents the starting state of the process execution, indicating the conditions that must be satisfied for the first activities to begin. The final marking represents the desired state that should be reached upon the completion of the process, signifying that all required activities have been executed and the process has successfully ended. By obtaining both the Petri Net and its corresponding initial and final markings, it becomes possible to simulate, analyze, and verify the correctness of the process flow, as well as to perform conformance and performance checks against the real event log.

```
def generate_and_view_petri_net(event_log, view_net=False):
    petri_net, initial_marking, final_marking = pm4py.discover_petri_net_alpha(event_log)
    pm4py.view_petri_net(petri_net, initial_marking, final_marking)
```

Figure 8. Petri Net code

Below is the Petri Net graph generated from the municipality’s process data, providing a detailed visualization of the workflow structure.

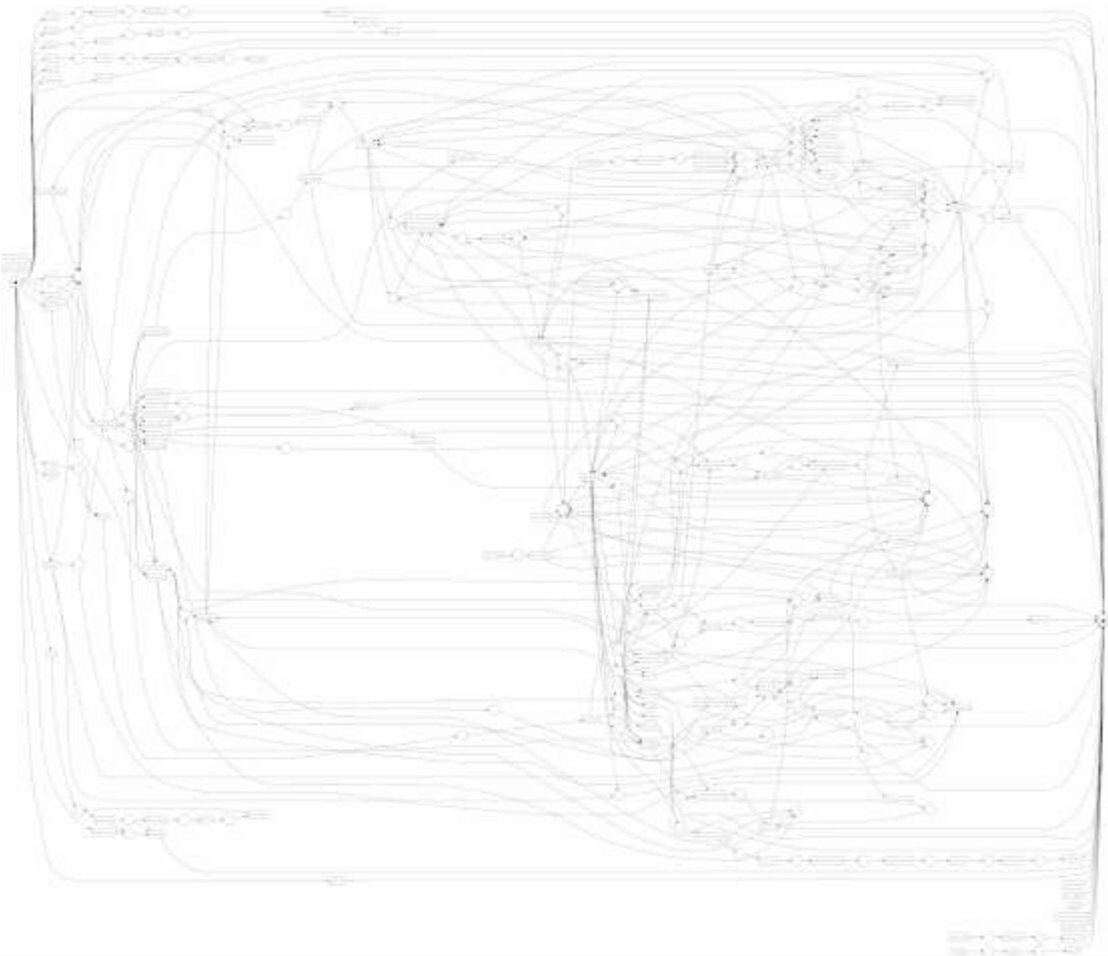

Figura. 9 Petri net Graph

COMPARATIVE VISUALIZATION OF PROCESS MODELS: PETRI NET VS. HEURISTIC NET

When comparing the Petri Net and the Heuristic Net generated from the same event log, it becomes evident that the Petri Net, while formally precise and sound, appears significantly more complex and visually cluttered. This complexity is due to the representation of every possible transition, token flow, and marking, which can make the model harder to interpret for non-technical stakeholders. In contrast, the Heuristic Net filters out infrequent paths and focuses on the most dominant flows, resulting in a cleaner and more comprehensible model. This makes it more practical for managerial decision-making, especially when the goal is to quickly identify bottlenecks and inefficiencies without navigating through an overly detailed network.

The Petri Net model offers a complete and formally sound representation of the process, capturing all possible transitions and states, but results in a highly complex and visually dense structure. In contrast, the Heuristic Net model focuses on the most frequent and relevant paths, filtering out infrequent behavior and producing a cleaner, more interpretable process map suitable for managerial analysis. To facilitate the interpretation of results and provide an intuitive exploration of the discovered processes, we developed an interactive HTML interface. This interface integrates the process discovery outputs generated through PM4Py, including the Heuristics Net and the corresponding Petri Net visualizations. The user can view the Log Summary, which displays the total number of cases and steps, alongside detailed case-level information such as the sequence of activities, execution times, and frequency of occurrence. The interface provides a graphical process map, enabling users to trace the most frequent paths, identify loops, and detect potential bottlenecks directly from the visualization. This approach not only supports detailed analysis but also enhances transparency by presenting the workflow data in a user-friendly and accessible format.

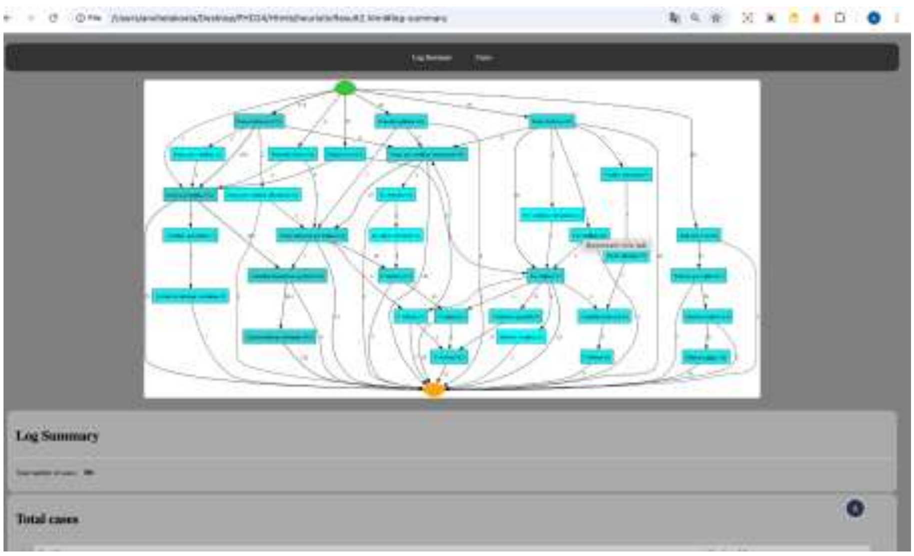

Figure 10. User interface view of the process mining system generated from municipal data

As we can see the interface provides an interactive visualization of the discovered processes, enabling users to explore event logs, process maps, and performance metrics derived from real municipal service data. Through these insights, it is possible to analyze every aspect of a process, verifying whether the path followed by a specific case is optimal or contains unnecessary steps. The interface also allows administrators to identify cases handled by personnel whose performance is below standard or whose actions suggest possible misconduct, enabling corrective measures. Furthermore, it provides statistics on how often each service is requested by citizens and how the administration handles these requests, supporting both process optimization and transparency initiatives.

RESULTS FROM ANALYSING SEQUENCES OF PROCESSING IN MUNICIPALITY

The analysis of process sequences included the calculation of cycle time and waiting time using interval event logs containing both start and end timestamps. This allowed for an incremental computation of lead time and cycle time for each event, with the final values, recorded on the last event of a case, reflecting the total execution time of the process. Through this approach, it was possible to identify specific activities responsible for delays and to determine where bottlenecks occurred in the workflow. For example, in figure 11 the same case was handled by the same person on one occasion for a duration of five hours, while in another instance it remained open for only one hour. Such discrepancies raise questions about process consistency and the factors influencing these variations.

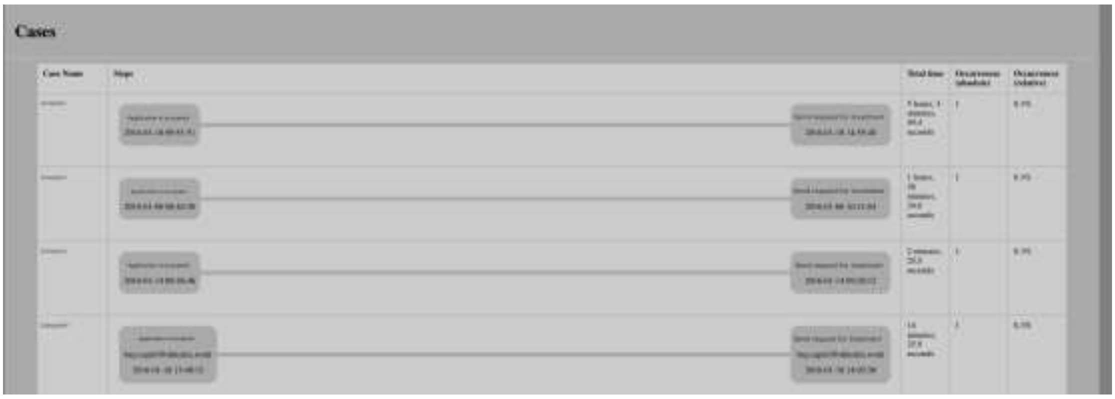

Figure 11. Detailed event log view showing case steps, total processing time, and occurrence frequency for case Complaints

The results also revealed cases where the same request was handled by different human resources at different times, raising questions about workload distribution, consistency of case management, and possible interruptions in the process flow. In some instances, identical cases showed significant variation in total processing time, suggesting inefficiencies and a lack of standardization in execution. For example in figure 12, one case took four days and five hours to complete, while another identical case was finalized in just one hour. Such extreme differences prompt further investigation into potential causes, including uneven task allocation, procedural bottlenecks, or inconsistencies in how cases are prioritized and processed.

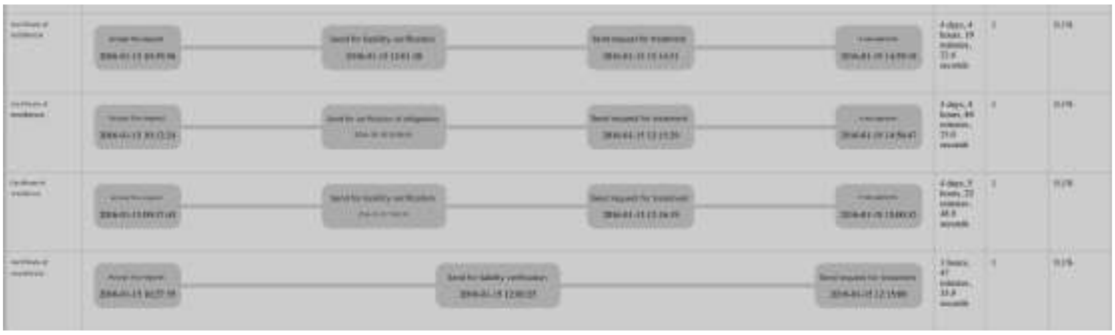

Figure 12. Detailed event log view showing case steps, total processing time, and occurrence frequency for case Certificate of residence

Proposed Solution

If organizations have real-time visibility into running process instances, they can take timely and targeted actions to counteract undesired situations before they impact service delivery. In the context of municipal administration, this means detecting delays, bottlenecks, or workload imbalances as they occur, allowing managers to redistribute tasks, escalate cases, or provide additional resources where necessary. Such proactive intervention not only prevents operational inefficiencies and potential losses but also improves citizen satisfaction by ensuring that requests are handled consistently and transparently. Integrating real-time monitoring with process mining enables a shift from reactive to preventive management, transforming workflow optimization from a one-off analytical exercise into a continuous improvement cycle.

FUTURE WORKS AND RECOMMENDATIONS

This study demonstrated the potential of process mining, particularly using the PM4Py framework, for analyzing and optimizing administrative workflows in Albanian municipalities. Building on the insights obtained from the process mining analysis, future work will focus on enhancing the detection of inefficiencies and irregularities through the integration of Anomaly Detection, Social Network Analysis (SNA), and Clustering techniques. Anomaly Detection will be implemented to automatically identify cases and activities that deviate significantly from expected behavior. This will be achieved using the Scikit-learn library in Python, specifically the Local Outlier Factor method, which computes a score reflecting the degree of anomaly for each observation. The algorithm detects instances with substantially lower density compared to their neighbors, thereby highlighting unusual processing times, atypical task sequences, or irregular resource allocations. Social Network Analysis (SNA) will be applied to uncover patterns of collaboration, task handovers, and interaction among employees involved in process execution. Key SNA metrics to be computed include:

- Handover of Work – measuring how often an individual's work is followed by another person's activity in the process.

- Subcontracting – counting how often a task is interrupted by another individual before returning to the original person.
- Working Together – assessing the frequency of two individuals working collaboratively on the same process instance.
- Similar Activities – measuring the similarity of work patterns between two individuals based on shared task execution.

Clustering will then be used to group individuals based on SNA metrics. For example, employees who frequently work together will be grouped according to the Working Together metric, while those handling similar tasks will be grouped under the Similar Activities metric. Such clustering can help in identifying functional teams, detecting overload risks, and streamlining task distribution.By combining process mining with anomaly detection, SNA, and clustering, the proposed solution aims to create a proactive, AI-assisted workflow optimization framework. This will allow municipal administrations to move from static process analysis to continuous monitoring and improvement, enabling timely interventions that enhance transparency, reduce delays, and improve citizen satisfaction.

This paper is recommended for municipalities in Albania and other developing regions, adopting an AI-augmented process mining framework can significantly improve transparency, reduce bureaucracy, and increase citizen trust. Future projects should focus not only on technology adoption but also on change management, capacity building, and policy alignment to sustain improvements over time.

CONCLUSION

This paper presented a practical application of process mining techniques to analyze and improve administrative workflows in a municipality in Albania. Using real operational data stored in a SQL-based database, we generated event logs and applied PM4Py algorithms: Alpha Miner, Alpha++, Heuristic Miner and Inductive Miner, to discover and visualize process patterns. By comparing Heuristic Nets and Petri Nets, we were able to map the current workflow execution, identify bottlenecks, detect repetitive or unnecessary steps and highlight areas prone to delays or interruptions. The results provide a clear and data-driven view of how municipal services are processed in practice, providing actionable insights for process optimization and better resource allocation. While the current work focused on historical analysis of process data, future research will integrate Artificial Intelligence methods, such as anomaly detection and Social Network Analysis, to enable real-time monitoring, predictive analytics, and proactive workflow improvements. The methodology and findings from this study can serve as a reference for other municipalities in Albania and similar developing contexts seeking to increase transparency, reduce inefficiencies, and improve service delivery through process mining.

ACKNOWLEDGEMENT

The authors would like to express their gratitude to the municipal administration staff who provided access to workflow data and participated in interviews during the service categorization phase. Special thanks go to the Information Office (Zi1N) for facilitating the data collection process and to the Faculty of Natural Sciences, University of Tirana, for academic guidance and institutional support throughout the research.

LIMITATION

While this study delivers valuable insights into the application of process mining in public administration, several limitations should be acknowledged. First, the analysis was based on data from a single municipality, which may limit the generalizability of the findings to other local administrations. Second, the accuracy and completeness of the event logs depended on the quality of the source data, meaning that missing or inconsistent records could have influenced the results. Third, the study focused solely on historical data analysis, without implementing real-time monitoring or automated interventions. Finally, although AI techniques such as anomaly detection and Social Network Analysis are proposed for future work, their practical impact remains to be validated in operational municipal environments.

REFERENCES

van der Aalst, W.M.P. (2012) Process mining: Overview and opportunities, ACM Transactions on Management Information Systems (TMIS), 3(2), pp. 1–17.

van der Aalst, W.M.P., Weijters, A.J.M.M. and Maruster, L. (2004) Workflow mining: Discovering process models from event logs, IEEE Transactions on Knowledge and Data Engineering, 16(9), pp. 1128–1142.

van der Aalst, W.M.P., Adriansyah, A. and van Dongen, B.F. (2012) Process mining manifesto, In: Daniel, F., Barkaoui, K. and Dustdar, S. (eds.) Business Process Management Workshops. Springer, pp. 169–194.

van der Aalst, W.M.P., Reijers, H.A. and Song, M. (2007) Business process mining: An industrial application, Information Systems, 32(5), pp. 713–732.

Berti, A., van Zelst, S. and Schuster, D. (2023) PM4Py: A process mining library for Python, Software Impacts, 16, 100556.

Çelik, U. and Akçetin, E. (2018) Process mining tools comparison, AJIT-e: Online Academic Journal of Information Technology, 9(34), pp. 97–114.

Palangsantikul, P., Kungcharoen, K., Narksarp, W. and Premchaiswadi, W. (2024) Process mining techniques utilizing ChatGPT: A comparative analysis with Disco software on call center log data, Proceedings of the 22nd International Conference on ICT and Knowledge Engineering (ICT&KE), Bangkok, Thailand, pp. 1–7.

Badakhshan, P., Geyer-Klingeberg, J., El-Halaby, M., Lutzeyer, T. and Affonseca, G.V.L. (2020) *Celonis Process Repository: A bridge between Business Process Management and Process Mining*, in Proceedings of the International Workshop on Business Process Management (BPM), Munich, Germany.

van der Aalst, W.M.P., van Dongen, B.F., Günther, C.W., Rozinat, A., Verbeek, H.M.W. and Weijters, A.J.M.M. (2009) *ProM: The process mining toolkit*, in Proceedings of the International Conference on Business Process Management (BPM), Ulm, Germany.

Mans, R.S., Schonenberg, H., Song, M. and van der Aalst, W.M.P. (2008) *Application of process mining in healthcare – A case study in a Dutch hospital*, in Fred, A.L.N., Filipe, J. and Gamboa, H. (eds.) *Biomedical Engineering Systems and Technologies: International Joint Conference, BIOSTEC 2008, Funchal, Madeira, Portugal, January*

28–31, 2008, Revised Selected Papers. Communications in Computer and Information Science, vol. 25. Berlin: Springer, pp. 425–438.

Li, Y., & Deng, S. L., Research on automatic government process remodeling in E-government. In Proceedings - International Conference on Management and

Măruşter, L., & Van Beest, N. R. T. P. , Redesigning business processes: A methodology based on simulation and process mining techniques. In Knowledge and Information Systems (Vol. 21, Issue 3, pp. 267–297, 2009).

Rawiro, D., Gaol, F.L., Supangkat, S. and Ranti, B. (2022) *Process mining applications in government sector: A systematic literature review*. Proceedings of the 5th European International Conference on Industrial Engineering and Operations Management, Rome, Italy, July 26–28, 2022.

Bozkaya, M., Gabriels, J., & Van Der Werf, J. M. , Process diagnostics: A method based on process mining. Proceedings - International Conference on Information, Process, and Knowledge Management, EKNOW 2009, 1, 22–27,2009.

Kang, Y. S., Lee, K., Lee, Y. H., & Chung, K. Y., RFID-based supply chain process mining for imported beef. Korean Journal for Food Science of Animal Resources, 33(4), 463–473, 2013.

Lamine, E., Fontanili, F., Mascolo, M. Di, & Pingaud, H. (2015). Improving the management of an emergency call service by combining process mining and discrete event simulation approaches. In IFIP Advances in Information and Communication Technology (Vol. 463, pp. 535–546). https://doi.org/10.1007/978-3-319- 24141-8_50

Bentley, M. J., Lin, A. C., & Hodson, D. D., Overcoming challenges to air force satellite ground control automation. 2017 IEEE Conference on Cognitive and Computational Aspects of Situation Management, CogSIMA 2017, 1–7, 2017.

Haq, R., Kapp, J. M., Schlemper, S., & Simoes, E. J., Using Process Mining to Assess the Fidelity of a Home Visiting Program. Frontiers in Public Health Services and Systems Research, 5(4), 5–11, 2016.

Shrivastava, S., & Pal, S. N. (2017). A big data analytics framework for enterprise service ecosystems in an eGovernance scenario. In ACM International Conference Proceeding Series: Vol. Part F1280 (pp. 5–11).

Molnár, B. (2017). Proposal for application of data science methods in E-Government: A Case-study about the application of available techniques for performance measurement with the help of data science. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 10441 LNCS, 143–157.

Santoro, F. M., Revoredo, K. C., Costa, R. M. M., & Barboza, T. M., Process Mining Techniques in Internal Auditing: A Stepwise Case Study. ISys - Brazilian Journal of Information Systems, 13(4), 48–76, 2020.

Larsson, K. K., Digitization or equality: When government automation covers some, but not all citizens. Government Information Quarterly, 38(1), 101547,2021. Berti, A., van Zelst, S.J. and Schuster, D. (2023) PM4Py: A process mining library for Python, Software Impacts, 17, 100556.

Jessen, U., Sroka, M. and Fahland, D. (2023) Chit-chat or deep talk: Prompt engineering for process mining.

Zeng, Q., Wang, M., Li, M. and Zhao, Y. (2011) Process mining-based workflow model fragmentation for distributed execution

Tax, N., Verenich, I., La Rosa, M. and Dumas, M. (2017) Predictive business process monitoring with LSTM neural networks, Data & Knowledge Engineering

Teinemaa, I., Dumas, M., La Rosa, M. and Maggi, F.M. (2021) Outcome-oriented predictive process monitoring: Review and benchmark, ACM Transactions on Knowledge Discovery from Data (TKDD), 15(4), pp. 1–42.

Leemans, S. J. J., Fahland, D., & van der Aalst, W. M. P. (2013). Discovering block-structured process models from event logs – A constructive approach. Application and Theory of Petri Nets and Concurrency, 311–329. Springer.