DISSERTATIONES
INFORMATICAE
UNIVERSITATIS
TARTUENSIS
69

KATERYNA KUBRAK

Towards User-Centered Prescriptive Process Monitoring Systems

DISSERTATIONES INFORMATICAE UNIVERSITATIS TARTUENSIS

69

DISSERTATIONES INFORMATICAE UNIVERSITATIS TARTUENSIS

69

KATERYNA KUBRAK

Towards User-Centered Prescriptive Process Monitoring Systems

UNIVERSITY OF TARTU
Press

Institute of Computer Science, Faculty of Science and Technology, University of Tartu, Estonia.

Dissertation has been accepted for the commencement of the degree of Doctor of Philosophy (PhD) in Computer Science on May 6, 2025 by the Council of the Institute of Computer Science, University of Tartu.

Supervisors

Assoc. Prof.	Fredrik Payman Milani University of Tartu, Estonia
Assist. Prof.	Alexander Udo Nolte Eindhoven University of Technology, The Netherlands Carnegie Mellon University, USA
Prof.	Marlon Gerardo Dumas Menjivar University of Tartu, Estonia

Opponents

Prof.	Pnina Soffer University of Haifa, Israel
Assoc. Prof.	Adela del Río Ortega University of Seville, Spain

The public defense will take place on June 13, 2025 at 14:15 in Narva Rd. 18-1020.

The publication of this dissertation was financed by the Institute of Computer Science, University of Tartu.

ISSN 2613-5906 (print) ISSN 2806-2345 (pdf)
ISBN 978-9916-27-893-2 (print) ISBN 978-9916-27-894-9 (pdf)

University of Tartu Press
http://www.tyk.ee/

In memory of my grandfather Hryhorii (1939-2020)
Пам‘яті мого дідуся Григорія (1939-2020)

ABSTRACT

A prescriptive process monitoring (PrPM) system is an information system that recommends actions during the execution of instances of a business process with the aim of improving the performance of the process. Existing techniques for developing PrPM systems differ in the performance objectives they optimize for, the recommendations they generate, and the algorithms they use to determine which recommendations to make and when.

Most existing research on PrPM systems focuses on the efficiency and accuracy of the algorithms used to generate recommendations. In contrast, less attention has been given to the usability of these systems. This gap limits the applicability of PrPM systems in practice. This thesis addresses this gap by studying how users interact with the recommendations produced by PrPM systems.

Specifically, this thesis aims to answer four research questions. (1) What are the outputs of PrPM techniques? (2) What user groups could benefit from working with the outputs of PrPM techniques? What are the users' information needs? (3) How to translate these information needs into a user interface design for PrPM systems? (4) How to enhance the understandability of PrPM outputs?

To answer these questions, first, the thesis provides a comprehensive overview of existing PrPM techniques and their outputs, leading to a conceptual framework. This framework categorizes PrPM techniques based on their objectives, types of recommendations, required data inputs, the policies they follow, and how the goodness of the PrPM technique is evaluated. These categories help to understand the range of PrPM outputs that can be presented to end users.

Second, the thesis tackles the question of how an interface for PrPM can be designed. Based on an analysis of the outputs of PrPM techniques and a series of interviews with potential users of PrPM systems, we map the information needs of three distinct user groups (process analysts, operational workers, and operational managers). Based on this mapping, we design and evaluate a PrPM tool, namely Kairos. Kairos is a web-based tool that provides recommendations to perform actions during the execution of ongoing cases of a process based on event logs of business processes. As a result of this user study, we derive suggestions for designing user interfaces for PrPM systems.

Third, to assist users in understanding the outputs of PrPM techniques, the thesis proposes a prompting method for a Large Language Model (LLM) to enhance the explainability of PrPM outputs. The prompting method is implemented in Kairos and evaluated with end users, leading to insights into the potential benefits and challenges of designing LLM-based user interfaces for PrPM systems.

In conclusion, this thesis contributes to the design of user-centered PrPM systems by ensuring recommendations are not only technically robust but also usable and actionable for end users. Its findings advance both the theoretical understanding and practical implementation of PrPM systems.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

Acronyms

AI Artificial Intelligence.

BPM Business Process Management.

DSR Design Science Research.
DSRM Design Science Research Methodology.

HCAI Human-Centered AI.

KPI Key Performance Indicator.

LLM Large Language Model.

ML Machine Learning.

PPM Predictive Process Monitoring.
PrPM Prescriptive Process Monitoring.

SLR Systematic Literature Review.

UCD User-Centered Design.
UI User Interface.

XAI Explainable AI.
XAIQB eXplainable AI Question Bank.

LIST OF ORIGINAL PUBLICATIONS

Publications in scope of the thesis

I **Kubrak, K.**, Milani, F., Nolte, A., and Dumas, M. (2022). Prescriptive process monitoring: Quo vadis? In: *PeerJ Computer Science 8:e1097.* DOI: `10.7717/PEERJ-CS.1097`
Author's contribution: Lead author. Conceptualization, implementation, analysis, writing.

II **Kubrak, K.**, Milani, F., Nolte, A., and Dumas, M. (2023). Design and Evaluation of a User Interface Concept for Prescriptive Process Monitoring. Advanced Information Systems Engineering. CAiSE 2023, Zaragoza, Spain. In: *Lecture Notes in Computer Science, vol 13901*, pp. 347–363. Springer, Cham. DOI: `10.1007/978-3-031-34560-9_21`
Author's contribution: Lead author. Conceptualization, implementation, evaluation, analysis, writing, conference presentation.

III **Kubrak, K.**, Botchorishvili, L., Milani, F., Dumas, M., Nolte, A., Shoush, M., and Qu, Z. (2023). Kairos: A Tool for Prescriptive Monitoring of Business Processes. Doctoral Consortium and Demo Track 2023 at the 5th International Conference on Process Mining. ICPM 2023, Rome, Italy. In: *CEUR Workshop Proceedings, vol. 3648.* `https://ceur-ws.org/Vol-3648/paper_214.pdf`
Best demo award of ICPM 2023.
Author's contribution: Lead author. Conceptualization, evaluation, analysis, writing, conference presentation.

IV **Kubrak, K.**, Botchorishvili, L., Milani, F., Nolte, A., and Dumas, M. (2024). Explanatory Capabilities of Large Language Models in Prescriptive Process Monitoring. Business Process Management. BPM 2024, Krakow, Poland. In: *Lecture Notes in Computer Science, vol 14940*, pp. 403-420. Springer, Cham. DOI: `10.1007/978-3-031-70396-6_23`
Best paper award of BPM 2024.
Author's contribution: Lead author. Conceptualization, evaluation, analysis, writing, conference presentation.

V **Kubrak, K.**, Milani, F., Nolte, A., Botchorishvili, L., Dumas, M., Shoush, M. *(Under review)* Supporting Analysts and Managers to Utilize Prescriptive Process Monitoring: A User Interface Design and Evaluation. *ACM Transactions on Management Information Systems.*
Author's contribution: Lead author. Conceptualization, evaluation, analysis, writing.

Other published work of the author

VI **Kubrak, K.**, Milani, F., and Nolte, A. (2022). Process Mining for Process Improvement - An Evaluation of Analysis Practices. Research Challenges in Information Science. RCIS 2022, Barcelona, Spain. In: *Lecture Notes in Business Information Processing, vol 446*, pp. 214-230. Springer, Cham. DOI: `10.1007/978-3-031-05760-1_13`

VII **Kubrak, K.**, Milani, F., and Nolte, A. (2023). A visual approach to support process analysts in working with process improvement opportunities. In: *Business Process Management Journal, Vol. 29 No. 8*, pp. 101-132. DOI: `10.1108/BPMJ-10-2021-0631`

VIII **Kubrak, K.**, Milani, F., and Nava, J. (2023). Digital Technology-Driven Business Process Redesign: A Classification Framework. International Conference on Research Challenges in Information Science. RCIS 2023, Corfu, Greece. In: *Lecture Notes in Business Information Processing, vol 476*, pp. 205-221. Springer, Cham. DOI: `10.1007/978-3-031-33080-3_13`

IX Leemets, J.-L., **Kubrak, K.**, Milani, F., Nolte, A. (2023). Persuasive Visual Presentation of Prescriptive Business Processes. International Conference on Research Challenges in Information Science. RCIS 2023, Corfu, Greece. In: *Lecture Notes in Business Information Processing, vol 476*, pp. 398-414. Springer, Cham. DOI: `10.1007/978-3-031-33080-3_24`

X Milani, F., **Kubrak, K.**, Nava, J. (2024). Strategic redesign of business processes in the digital age: A framework. In: *Data & Knowledge Engineering, 102367.* DOI: `10.1016/J.DATAK.2024.102367`

1. INTRODUCTION

Organizations constantly seek to improve their business processes to increase efficiency and outperform competitors [32]. A business process is a set of activities that, when executed in a specific order, delivers a service or a product to customers [32]. In the past, process analysts identified improvement opportunities by manually modeling and analyzing processes. Over the past two decades, data-driven methods, such as process mining, have emerged as a viable alternative [98].

Process mining techniques use event logs extracted from information systems. Event logs capture the execution of a business process with data, such as case ID, timestamps, resources, and other case attributes [1]. Process mining incorporates techniques for different use cases that could be categorized as descriptive (e.g., process discovery, conformance checking, and performance measurement [1]) or predictive, which can predict the outcome of ongoing cases of a process [39]. Previous work highlights that to leverage the value of predictive techniques, prediction-based recommendations should be made to end-users [91].

As an extension of predictive use cases, Prescriptive Process Monitoring (PrPM) techniques have emerged. PrPM techniques provide recommendations for ongoing cases that can optimize the process performance with respect to one or more measures [66], such as success rate (percentage of cases that end in a positive outcome) [37], on-time completion rate [150], cycle time [12], or processing time [115]. In recent years, research has focused on improving techniques to become more accurate and efficient, as well as consider different aspects, such as type of recommendation (guiding [52], correlation-based [45, 16] or causality-based [15, 134]), and type of recommended action (resources to assign [131, 113] or an activity to perform [84, 16]).

However, comparatively little attention has been given to the presentation of the outputs of PrPM techniques. While some have developed a User Interface (UI) for their proposed technique (e.g., [155, 60, 5]), their focus is primarily on showcasing the technological capabilities, not on addressing end-user needs. This gap in research regarding user interaction with PrPM outputs presents a limitation for the practical implementation of these techniques.

The importance of addressing end-user needs in facilitating technology acceptance has been previously studied. Research on expert systems [156] and, more recently, on recommender systems [4] demonstrates the importance of considering end-user' needs. In the context of Artificial Intelligence (AI) systems, to which PrPM is also related [22], acceptance of outputs remains challenging due to, for instance, unsatisfactory user experiences with user-AI interfaces [63]. Specifically in PrPM systems, previous research has shown that end users struggle to understand and act on the prescribed recommendations [30].

To address these challenges, the concept of Human-Centered AI (HCAI) has emerged that proposes prioritizing end-users' needs when designing and developing AI systems [130]. HCAI is operationalized through three main approaches:

explainable AI, which focuses on making black-box AI models understandable; user-friendly AI, which emphasizes usability and usefulness; and responsible AI, which addresses trustworthiness and fairness [63, 28]. These approaches aim to improve user interactions with AI systems and, thereby, reduce the barriers to acceptance.

User-friendly AI can be seen as the starting approach to the other two. It helps the users to develop an understanding of the AI system's capabilities and enables effective communication channels [63, 140]. This knowledge aids users in assessing the system's reliability and credibility (explainable AI) and ultimately using the system responsibly (responsible AI) [63, 140]. User-friendly AI includes the concept of usability, which describes how effectively, efficiently, and satisfactorily users can achieve their goals with a system [139].

In summary, it is necessary to ensure the usability of PrPM outputs before they can be enhanced with explainability and responsibility features. Users need to effectively interact with outputs to maximize the benefits of these techniques in practice. However, a research gap remains on how to achieve this.

1.1. Problem Statement

To address this research gap, we formulate the research objective (RO) as *to provide end users with usable outputs of prescriptive process monitoring techniques.* This RO is divided into four research questions (RQs).

First, there is a variety and fragmentation of PrPM techniques. Existing PrPM techniques require different input data, target different objectives, and offer a variety of outputs. It is not clear which outputs are valuable to end users. Therefore, we review existing PrPM techniques and categorize the types of objectives they target, the data they require, and outputs they produce. With such a mapping, we can align the outputs of PrPM techniques with end-users needs. Therefore, the first research question is:

- **RQ**$_1$. *What are the outputs of PrPM techniques?*

Second, in this thesis, we aim to make the outputs of PrPM techniques usable. For that, we need to understand the potential users. While process analysts are the primary users of process mining techniques for process analysis, other stakeholders, such as managers, also benefit from such outputs [74, 157]. Research has shown that the end-users of PrPM techniques extend beyond process analysts. For instance, the outputs of PrPM have been used by claim processors [30] and physicians [94], showing their usefulness for other stakeholders such as managers and process workers. In light of this, we formulate the second research question:

- **RQ**$_2$. *What user groups could benefit from working with the outputs of PrPM techniques? What are the users' information needs?*

Third, the mapping of end-users' information needs enables the design and development of a user interface. Designing an interface involves analyzing the differences between the identified end-user groups and investigating how their needs can be addressed in a UI [26, 4]. Specifically, the design of the interface needs to be aligned with the needs of each user-group to ensure that the PrPM outputs are usable. Therefore, the third research question is:

- **RQ**$_3$. *How to translate these information needs into a user interface design for PrPM systems?*

Finally, the effectiveness of PrPM interfaces depends on the extent that recommendations are followed, which itself depends on the users' ability to understand what the recommendation means. Previous work has highlighted the challenges of providing understandable explanations of outputs to business users [122], leading users relying on their own judgment and ignoring the recommendations [30]. The fourth research question is formulated as:

- **RQ**$_4$. *How to enhance the understandability of PrPM outputs?*

1.2. Research Method

To achieve our research objective, we adopt the Design Science Research (DSR) approach, aimed at advancing human knowledge through the development of innovative artifacts [54]. There are different approaches to applying DSR, e.g., [65, 78, 117]. The most widely used approach in information systems research [145] is Design Science Research Methodology (DSRM) [117], which we also follow in this work.

DSRM consists of six phases: *(1) Problem identification and motivation, (2) Definition of the objectives for the solution, (3) Design and development, (4) Demonstration, (5) Evaluation,* and *(6) Communication* [117]. Figure 1 provides an overview of each of the phases mapped to the RQs of this thesis.

Phase 1: Problem Identification and Motivation. This phase corresponds to **RQ**$_1$. The objective of this phase is to define a specific problem and justify the value of its solution. According to DSRM, completing this phase requires knowledge of the problem's current state and an understanding of the importance of its solution [117]. To answer the research question, we conduct a Systematic Literature Review (SLR) [73] of existing PrPM techniques. The outputs of this phase are: (i) a structured review of the prescriptive process monitoring domain, (ii) a framework that classifies PrPM techniques, and (iii) identified research gaps and their implications.

Phase 2: Definition of the objectives for the solution. In this phase, the solution objectives are inferred from the problem definition and knowledge of what is possible and feasible. To do this, knowledge of the state of problems and current solutions, if any, are required [117]. First, to consolidate the knowledge of the state of the problem, we build on the domain review from *Phase 1*. Then, for the knowledge on current solutions, we conduct an analysis of existing PrPM tools. As a result, we elicit information items to be included in the UI for PrPM outputs.

The phases *(3) Design and development*, *(4) Demonstration*, and *(5) Evaluation* are conducted three times in separate development cycles.

Development Cycle I. This cycle corresponds to $\textbf{RQ}_2$.

Phase 3: Design and development. In this phase, an artifact is developed. This includes determining the artifact's desired functionality, its architecture, and creating an artifact [117]. To evaluate the usefulness and relevancy of the information items, we design a wireframe.

Phase 4: Demonstration and *Phase 5: Evaluation.* The purpose of these phases is to demonstrate the artifact in a specific context, and observe and measure how well the artifact supports a solution to the problem [117]. Here, we conduct an evaluation of the wireframe with end users.

The outputs of this cycle are: (i) a pre-design of the interface for PrPM outputs, (ii) a mapping of end-user groups that could benefit from working with PrPM outputs, and (iii) refined information items for each group.

Development Cycle II. This cycle corresponds to $\textbf{RQ}_3$.

Phase 3: Design and development. In this cycle, we developed an interface for PrPM outputs called Kairos. It incorporates functionality for the end-user groups determined in the previous phase. The functionality for each group is based on the respective information items. The PrPM techniques included in Kairos stem from the review done in phase 1.

Phase 4: Demonstration and *Phase 5: Evaluation.* Here, we conduct an evaluation of the interface with representatives of the identified end-user groups. The demonstration of the artifact is achieved through the users interacting with it using a specific event log as an input. The evaluation focuses on the usability and usefulness of the interface. The findings of the evaluation are summarized into a set of refinements for the interface. Based on the evaluation, a set of suggestions for designing PrPM interfaces is formulated.

The outputs of this cycle are (i) the open-access web-version of the interface (Kairos), (ii) a list of finalized information items that represent the end-user needs when working with the outputs of PrPM techniques, and (iii) a list of suggestions for designing PrPM interfaces.

Development Cycle III. This cycle corresponds to $\textbf{RQ}_4$.

Phase 3: Design and development. Following the results of the evaluation in Cycle II, we observed that one refinement that could be made in the interface is to improve the understandability of recommendations presented to the end users in the interface. Therefore, in this phase of Cycle III, we design an artifact: a prompting method that enables a Large Language Model (LLM) to elaborate on PrPM recommendations. The prompting method is implemented in a chatbot that is added as part of Kairos to enable its further evaluation.

Phase 4: Demonstration and *Phase 5: Evaluation.* In these phases, we conduct an evaluation of the developed artifact with the end users, where they can interact with the chat that utilizes the prompting method. The focus of the evaluation is on the outputs of the chat and the interaction between the chat and the user.

The outputs of this cycle are: (i) the prompting method, (ii) the LLM-based chat as an additional feature in the accessible web-version of Kairos, and analyses of (iii) the questions asked by the end users, (iv) answers given by the chat, and (v) user-chat interaction.

Phase 6: Communication. The last phase of DSRM incorporates communicating the problem, the developed artifacts and their novelty, as well as rigor of its design, and effectiveness to the end users [117]. The mapping of publications to the respective RQs and phases is demonstrated in Figure 1.

1.3. Contributions

This thesis provides the following contributions:

Contribution 1: A framework that classifies PrPM techniques along the following dimensions: objective, recommendation types, data input, policy, user evaluation aspect (Chapter 3). Our contribution aims at supporting researchers of PrPM techniques. They benefit from this contribution as they can gain insight into the current state of the art of the field and identify potential directions for future research. Developers of process mining tools who are interested in incorporating PrPM into their tools can also benefit from this work by better understanding the limitations and perspectives of existing methods.

Contribution 2: An open-access web-based tool for PrPM outputs (Kairos), an overview of end-user groups for PrPM with their information needs, and a set of suggestions for designing PrPM interfaces (Chapter 4, Chapter 5). These contributions have implications for end users and developers of process mining tools. Specifically, end users benefit from using Kairos to review the recommendations and find process improvement opportunities. Additionally, developers can follow the suggestions when developing PrPM interfaces.

Contribution 3: A prompting method to present explanations of recommendations in PrPM, and insights into the potential benefits and challenges of designing LLM-based systems for enhancing explainability in PrPM interfaces (Chapter 6). These contributions help the end users working with PrPM interfaces to understand the outputs shown in those better. Additionally, researchers can use the challenges described to design future research experiments.

1.4. Thesis Outline

The rest of this thesis is structured as follows. Chapter 2 introduces background. Chapter 3 corresponds to *Contribution 1* and thus describes the process and results of the SLR. Chapters 4 and 5 describe *Contribution 2*, with the former outlining the objectives for the solution, and the latter describing design and evaluation of the PrPM interface. Chapter 6 refers to *Contribution 3* and thus describes the design and evaluation of the prompting method to enhance the explanations of recommendations in PrPM. Finally, Chapter 7 concludes the work with outlining the answers to the RQs.[1]

Table 1 summarizes the RQs, contributions, respective chapters and publications. The respective chapters may contain sentences or fragments of sentences from the specified publications.

Table 1. Overview of research questions, contributions, and respective publications.

Research Question	Contribution	Publication
RQ$_1$ What are the outputs of PrPM techniques?	*Contribution 1* (Chapter 3): A framework that classifies PrPM techniques along dimensions of objective, recommendation types, data input, policy, user evaluation aspect.	`Publication I:` - PeerJ journal paper [66]

RQ$_2$ What user groups could benefit from working with the outputs of PrPM techniques? What are the users' information needs?
RQ$_3$ How to translate these information needs into a user interface design for PrPM systems?
	Contribution 2 (Chapter 4, Chapter 5): An open-access web-based tool for PrPM outputs Kairos, an overview of end-user groups and their information needs, and a set of suggestions for designing PrPM interfaces.	`Publication II:`
		- CAiSE'23 conf. paper [67]
		`Publication III:`
		- ICPM'23 demo paper [68]
		`Publication V:`
		- TMIS journal paper [70]

RQ$_4$ How to enhance the understandability of PrPM outputs?	*Contribution 3* (Chapter 6): A prompting method to present explanations of recommendations in PrPM, and insights into the potential benefits and challenges of designing LLM-based systems for enhancing explainability in PrPM interfaces.	`Publication IV:` - BPM'24 conf. paper [69]

[1] *Note on the usage of AI tools during PhD thesis writing.* I acknowledge the usage of ChatGPT (model 4o mini) to reformulate my thoughts or find synonyms for some words, and Grammarly to improve the grammatical accuracy of the text.

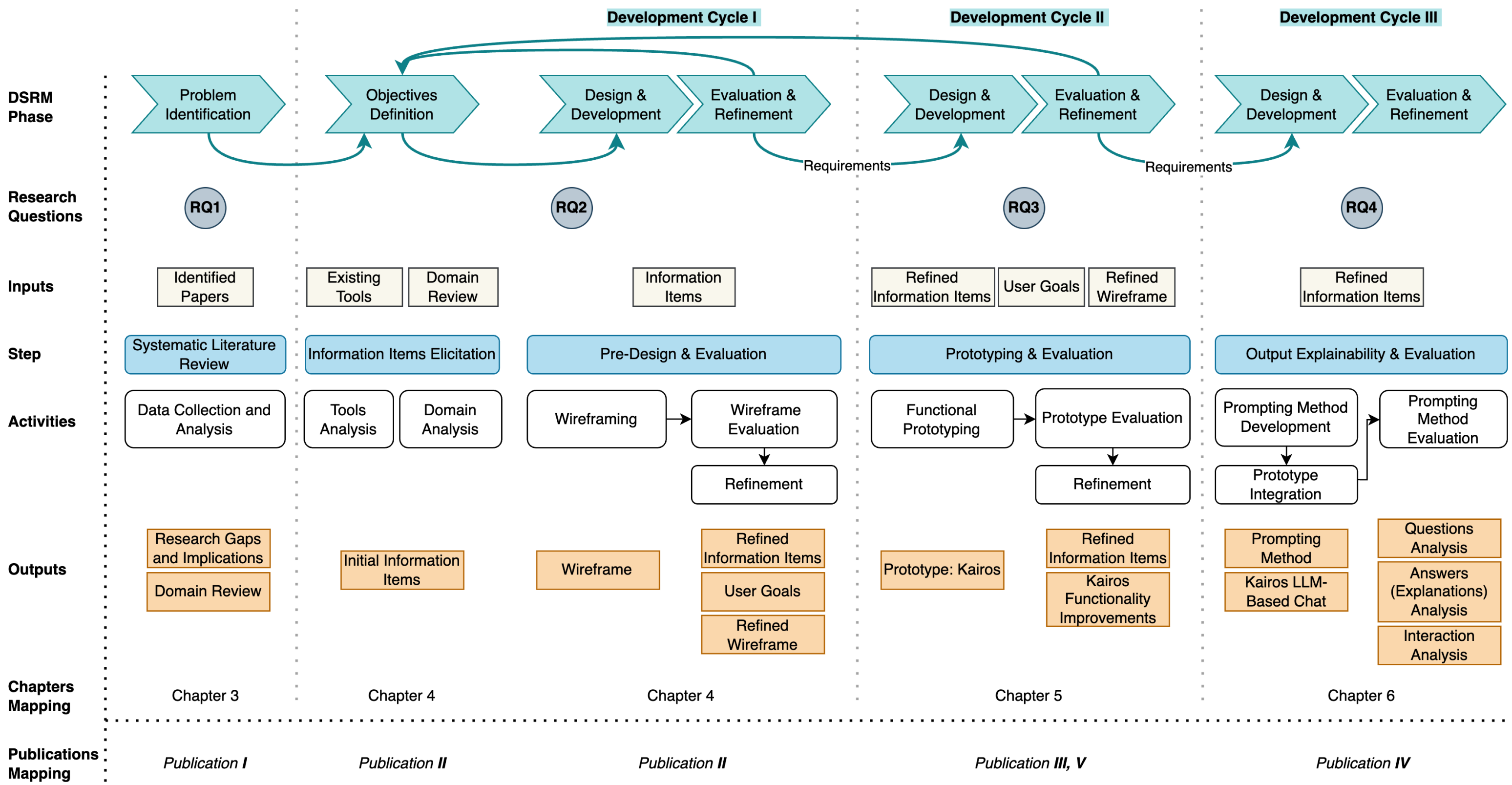

Figure 1. Mapping of the research method to RQs, research activities, thesis chapters, and publications. The phases are based on DSRM [117].

2. BACKGROUND

This chapter outlines the key concepts of the thesis. In Section 2.1, we introduce the fields of Business Process Management and Process Mining. Then, Section 2.2 gives an overview of Predictive and Prescriptive Process Monitoring. In Section 2.3, we introduce the concept of Human-Centered AI, and describe the current state of explainability, responsibility, and usability in Predictive and Prescriptive Process Monitoring. Finally, Section 2.4 describes the notions of user information needs and design principles for user interfaces.

2.1. Business Process Management and Process Mining

Business processes are defined as "the chain of activities, events, and decisions which are performed by the organization to deliver value to its customers." Essentially, a business process describes any set of activities that an organization carries out to provide value to its customers. Business processes can be found anywhere, for example, an order-to-cash process in an online shop, which begins with a customer order and ends with delivery and payment. Another example is an issue-to-resolution process in a customer service provider, which starts with a customer complaint and concludes with its resolution [32].

The design and implementation of an organization's business processes have a direct impact on customer satisfaction. Well-designed, customer-oriented processes enhance the quality of services delivered, while efficient internal processes improve the speed and cost-effectiveness of delivery [32]. Furthermore, high-performing customer-oriented and internal processes can provide a competitive edge, enabling an organization to outperform its peers [32]. Therefore, it is crucial for the organizations to continually seek opportunities to improve their processes to maintain competitiveness.

Business Process Management (BPM) is a body of principles, methods, and tools to discover, analyze, redesign, implement and monitor business processes [32]. These activities comprise the BPM lifecycle. Historically, the activities in the lifecycle were performed by process analysts manually. For instance, discovering and modeling a business process required extensive interviews and workshops with process stakeholders, followed by manual analysis of the process to identify improvement opportunities [32, 22]. Process mining techniques emerged as an alternative to help automate the BPM lifecycle activities based on the data available in information systems of the companies.

Process mining techniques discover data-driven models of business processes using event logs extracted from information systems [1]. An event log captures execution data, such as case ID, timestamp, activity, resources, and other contextual attributes of cases (process instances) of a process. For example, in a loan application process at a bank, application A (case ID) is verified (activity) by a senior specialist B (resource) on 16/01/24 at 09:02:45 (timestamp). In this example, contextual attributes may be purpose of a loan, number of offers, credit score, etc.

Process mining incorporates techniques for different use cases that could be categorized along the data-driven BPM pyramid [22] (Figure 2). Descriptive techniques include process discovery, conformance checking, and model enhancement. Process discovery aims at extracting process models from event logs [6]. Process discovery can be utilized to facilitate discussions among stakeholders, as a shared understanding of a process model helps to reach consensus. Additionally, it can support the generation of process improvement ideas, as the visual representation of the process and its associated issues enhances re-engineering efforts [1]. Conformance checking is used to check the alignment between the information extracted from the event logs and the process models existing in the organizations [33]. The goal of model enhancement is to improve the existing process model with information extracted from the event log [1]. For example, by looking into the resources performing different sets of activities, resource performance can be analyzed and assessed [97].

Predictive and prescriptive techniques are described in the next subsection. In essence, predictive techniques predict the outcome of ongoing cases of a process, and prescriptive recommend actions to avoid or mitigate the predicted negative outcome [39, 66]. The final layer of the pyramid, augmented process execution, extends beyond informing or recommending decisions to enhance business processes. The goal of augmented process execution is to autonomously manage and optimize processes to achieve desired business outcomes, while operating within the constraints and boundaries defined by managers [22].

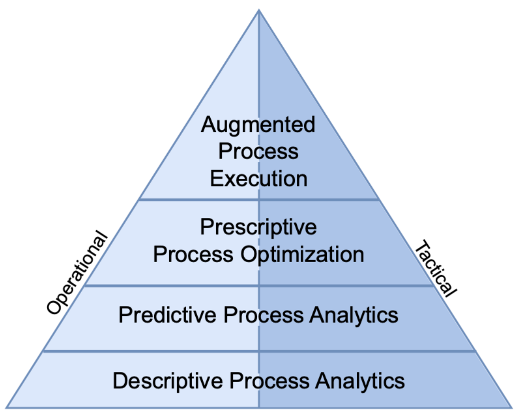

Figure 2. A pyramid categorization of data-driven BPM techniques [22].

As described by Röglinger et al. [123], BPM encompasses both radical change, such as process re-engineering, and incremental change, such as continuous process improvement. PrPM contributes to the latter by enabling organizations to continuously optimize process execution at runtime. This approach contrasts with redesign-based improvement methods by enabling end-users to intervene in ongoing cases, thus enabling a bottom-up mode of improvement.

2.2. Predictive and Prescriptive Process Monitoring

Predictive Process Monitoring (PPM) techniques predict the outcome of ongoing cases of a process. In this way, PPM is forward-looking as it provides predictions about the future, thus allowing to predict potential deviation before they occur and potentially take preventive measures [39, 40]. PPM techniques typically involve two main stages [40]. The first stage is the training phase, during which one or more models are created or enhanced using information from the execution log. The second stage is the runtime phase, where the trained model(s) are used to generate predictions for an ongoing case. PPM techniques can be broadly categorized into two groups: model-based approaches and Machine Learning (ML)-based approaches [40]. Model-based approaches rely on explicit models, such as annotated transition systems, whereas ML-based approaches use statistical and ML techniques, such as classification, regression, or neural networks. Instead of explicit models, they rely on implicit predictive models that are created by transforming event log information into feature representations, which serve as input for machine or deep learning algorithms [40].

PPM techniques can also be categorized from an output perspective, i.e., what they output and how that could be used in practice [39]. In this regard, di Francescomarino et al. [39] categorize PPM techniques across three main dimensions: numeric predictions (e.g., time and cost predictions), categorical predictions (e.g., risk and outcome predictions), and next-activity predictions. These categories describe the types of outcomes each technique predicts, such as remaining process time, compliance risks, or future process activities.

Prescriptive process monitoring techniques take one step further and provide recommendations on how to intervene in an ongoing case of a process to avoid or mitigate a negative outcome. As in PPM, first, a prediction is made using historical data from completed cases of the process, such as the remaining time until a case's completion or the expected outcomes of a case. Then, the PrPM system generates recommendations for specific actions (also referred to as treatments [14] or interventions [134]) based on a cost model that includes the costs and benefits of the actions.

While ML approaches are also prevalent in PrPM techniques, other prescriptive analytics methods or a combination thereof are used. A review on prescriptive analytics by Lepenioti et al. [85] categorizes prescriptive analytics methods into the following categories: machine learning/data mining (ML/DM), probabilistic models, mathematical programming, evolutionary computation, simulation, logic-based models. In this light, examples of applying ML/DM in PrPM exist (e.g., reinforcement learning [16, 133], neural networks [150, 115]), as well as mathematical programming (e.g., mixed integer programming [52]), logic-based models (e.g., criteria-based rules [48]).

From the output perspective, PrPM techniques can be categorized into two primary objectives: optimizing process outcomes (e.g., avoiding deadline violations or undesired events) and optimizing process efficiency (e.g., reducing cycle time or improving resource utilization). The recommendations provided by the techniques are typically aligned with control flow (e.g., recommending an activity) or resource perspectives (e.g., assigning tasks to specific resources) [66]. A detailed outline of PrPM techniques, following the categorization, is provided in Chapter 3.

Illustrative Example of PrPM

Returning to an example of a loan application at a bank, Figure 3 provides an illustration of applying PrPM techniques to it. In this illustrative example, an application was received and the specialist started processing it. The specialist first confirmed the application, then checked the credit score of the applicant, and proceeded with verifying the applicant's documents. Then, the specialist sent an offer. The customer has not yet responded to the offer. At this point in time, the PrPM system predicted that the customer is likely to reject the offer. Therefore it recommends to send an additional offer. In this specific case, the recommendation is aimed that optimizing the process outcome (offer accepted/rejected).

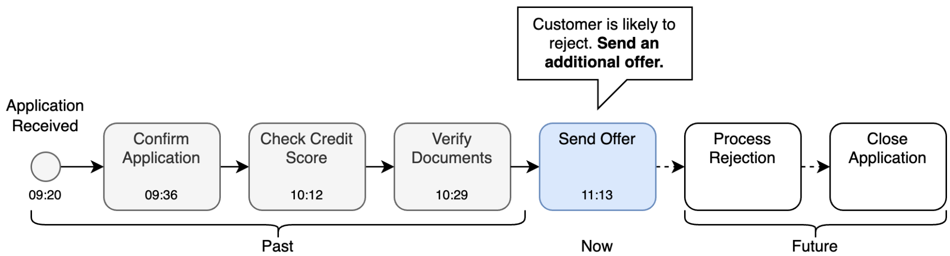

Figure 3. An illustrative example of the application of PrPM.

2.3. Human-Centered AI in Predictive and Prescriptive Process Monitoring

Human-Centered AI is a concept that emphasizes the importance of focusing the design and development of AI systems on humans' needs and values [130]. It requires various approaches to be fully operationalized: explainable AI, responsible (trustworthy and fair) AI, and user-friendly (usable) AI [63].

In the following sections, we elaborate on each of these concepts in the context of predictive and prescriptive process monitoring: Section 2.3.1 describes explainability, Section 2.3.2 – responsibility, and Section 2.3.3 – usability in PPM and PrPM.

2.3.1. Explainability in PPM and PrPM

Explainability aims at providing some sort of explanation to the user about how the output of the black-box AI model was reached [24]. The main goal is to help the users understand how the models make decisions and generate outputs.

Explainability is a challenge that is currently being actively researched in *predictive* process monitoring (e.g., [93, 41, 122, 153, 18, 92]). The range of works varies from explanations of specific models to frameworks for generating explanations. For instance, a conceptual framework for explainable process prediction provides guidance for design and development of explanatory predictive systems [92]. It proposes to generate explanations for a specific subjects (e.g., a process owner), based on the objective (e.g., requirements of a specific user group), and with a focus on outcomes (e.g., business vs technical outcomes). It also guides the choice of techniques for the explanations. Another study proposes a framework that helps with generating explanations for inspecting predictive models [153].

Other studies propose specific techniques for generating explanations for PPM systems. For example, Galanti et al. [41] utilize SHAP values to explain predictions in ongoing cases at run-time. SHAP (SHapley Additive exPlanations) values are a model-agnostic explainability technique that quantifies the contribution of each input feature to a particular prediction. Buliga et al. [19] propose a method for generating pattern-based counterfactual explanations for PPM.

According to one recent study on user evaluation of explanations in PPM [122], different plots (including visualizing SHAP values) are generally understandable for users with familiarity of BPM and ML. However, the study also suggests that analysts with foundational understanding of BPM and ML still struggle with metric-driven explanations and what conclusions to draw. Therefore, it is important to provide the end users, e.g., process analysts, with explanations that are understandable for them.

Explainability is still an emerging area of research within *prescriptive* process monitoring, though. Within PrPM, there is a need to explain not only the prediction, but also the recommendation. Several recent examples of incorporating explainability into PrPM exist. For instance, Padella et al. [112] use SHAP values to provide an explanation for the next activity recommendation in an ongoing case. With this, the user receives explanations about how the contribution of each variable would change following or not the recommendation [112]. However, evaluation of the approach with real users is yet to be done. Another study focuses on specifically introducing a white-box PrPM technique [134]. In it, a framework that enables stakeholders to define intervention policies in business processes is proposed. Another recent study incorporates the LIME algorithm for local feature explanation [5]. Thus, the user receives a textual explanation of why a decision is recommended based on the specific features of the instance (e.g., the customer is missing a visa, therefore the travel request should not be approved).

2.3.2. Responsibility in PPM and PrPM

Responsibility incorporates several concepts: fairness, accuracy, confidentiality, and transparency [89, 2]. As such, fairness refers to avoiding drawing unfair conclusions and communicating about biases present in the data. Accuracy includes providing interpretable measures for accuracy and communicating about the methods. Confidentiality incorporates respecting the privacy of users and protection of sensitive data. Last, transparency refers to providing full traceability to the data source and communicating about data quality problems [89, 2].

There are many studies focusing on responsibility-topics generally in process mining [89, 34]. Specifically in PPM and PrPM, responsibility can be considered from multiple perspectives, e.g., how to make fair and trustworthy predictions, how to ensure accuracy of data used in prediction models, and how to preserve the privacy of process participants in event logs [89]. In PPM and PrPM, these topics are yet to be explicitly addressed. As an example, a study by Padella et al. proposes a PrPM technique that explicitly takes into account resource experience [113]. The aim of the technique is to break the vicious cycle of recommending experienced resources to specific tasks (since their experience helps complete the tasks more efficiently) and thus avoiding the less experienced ones. As such, taking into account the less experiences resources helps to increase the fairness, and also give them the opportunity to increase their experience [113]. In another example, Bozorgi et al. [13, 14] proposed discovering causal rules from event logs and considering cost as a factor when generating recommendations for how to intervene in an ongoing case. By making the causal relationships explicit, i.e., understanding the underlying causes of different outcomes in the process, fairness can be incorporated into the decision-making [89]. For instance, if recommendations affect employees, fairness constraints might ensure that no particular employee group (e.g., gender, department, or seniority) is unfairly burdened or advantaged by the recommendation.

2.3.3. Usability in PPM and PrPM

Usability refers to the extent to which a system enables users to achieve specific goals effectively, efficiently, and with satisfaction within a defined context of use [139]. In the context of AI systems, usability is defined by the system's capacity to support effective human-AI collaboration through interfaces, integration into existing workflows, and reduction and operational barriers [63].

Since *predictive* process monitoring is more mature than prescriptive, many applications of user interfaces and visualization can be found in it (e.g., [122, 143, 93, 153, 41, 121, 64]). One notable example is Nirdizati, a web-based tool for PPM [64, 121]. It provides an overview of ongoing and completed cases with some summary statistics for them (e.g., number of events in the case and starting time), and visualization options, such as pie charts for case outcomes and bar charts for case durations. Based on this tool, a PPM dashboard was introduced in Apromore[2] [143], a commercial process mining tool. In this dashboard, end users, such as process workers and operational managers, can receive notifications whenever process targets are predicted to be violated.

[2] `https://apromore.com/`

Some works in *prescriptive* process monitoring proposed an algorithm and implemented an interface for it. For example, a study on discovery of treatment rules that increase the probability of positive case outcomes based on causal machine learning, also proposes a UI to help configure the technique and visualize its outcomes [13]. Thus, the interface is suited to users who have knowledge of machine learning. A recent study proposed a process-aware decision support system, integrated as a proof-of-concept in an IBM tool suite [5]. In a different study, the authors propose an interface that displays process goals, predictions, and recommendations [60]. In another study undertaken within the medical domain [155], a UI is proposed that visualizes a recommended treatment (a process trace) for a particular patient. In these two papers, interfaces are developed to exemplify a specific technique for a specific case. Such approaches, however, make it difficult to ensure that the information presented to users serves their needs.

The transition from *predictive* to *prescriptive* process monitoring brings about additional usability requirements on the underlying systems, as prescriptive monitoring systems not only present insights (predictions), but also actions (recommendations). In a *predictive* process monitoring system, it is important to ensure that the users understand the predictions that the system emits as well as the factors that determine these predictions. *Prescriptive* process monitoring requires interfaces that also communicate recommendations clearly. Previous research indicates that key usability factors for enhancing the user experience in prescriptive systems include ease of use and empowering the user to make informed decisions [136]. In this regard, LLMs are expected to play an increasing role in enhancing user interaction in prescriptive systems [105]. Integrating LLMs into these systems can improve usability by enabling natural language communication and presenting recommendations and their rationale in plain text, making them more accessible and easier for users to understand [105].

2.4. User-Centered Design

User-Centered Design (UCD) is used as a general term to describe a philosophy and methods which focus on designing for and involving users when designing various systems [3]. UCD emphasizes the importance of keeping the user in the loop, from eliciting the requirements, through systems design, to evaluating the usability and usefulness of a developed system [3, 146].

Below, we elaborate on user information needs (Section 2.4.1) and design principles for UIs in information systems (Section 2.4.2) to facilitate usability.

2.4.1. User Information Needs

The importance of considering the information needs of end users when developing systems aimed at helping them has been highlighted in examples from various fields. In this regard, the system should provide users with the information that will guide them to useful decisions [4]. In the field of human-robot interaction, it has been shown that presenting relevant information elements enhances situational awareness and, thus, helps in decision-making [26]. Similarly, identifying information items that are relevant to the situation helped in increasing an effective fire emergency response [109]. In healthcare information systems, guidelines exist on designing decision-support systems for clinical interventions [57]. The guidelines highlight the importance of adjusting the system attributes to the clinician roles.

Research in process mining has, however, historically seen developments of techniques for different use cases (as described in Section 2.1). Less attention has been paid to the applicability of those techniques in practice [159] through, for example, building usable and useful user interfaces. Just in the summer of 2023, a seminar was organized that was entirely dedicated to the topic of the role of humans in process mining [25]. As an outcome of this seminar, more works started emerging that provide guidance for developing tools that aim at addressing the end-user needs. For example, a Tiramisù framework was proposed which utilizes visual analytics for interactive visualization of multi-faceted process information [7]. The framework is designed to support end-users in their process analysis tasks by facilitating investigation and insight generation. For that, interface designers should follow the Tiramisù framework's four main components: backdrop (a layer serving as a common context for the process), dimension layers (overlay and on-deman layers superimosed on the backdrop), visual mappings (mapping from the process data, which can be a process model, statistics, an abstracted event log, or a combination of them, or non-process data to the backdrop), and visualization configuration (selecting and configuring the other three components of the framework) [7]. We use the Tiramisù framework to describe the PrPM interface developed in Chapter 5.

There are also some examples of specifically considering user needs for daily tasks in process mining. As such, in Klinkmüller et al. [74], the authors analyze Business Process Intelligence Challenge (BPIC) reports to work practices used by process analysts in process mining projects. The authors elicit information needs associated with visual representations. However, the results only refer to the visual representations used by analysts to describe certain data in the reports (e.g., conformance checking, drift detection). Similarly, other studies seek to explore the analysis strategies of process analysts [159, 76, 138]. For instance, Zerbato et al. [159] identify the analysis strategies and organize them into four distinct phases, each representing specific intermediate goals that guide the progression of the analysis. For each phase, the study outlines common challenges that analysts encounter. A study by Sorokina et al. [138] takes a theory-guided approach and aims to shed light on the analysts' cognitive processes when performing process mining analysis. In summary, the purpose of such works is to describe the practices used by process analysts in process mining which can further be used as a guidance when developing systems and tools for them (e.g., recently, a whole research project was initialized that is aimed at developing tools to support the daily work of process analysts [160]). In this thesis, we focus on understanding the needs of end users specifically in the context of PrPM.

Other research projects that consider information needs of end users in process mining have been emerging. For example, a project proposes to focus on understanding what healthcare professionals want to know before designing suitable dashboards [124]. However, an explicit study in this project is yet to be done.

In summary, designing an interface for PrPM outputs that contains relevant information items can help increase the effectiveness of task completion. This, in turn, leads to the the usefulness of the interface overall.

There are many methods that can be used to involve users in the design process [3, 120]. They can be classified based on the purpose they can be used for (e.g., initial inquiry into users' needs and preferences, usability evaluation). Depending on the purpose, they are applied in different parts of the design cycle (e.g., early on, mid-way, final stage). As such, at the beginning of the design cycle, background interviews and questionnaires are common to collect the data related to the needs and expectations of users [3, 120]. To validate the developed solution, usability testing based on interviews and questionnaires are used to collect quantitative and qualitative data related to user satisfaction with the artifact [3, 120]. Beyond usability, evaluating the usefulness of the developed solution is crucial for assessing whether it effectively supports users in their tasks. In this context, the Technology Acceptance Model (TAM) [29] provides a framework for understanding how users accept and adopt new technologies. TAM suggests that two key factors influence adoption: perceived usefulness – the extent to which users believe the system enhances their performance, and perceived ease of use – how effortless the system is to use. These factors, in turn, impact behavioral intention to use, which predicts actual adoption [29].

In our evaluation, we evaluate the usability of the interface and the usefulness of information items presented in it through semi-structured interviews in Chapter 5. We also apply interviews and questionnaires to evaluate developed artifacts in Chapters 4 and 6.

2.4.2. User Interfaces Design Principles

Design Principles in Information Systems

Over the years, many design principles for user interfaces have been proposed. The most widely referenced ones are those of Shneiderman [129], Nielsen [106], and Norman [108]. To unify and categorize the design principles, proposed by many authors, Ruiz et al. [125] conducted a systematic literature review. Based on the review, the authors propose a collection of 36 core principles, based on number of citations, that are considered the most relevant in user interface design. Table 2 presents an overview of the top nine (citations >20,000) principles that we also utilize in this thesis (Chapter 5). The design principles from the core collection have also been used in the development of user interfaces in information systems (e.g., [154, 119, 77, 137]).

Table 2. Design principles overview (adapted from [125]).

Design principle	Description
Offer informative feedback	User should always know where they are in the system and what is happening
Strive for consistency	All the words, elements etc. should remain similar in the system, and similar actions have similar result
Prevent errors	Suggestions or constraints on data to be inputted are offered, automatic checks for the inputted data performed
Minimize user's memory load	Options are clearly visible, hierarchical structure is used
Simple and natural dialogue	Dialogues should not contain unnecessary information
Good error messages	Error messages should explain the problem clearly and provide suggestion what to do next
Provide shortcuts (customizations)	Customization options, as well as shortcuts to actions
Provide clearly marked exits	Return options to previous action or page
Speak the user's language	Words and concepts tailored to the user's rather than the system's preferences

Several design theories for information systems in specific contexts have been proposed. For example, Landwehr et al. [80] formulate design knowledge for image-based decision support systems. Following the design principles of image acquisition, image processing, provision of metadata, provision of interpretability, and visual data exploration can help develop efficient and useful image-based decision-support systems (e.g., power line maintenance systems). Further, the focus in the research efforts has particularly been on developing design theories and principles for explainable systems. For instance, Herm et al. [51] proposed a design theory for explainable intelligent systems. The theory comprises ten design features. The design features include categories on global explanations (e.g., "Provide (performance) metrics"), local explanations (e.g., "Provide archive of historical decisions", "Provide information about decision alternatives"), and personalized interface (e.g., "Incorporate granularity and navigability").

In addition, design principles have been proposed for creating diagrams (including process models). For instance, Malinova and Mendling [88] elicited a cognitive framework, called CogniDia, to explain how people understand and perform tasks using diagrams. This framework combines key ideas from cognitive theories and applies them to shed light into how diagrams are used. In addition, it brings together a set of criteria that support effective cognitive processing when working with diagrams. Based on this, the authors propose guidelines to create diagrams for practitioners. The guidelines are categorized on the cognitive steps of visual processing, verbal processing, task processing, and semantic processing. For example, for the step of visual processing, the guidelines are to ensure that the diagram is aesthetically pleasing and simple, harmonize elements in the diagram, emphasize key elements, and group similar events.

Another key element for developing UIs is considering the visualization principles. Here, a widely referenced work is the book on information visualization by Ware [147]. In it, the author presents the principles of visualization to improve clarity, utility and persuasiveness. The principles concern a range of visual elements, such as color, shapes, patterns.

In summary, design principles provide a foundational framework for developing UIs. In specific contexts in information systems, these principles are adapted to address the challenges in the context. We use them in Chapter 5.

XAI Principles

In the context of AI systems, four principles for Explainable AI (XAI) were proposed [118] which emphasize how AI systems interact with end users. These principles are designed to address a wide range of motivations and use cases, and they apply broadly to AI systems, extending beyond machine learning to include other AI techniques. XAI principles complement UCD by ensuring that AI-generated outputs are transparent, interpretable, and useful to end users.

XAI principles constitute "Explanation", "Meaningful", "Explanation Accuracy", and "Knowledge Limits" [118]. The "Explanation" principle is formulated as *"A system delivers or contains accompanying evidence or reason(s) for outputs and/or processes."* It aligns with UCD's emphasis on providing users with information that supports their understanding and interaction with a system. AI systems should accompany their outputs with supporting evidence or reasoning, so that users can make informed decisions.

The "Meaningful" principle is formulated as *"A system provides explanations that are understandable to the intended consumer(s)."* [118]. It ensures that explanations are understandable to the intended end users. This reinforces the aim of UCD, that is to tailor information presentation to user needs. Different user groups may require different levels of detail or explanation styles.

The third principle is "Explanation Accuracy": *"An explanation correctly reflects the reason for generating the output and/or accurately reflects the system's process."* [118]. UCD methodologies, such as iterative testing and user feedback loops, help refine explanations to ensure they are both accurate and useful.

The last principle, "Knowledge Limits", is formulated as *"A system only operates under conditions for which it was designed and when it reaches sufficient confidence in its output."* [118]. It aligns with UCD by ensuring that AI systems communicate when they lack sufficient confidence or relevant data to provide a reliable output. This principle supports UCD by ensuring that users are not left to interpret ambiguous or unreliable AI outputs on their own.

In summary, XAI principles help make AI systems more aligned with user needs. This enhances usability, and supports more effective human-AI collaboration in decision-making processes. We incorporate XAI principles in Chapter 6.

3. PROBLEM IDENTIFICATION: SYSTEMATIC LITERATURE REVIEW

This chapter corresponds to the *Problem Identification* phase of DSRM in Figure 1. The purpose of this phase is to answer $\mathbf{RQ}_1$ *What are the outputs of PrPM techniques?* To answer it, we conduct a systematic literature review (SLR) [73]. SLR is a suitable method because it provides a structured approach to identifying and analyzing existing research [73]. An SLR ensures comprehensive coverage of the topic, enabling the identification and classification of outputs produced by PrPM techniques. Understanding the range of these outputs is the first step to gaining insights into how they can be made usable to end users.

Section 3.1 describes the method, and Section 3.2 shows the results of the review: PrPM objectives, prescribed recommendations, required data input, policy, user evaluation aspect, and a framework to classify PrPM techniques. Then, in Section 3.3 we discuss research gaps and implications, and in Section 3.4 we provide a summary.

3.1. Method

We followed the guidelines proposed by Kitchenham [73] which consist of three steps: (1) planning, (2) conducting, and (3) reporting. The first step includes motivation for a review, definition of research questions, and development of a review protocol. The second step includes identifying studies, selecting primary studies, quality assessment, data extraction, and data synthesis. Finally, the third step includes the dissemination of the report.

(1) Planning the review

For the first step, we identified research questions. Then, we developed a search strategy, identified suitable electronic databases, and defined inclusion and exclusion criteria. Finally, we defined the data extraction strategy.

Research Questions. We decomposed $\mathbf{RQ}_1$ into a set of sub-questions. These questions serve to address the different aspects of PrPM outputs.

- Given that PrPM techniques aim at prescribing recommendations that produce business value, i.e., achieve an objective, we formulate the first research question as: RQ1.1. *What are the objectives of PrPM techniques?*
- The second research question aims at discovering how these objectives can be achieved: RQ1.2. *What recommendations are prescribed by existing PrPM techniques?*
- Third, we explore the data required by the proposed techniques: RQ1.3. *What input data do PrPM techniques require?*
- Fourth, we review what policies PrPM techniques use to decide when to provide recommendations. Policy refers to deciding in which cases to pre-

scribe a recommendation: RQ1.4. *What policies do PrPM techniques employ?*

- Finally, we explore whether the output of the proposed techniques is evaluated with end-users (through a user interface or otherwise): RQ1.5. *To what extent have the PrPM outputs been evaluated with end-users?*

Search Strategy. In the search string, we included "process mining" to scope the study to techniques that rely on event logs. We derived the term "prescriptive" from the research questions. We also included the terms "recommender" (e.g., [84, 71]) and "decision support" (e.g., [44]), as we found these terms to be sometimes used instead of "prescriptive". Accordingly, we formulated the following search string: *(recommender OR "decision support" OR prescriptive) AND "process mining"*

While conducting the first search, we noted that the term "prescriptive process mining" was not consistently used. Therefore, only using this search string might have resulted in missing relevant studies. We addressed this issue by examining the papers we identified using the first search string to identify other terms. We noted that terms such as "next-step recommendation" [60], "recommendation system" [27], "next best actions" [150], "proactive process adaptation" [95] have been used synonymously for "prescriptive process monitoring". We also noticed that the phrase "business process" often appeared in titles and keywords. Therefore, we formulated a second search string: *(recommender OR "next activity" OR "next step" OR "next resource" OR proactive) AND "business process"*

Electronic Libraries. We applied both search strings to ACM Digital Library, Scopus (includes SpringerLink), Web of Science, and IEEE Xplore to identify potentially relevant papers. The databases were selected based on the coverage of publications within the field of process mining. Finally, we conducted backward and forward referencing [110] to identify additional relevant papers.

Table 3. Exclusion/inclusion criteria applied.

Criterion	Description
EC1	The paper is digitally accessible.
EC2	The paper language is English.
EC3	The paper is not a duplicate.
EC4	The paper is longer than six pages.
IC1	The paper is relevant to the domain of prescriptive process monitoring.
IC2	The paper presents, reviews, discusses, or demonstrates a technique or a case for prescriptive process monitoring.
IC3	The paper describes at least one way to identify candidate recommendations for an ongoing process case.

Inclusion and Exclusion Criteria. Next, we defined exclusion and inclusion criteria (Table 3). We excluded papers for which the answer to any of the defined exclusion criteria was "no": not digitally accessible (EC1), not in English (EC2), duplicates (EC3), and shorter than six pages (EC4). Exclusion criterion EC1 ensured that papers could be accessed, and EC2 that they could be understood by the author. Papers unavailable through open access, institutional subscriptions, or general internet searches were excluded, as were those written in languages other than English. Criterion EC3 eliminated duplicates resulting from papers being indexed in multiple digital libraries. EC4 was applied to exclude papers under six pages, as these were unlikely to provide detailed information about PrPM techniques necessary for our analysis.

We also defined three inclusion criteria: (IC1) the paper is relevant to the domain of PrPM, (IC2) it presents, reviews, discusses, or demonstrates a technique or case related to PrPM, and (IC3) it describes at least one way for identifying candidate recommendations for ongoing process cases. Papers had to meet all inclusion criteria to be retained. IC1 ensured the inclusion of papers within the PrPM domain. IC2 focused on including studies that provided theoretical discussions or practical applications of techniques. IC3 guaranteed that the papers contained sufficient information to address our research questions. Following a top-down approach, papers failing any of the inclusion criterion were excluded without further evaluation against remaining criteria.

Table 4. Data extraction form.

Extracted Data	**Description**
Identification Data	
ID	Unique identifier of the paper
Title	Title of the paper
Author(s)	Authors of the paper
Year	Year of publication of the paper
Publication Venue	Venue where the paper was published
Study Context	
Process	Type of the process used in the example
Domain	The domain the dataset represents
Company	The company type in the domain the dataset represents
Dataset	Whether the dataset is real (taken from a real company) or synthetic (generated artificially)
User Evaluation/UI	Whether an end-user evaluation is conducted or UI is presented
Prescriptive Parameters	
Recommendation	Specific recommendation prescribed
Process Aspect	The process aspect (e.g., control flow) for which the recommendation is prescribed
Objective	Why the recommendation is prescribed
Performance Metric	Performance metric to measure the effectiveness of the prescribed recommendation
For Whom	Who the recommendation is prescribed for (e.g., process worker)
Data & Method	
Input	Input data used in the technique
Policy	Policy used to prescribe the recommendation

Data Extraction Strategy. Finally, we established the data extraction strategy (Table 4). We began by capturing the metadata for all papers, including the title, authors, publication venue, and year. Next, we identified the specific data needed to address each research question. For RQ1.1, we focused on extracting information about the objectives of using PrPM techniques and the performance metrics targeted in each study. For RQ1.2, we identified data on the prescribed recommendations, the process perspectives considered, and the intended users of these recommendations. To address RQ1.3, we extracted details about the input data required by the techniques described in the papers. Additionally, we collected information on the policies employed to trigger recommendations (RQ1.4). Finally, for RQ1.5, we explicitly marked the studies that conducted an end-user evaluation through a user interface or otherwise.

(2) Conducting the review

We executed the search and identified a total of 2614 papers (see Figure 4).[3] We filtered them using exclusion criteria EC1 and EC2. This resulted in the removal of 516 papers. Thus, 2098 papers remained and were filtered based on EC3. With this, 324 more papers were removed. Out of the remaining 1774 papers, we removed short papers (EC4). This resulted in 1613 papers remaining. These were filtered by title, thus removing papers that were clearly out of scope. The remaining 270 papers were filtered by abstract, resulting in 100 papers remaining. Finally, we applied the inclusion criteria by reading the whole paper and removed 78 papers. As a result, 22 papers remained. A total of 12 papers were added through backward referencing. This could be explained by the fact that the terminology is not consistent. As a result, the final list contained 34 papers.[4]

(3) Reporting the review

We describe the selected papers along the dimensions specified in the subresearch questions. In addition, we derive a framework to classify the identified PrPM techniques. To do so, we started by clustering the techniques according to what they were aiming to improve (RQ1.1), e.g., "cycle time minimization", "cost optimization". For each group, we followed the research questions to classify the techniques further, such as following the recommendations they prescribe (RQ1.2), the input data they require (RQ1.3), and the policies they utilize to prescribe recommendations (RQ1.4), as well as marking the studies that conducted a user evaluation (RQ1.5).

In addition to the framework, we also provide an overview of research gaps identified from the studies review.

[3]The search in the journal publication, on which this chapter is based [66], was conducted in September-October 2021. We conducted an additional search in December 2024 to add the papers that were published in 2021-2024 period. The numbers in the figure are aggregated numbers.

[4]While conducting the new search, we also revised the previously included studies. Additionally, we identified studies published in 2021-2024 that extended the previously included studies. Therefore, we included the extended study and removed the previous version (we elaborate on this in the description of primary and subsumed studies in the next section).

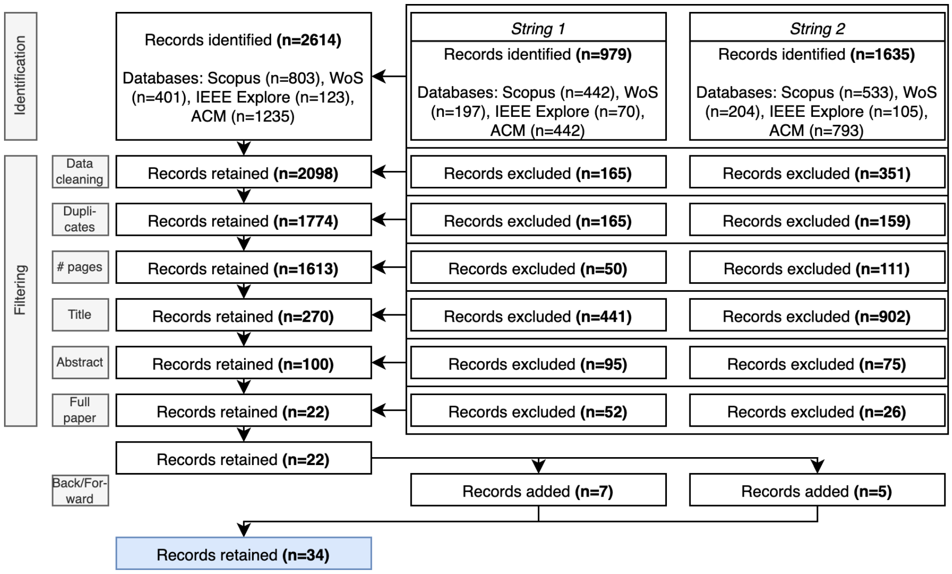

Figure 4. Papers filtering process.

3.2. Results

In this section, we report the results of the review. In Section 3.2.1, we describe the primary and subsumed studies, provide a quantitative overview of the primary studies, and give a summary of other identified papers in the PrPM domain that were not included in the list. The next sections relate to the sub-research questions. Section 3.2.2 describes PrPM objectives (RQ1.1). In Section 3.2.3, we describe the recommendations along their perspective, frequency, and purpose (RQ1.2). Section 3.2.4 outlines data input used in the identified studies (RQ1.3), and Section 3.2.5 the policies under which recommendations are prescribed (RQ1.4). In Section 3.2.6, we give an overview of studies that conducted a user evaluation (RQ1.5). Finally, Section 3.2.7 rounds the review into a framework to classify the PrPM techniques along the above dimensions.

3.2.1. Studies Overview

Primary and Subsumed Studies

During the selection process, we noted primary and subsumed studies. Primary studies are original studies that build upon and extend the findings of subsumed studies. In other words, primary studies contribute new methods or findings that expand the subsumed studies. For example, a technique was proposed in a conference paper [12], and a later journal paper improved the technique and added new dimensions to it [15]. Therefore, the conference paper is a subsumed study, and the journal paper is a primary study. Table 5 provides an overview.

Table 5. An overview of primary and subsumed studies.

Primary		Subsumed	
Haisjackl et al. (2010)	[48]	Schonenberg et al. (2008)	[126]
Nezhad et al. (2011)	[104]		
Barba et al. (2012)	[8]		
Nakatumba et al. (2012)	[101]		
Kim et al. (2013)	[72]		
Gröger et al. (2014)	[45]		
Ghattas et al. (2014)	[44]		
Wibisono et al. (2015)	[152]		
Huber et al. (2015)	[60]		
Conforti et al. (2015)	[27]		
Sindhgatta et al. (2016)	[135]		
Yang et al. (2017)	[155]		
Park and Song (2019)	[115]		
Dees et al. (2019)	[30]		
Weinzierl et al. (2020b)	[150]	Weinzierl et al. (2020a)	[149]
Metzger et al. (2020)	[95]		
de Leoni et al. (2020)	[84]		
Shoush and Dumas (2021)	[131]		
Khan et al.	[71]		
Fahrenkrog-Petersen et al. (2022)	[37]	Teinemaa et al. (2018)	[141]
Branchi et al. (2022)	[16]		
Agarwal et al. (2022)	[5]		
Zeltyn et al. (2022)	[157]		
Park and Song (2023)	[116]		
Padella and de Leoni (2023)	[111]		
Donadello et al. (2023)	[31]		
Bozorgi et al. (2023b)	[15]	Bozorgi et al. (2021)	[12]
Bozorgi et al. (2023a)	[14]		
Park et al. (2023)	[114]		
Hermann et al. (2024)	[52]		
Padella et al. (2024)	[113]		
Shoush and Dumas (2024)	[133]		
Seidel et al. (2024)	[127]		
Shoush and Dumas (2025)	[134]	Shoush and Dumas (2023)	[132]

Quantitative Overview of Primary Studies

The distribution of the papers over years of publication is depicted in Figure 5. Among the identified papers, we note a spike of publications since 2022. Thus, the years 2022-2025 produced 15 out of 34 papers, and in the year 2023, six papers were published. All identified papers were peer-reviewed, with 10 of them being journal articles and 24 conference papers.

We also note that the majority of papers use real-life event logs to validate their techniques. As such, 20 papers utilize real-life event logs (among which are personally acquired logs and logs from the Business Process Intelligence Challenge[5]), 10 papers use synthetic logs, and three papers conduct validation on both real-life and synthetic logs. The choice of event logs also dictates the domain.

[5] `https://www.tf-pm.org/competitions-awards/bpi-challenge`

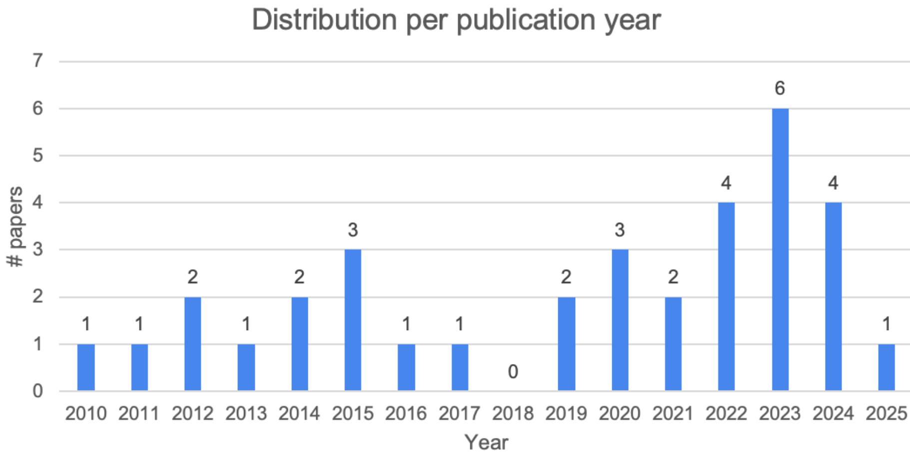

Figure 5. Distribution of papers per publication year.

For example, among BPIC event logs, the most popular are the BPIC'12/BPIC'17 logs from a financial institution (used in 11 papers). Logs acquired personally are from manufacturing firms, healthcare, and IT services.

Other Related Works

During filtering, we identified several papers that did not propose a PrPM technique (and thus were not included in the final list), but were relevant to the PrPM domain. In this sub-section, we describe these papers.

Several papers compare different computational approaches for specific PrPM tasks. As such, Weytjens et al. [151] compare two approaches that are common in PrPM techniques, reinforcement learning (RL) and causal inference (CI), on synthetic data. The authors conduct comparative experiments using these two techniques specifically for what they refer to as "timed process interventions" (we refer to them as "discrete recommendations" later in this chapter (see Section 3.2.3)). The authors conclude that online RL produced better and more robust policies than CI. Similarly, Mertens and Gailly [94] compare various types of neural networks, their topologies, and input encodings with appropriate benchmark techniques to determine the most effective options for next-activity recommendations in three healthcare event logs, evaluating them based on accuracy, consistency, training speed, and prediction speed. Metzger et al. [96] perform a comparative evaluation of several approaches for balancing the trade-off between prediction accuracy and prediction earliness in PrPM, using four public real-world event logs and two types of prediction models. By prediction earliness, the authors mean that there are multiple points in time when end-users may decide whether or not to trust the current prediction and act upon it [96]. The study compares cost savings across approaches, providing insights into their effectiveness and initial guidance for selecting the most suitable one in practice.

Yadegari and Karastoyanova introduce the concept of "autonomic process improvement" [43]. The authors propose an architecture designed to integrate predictive monitoring with automated process adaptation, which would enable processes to autonomously adjust in response to predicted violations. Compared to PrPM, this paper emphasizes autonomic, automated adaptations. While PrPM typically requires user intervention to implement recommendations, this paper brings forward a conceptual architecture and a research agenda to fully automate the process performance improvement, moving towards self-adjusting systems.

Two other papers propose conceptual solutions for the end-user aspects of PrPM. In this regard, Leemets et al. [83] explore the application of persuasiveness principles in PrPM visualizations, i.e., how to nudge the end-user to follow the recommendation given by the system. The authors collect a set of persuasiveness principles from the literature and develop a mockup with a fictional output of a general PrPM system. They evaluate the mockup with potential end-users and draw a set of suggestions for designing persuasive PrPM visualizations. However, the study does not propose, nor include a specific PrPM technique in the mockup. In another study [53], the authors develop a framework for promoting the adoption of recommendations in PrPM by incorporating organizational practices. The framework, designed from a socio-technical perspective, addresses the challenges of why process workers may not follow the recommendations despite understanding them. Through an empirical case study, they derive essential requirements and propose a framework that embeds the recommendations within the organizational structure [53].

3.2.2. PrPM Objectives

In our review, we identified two primary objectives that PrPM techniques aim to achieve: *optimizing the process outcome* and *optimizing the process efficiency*. This classification aligns with the definitions of quality and efficiency in service processes [32]. The objective of optimizing the process outcome focuses on ensuring a favorable process result. This is often represented using binary metrics, such as preventing a deadline violation [45] (in this case, the outcome is that the deadline is either violated or not). On the other hand, the objective of optimizing process efficiency relates to improving a specific quantitative aspect of process performance, such as minimizing cycle time [116].

The papers that correspond to the objective of *optimizing the process outcome* optimize either a temporal or a categorical outcome. Thus, for temporal outcomes, it can be a violation of planned cycle times or deadlines [45, 84, 157, 150]. For instance, in a manufacturing process, a recommendation is issued to a process worker to adjust the settings on the specific machine to avoid the predicted cycle time violation [45]. In another example by Zeltyn et al. [157], a management incentive process is described. In this process, people managers in a software company have a deadline on a specific day to submit their proposals for the employees' salary increase. The technique developed in this study provides recommendations to avoid violating this deadline [157].

Another group of studies focuses on avoiding or mitigating undesired categorical outcomes [37, 95, 5, 31, 133, 44]. For example, a technique uses the loan application process and designates the desired outcome as the client accepting the loan, and undesired as the customer rejecting it [131]. Another study experimented with multiple processes, defining categorical outcomes in each [5]. As such, the authors define the same objective in a loan application process, as the study above. In another process that describes travel request for a worker in a software company, a positive outcome is that the travel request is granted [5]. One technique [44] introduces the concepts of "hard" and "soft" goals. As such, the authors describe a bottle manufacturing process, where the hard goal is that the customer accepts the delivery. At the same time, various soft goals can be defined, such as overall cost, product quality, time to delivery [44].

The second primary objective focuses on *optimizing process efficiency*. Most studies in this category address temporal aspects, such as cycle time or processing time. For instance, several papers explicitly define reducing cycle time as their main objective [104, 152, 101, 116]. In this context, reducing cycle time means that the cycle time should be gradually improved by ensuring that each coming case takes less time than the previous average. Thus, it differs from the previously described objective of avoiding deadline violation. For example, assigning a resource (police officer) best suited for the next task decreased the average completion time in the driving license application process [152]. In other temporal metrics, Park and Song [115] aim to reduce processing time, the time spent by a resource resolving a task [32]. They present an example of reducing the processing time of manual tasks in a loan application process [115].

Beyond temporal metrics, some techniques target reducing defect rates, such as minimizing the likelihood and severity of faults through risk prediction [27]. Further, in a process of unemployment benefits process, granting reclamations to those people who are actually not eligible for them induces unnecessary cost for the company. Thus, the technique aims to minimize those costs [30].

Some techniques are flexible and can optimize various process efficiency aspects depending on the context [48, 16, 71]. For example, Branchi et al. [16] shows examples of two processes, loan application and traffic fines management. In the former, the technique aims to optimize the profit of the bank, such that it minimizes the cost of granting a loan to a customer (the cost of employee working time), while maximizing the interest of the bank granting the loan. In the latter, the aim is to maximize the credits received by the police department based on the fine payments received by the offender [16].

Finally, we identified studies that present techniques able to adapt to both objectives [15, 127]. As such, Bozorgi et al. [15] propose a technique that deals both with efficiency outcomes (e.g., cycle time, cost) and with binary case outcomes (positive vs. negative case outcome). For example, in a process describing the work at a call center of an administrative authority, a negative outcome is if the customer calls again within one month (yes or no) (optimizing the process outcome). To avoid that, the proposed activity is to increase the duration of the first call, thus providing more explanations to the customer's initial question. But the authors also give an example of a travel reimbursement process at a university, where the cycle time gets extended if a reimbursement request needs to have an additional approval from a decision-maker (optimizing the process efficiency). Thus, the technique provides recommendations to skip this activity in those cases where possible, thus reducing the cycle time [15].

3.2.3. Prescribed Recommendations

The recommendations provided by PrPM techniques in running cases can be categorized according to perspective (control flow, resource, various), frequency (continuous, discrete), and purpose (optimizing, guiding).

Recommendation Perspective

A common recommendation perspective is *control flow*, such as prescribing the next activity to perform [84, 157, 16]. More specifically, in de Leoni et al. [84], the next best activity is prescribed to the professional who helps a customer find a new job in an unemployment benefits process. In Khan et al. [71], a recommendation of the best path (representing sequence of activities leading to the process end) is based on a set of pre-defined KPIs. Another set of studies develops techniques that recommend a specific activity at a specific point in time [13, 14, 15, 133, 134, 30]. In such studies, an activity is pre-determined by, for instance, domain expert's knowledge. Then, the technique estimates whether applying this specific activity will have a positive effect on the outcome of the case. For example, Dees et al. [30] describe an unemployment benefits process at a social security institute. A situation might occur where customers incorrectly file the required forms, and thus receive reclamations which they are not eligible for. To avoid this, the proposed technique identifies customers with a high risk of getting a reclamation. As defined by the domain experts in this process, such an activity would be to send an email to the customer with explanations on how to fill the form correctly before they do so [30].

Another group of techniques focuses on the *resource* perspective. For instance, in Park and Song [116], resources are allocated for the predicted next tasks in an emergency department in a hospital. In another study, a mechanic is recommended to carry out car repairs because they are predicted to finish the task within a defined time given their schedule and expertise [135]. In Kim et al. [72], the best resource is recommended based on the predicted resource performance which allows for minimization of processing time or labor cost. Padella et al. [113] expand the recommendation of resources with an additional attribute, resource experience. In the above examples, the best-fitted resources are prescribed to the tasks in order to reach a desired objective. In this technique, however, the authors aim to balance the resource assignment also with less experienced resources. In this way, the authors aim to increase the fairness of resource allocation, such that less experienced resources get a chance to increase their experience [113].

Some techniques are designed to prescribe recommendations from both perspectives in the same case [131, 111, 104]. For example, in a loan application process, a recommendation to make an offer to a client is prescribed together with a suggestion for a specific clerk to carry out the task [131]. Similarly, in an IT service management process, recommending the next task and the specialist to perform it can help to resolve open cases faster [104].

There are techniques that can be labeled under "various" recommendations. As such, the work by Donadello et al. [31] focuses on control flow but, but unlike other techniques, it does not prescribe the next sequence of activities; instead, it defines a set of temporal constraints that must be met. For example, in a process of Sepsis treatment, an activity that refers to a certain drug treatment has to be immediately followed by a medical test to avoid that the customer goes into intensive care. Another paper describes a bottle manufacturing process, for which different decisions are recommended [44]. As such, the process worker is given recommendations about the material to use, the quality inspection level to set to achieve the goal of delivery acceptance. Hermann et al. [52] provide power-cost minimizing recommendations for process scheduling in an energy sector.

Finally, we identified studies that notify the user to intervene in an running case, but the choice of the specific action is left to the user [37, 95]. For example, Fahrenkrog-Petersen et al. [37] name such notifications "alarm." In their study, the authors define a threshold for the probability of a negative outcome of the case. When the probability exceeds this threshold, an alarm is raised to the user. Another paper employed a similar approach, however referring to it as a "proactive adaptation" [95].

Recommendation Frequency

When reviewing the identified papers, we noted that recommendations can be classified according to recommendation frequency. Recommendation frequency describes when recommendations are prescribed. In this regard, PrPM techniques can be *continuous* or *discrete*.

If a recommendation is *continuous*, the technique prescribes it for multiple or all activities of an ongoing case (e.g., prescribing the best-suited resource for each next task [152]). Similarly, Seidel et al. [127] propose a technique for knowledge-intensive processes (such as healthcare). In it, they aim to guide knowledge workers in selecting the most suitable next action that aligns with their goals and minimizes execution costs.

Discrete recommendations, in comparison, prescribe actions to be taken only when a need is detected. For instance, in Metzger et al. [95], recommendations are triggered only when it is detected that the probability of a negative outcome exceeds a defined threshold. In an example of a loan application process, a specific activity is defined as a recommendation, namely, to send additional offers to the customers who are likely to positively respond to it [15].

Recommendation Purpose

We observe that the techniques can also be categorized according to the purpose of the recommendation. As such, the purpose describes whether a technique is *optimizing* or *guiding*.

On the one hand, *optimizing* techniques propose recommendations aimed at improving the performance of an ongoing case with respect to a specified performance measure. These recommendations are frequently derived using machine learning (e.g., [45, 84, 37, 16]). In such cases, the technique predicts the potential outcomes of the ongoing case and subsequently prescribes a recommendation. These techniques are based on machine learning/data mining prescriptive analytics methods [85].

Optimizing techniques can be further categorized as either *correlation*-based or *causality*-based. As an example of a *correlation*-based technique, in Khan et al. [71], the technique suggests recommendations grounded in predictions derived from historical execution data, which is labeled with corresponding outcomes. This approach employs past execution data to train a model that correlates potential recommendations with their predicted effectiveness. Likewise, in the technique presented by Gröger et al. [45], decision trees are utilized to provide recommendations, where a categorized metric value is associated with specific attributes of a case. Furthermore, the technique by Weinzierl et al. [150] learns a deep neural network (DNN) for predicting next activities and values of a KPI. Then, the next best activities are recommended based on the KPI value predictions [150].

In contrast, the technique described by Shoush and Dumas [131] estimates the probability of an undesirable outcome and evaluates the impact of a particular recommendation on the outcome of the case. This technique first predicts the likelihood of an undesirable outcome, followed by the construction of a causal model to estimate the Conditional Average Treatment Effect (CATE) of the recommendation on the given case. In the objective of process outcome optimization, the CATE is the increase (or decrease) in the probability of a positive (or negative) case outcome. In the objective of process efficiency optimization, the CATE is the increase (or decrease) in the efficiency of the process. Similarly, in the technique proposed by Bozorgi et al. [15], a causal estimation module is employed to evaluate the effect of a specified recommendation on an ongoing case. These are some examples of *causality*-based techniques.

On the other hand, *guiding* techniques employ other types of prescriptive analytics methods, such as mathematical programming (e.g., mixed integer programming, stochastic optimization), evolutionary computation (e.g., greedy algorithm), or logic-based models (e.g., association rules, criteria-based rules) [85]. For example, the technique by Hermann et al. [52] is a mixed integer linear program that finds an optimal processing schedule for the active cases in the power plant process and their respective pending activities within a specified time horizon. The authors motivate the choice of the approach by specifying that they intentionally did not use machine learning to keep the procedure of how the recommendation is generated clear, understandable, and trustworthy, and to achieve reasonable results independent of the size of the dataset [52].

3.2.4. Required Data Input

Our review indicates that PrPM techniques use *control flow, resource, temporal, and domain-specific* data.

As expected, techniques that prescribe recommendations affecting control flow, such as the next task to be executed, rely on *control flow* data. For instance, in [27], decision trees are applied to data such as task duration and frequency to predict the likelihood of a case fault, e.g., exceeding the maximum cycle time and cost overrun.

Different kinds of *resource* data are used, e.g., resource performance, experience, labor cost. For example, service time of the resources is taken into account to provide resource allocation for the next tasks in a medical process, with the aim to reduce the total weighted completion time [116]. Another technique takes resource experience into account when assigning resources, thus aiming to provide fair treatment of resources with lower level of experiences [113].

Temporal data, e.g., day of the week, is also used to prescribe recommendations. Such data is commonly used in combination with other data, such as control flow or resource data. For instance, the most suitable resource for executing the next task is identified by considering the time of day (morning, afternoon, or evening), the inter-arrival rate, and task queue data as inputs [152]. In another example, [15] use temporal information, including the month, weekday, hour, and the period of inactivity before the most recent event, to assess the effectiveness of a recommendation aimed at reducing cycle time.

Domain-specific data, such as materials used in manufacturing processes [44], patient demographics, and treatment attributes in healthcare settings [155], is also employed as input for the techniques. For instance, in a loan application process, the data on changes in the number of loan offers sent to applicants and changes in the monthly interest offered to the applicant are used to capture the domain-specific requirements [133].

3.2.5. Policy

In PrPM techniques, the policy helps to determine in which cases to prescribe a recommendation [15]. For example, if a policy is random, then for each case it is randomly decided whether to prescribe a recommendation in it or not. Specific policies help to improve this decision. We first present policies that are applied in *guiding* techniques, and then the ones applied in *optimizing* techniques.

Some *guiding* techniques employ a similarity-based policy, in which a recommendation is prescribed based on the similarity between the current case and previous cases (e.g., [155, 104]). For instance, one technique finds next steps and ranks them according to their potential match to the context information of the open case, and the likelihood that they lead to a faster resolution based on information from past cases [104]. Other examples of policies in guiding techniques are a set of rules [8] or other process-specific constraints (e.g., change of power consumption or prices [52]).

One policy employed by *optimizing* techniques is the threshold-based probability of a negative outcome, wherein a recommendation is prescribed when a predicted value exceeds a predefined threshold (e.g., [37, 95]). In such approaches, the probability is evaluated in conjunction with other factors. For example, Fahrenkrog-Petersen et al. [37] combine the probability of a negative outcome with a cost model, which includes the costs associated with the recommendation, the undesired outcome, and any necessary compensations, along with an evaluation of mitigation effectiveness.

Several techniques are adapted to multiple types of policies, i.e., the policy can change depending on the purpose (e.g., [60, 15]). For example, in Bozorgi et al. [15] the final policy can be specified by the user. Specifically, first, the technique estimates the effect of the recommendation (e.g., calling the customer) on the cases. The estimates are used to create a net-value curve, which the policy maker uses to decide the percentage of future cases that will get the phone call based on organizational constraints. For example, the policy maker decides to prescribe the recommendation to half of the cases. This means that if the estimated effect for a new case is in the top 50% of the cases so far, the recommendation will be applied [15]. Two papers do not pre-define a policy, but aim to find the optimal one using different methods [14, 16]. For example, Branchi et al. [16] use reinforcement learning (RL) to compute an optimal policy that dictates which actions should be taken at each state in the process to maximize the reward (e.g., the KPI of interest). The policy is iteratively refined using Monte Carlo methods or policy iteration [16].

Other policy examples include recommendations triggered by exceeding predefined metric limits (e.g., [45, 150]). For example, the technique described in Gröger et al. [45] monitors the values of a target metric and prescribes adjustments to resource settings when a deviation from the target is observed. Other techniques propose a rule-based policy which can also be tuned by an analyst [134]. As such, Shoush and Dumas [134] develop a technique allowing analysts to filter cases for recommendation and then rank them based on importance, urgency, uncertainty, and capacity.

3.2.6. User Evaluation

The importance of finding the way of presenting the prescribed recommendations to the end-users and receiving feedback on them has been highlighted by Dees et al. [30]. In this study, the recommendation was to proactively send an email to the client. However, the step between receiving the recommendation and applying it was manual. In other words, the process workers were told to do so based on the recommendation that the researchers obtained from the technique, and not through their own interaction with a user interface.

Our review shows that only a handful of studies propose a user interface to bridge the gap between the technique output and the end-user application (we identified six: [5, 155, 60, 127, 52, 72]). For example, Agarwal et al. [5] introduce a UI in which a window pops up with a suggested action to proceed. As such, in a claims management process, a decision to accept the claim is presented to the user, along with a technique confidence expressed in percentage, and a short textual explanation of the top-3 reasons for the recommendation. Next, the UI in Yang et al. [155] is presented in the form of a web app which includes several visualizations: process flow visualization as a horizontal stacked diagram, process clustering as a dot visualization, and a recommendation in the form of text. The UI in Huber et al. [60] shows the remaining time, case goals, and deadline for the case, as well as the recommendation expressed as text. A study that used decision trees provided an possibility for the users to view the visualized decision trees [72]. The study by Seidel et al. [127] provides a prototype where process workers can input their goals through a form. Based on that, they receive the output in the form of scores, where the activity with the highest score should be conducted next. Hermann et al. [52] developed a software prototype for their technique and conducted an evaluation with four experts from the energy sector. However, the prototype does not feature an interface, but rather, outputs the results as a table.

Evaluation studies were conducted in three of the six papers. However, Seidel et al. [127] evaluated the usefulness of the recommendations. The 22 participants were split into three groups: (a) seven were given no recommendations; (b) seven were provided filter-based recommendations; (c) eight participants got scores as recommendations. However, the participants of the evaluation were students, and not the potential end-users of the tool. Next, Yang et al. [155] aimed at evaluating the performance of the technique (the output of which was shown in the interfaces) by comparing its output with what the domain experts would designate as the most suitable next task in the medical process. Finally, Hermann et al. [52] evaluated the developed software prototype through interviews with 10 experts. The evaluation assessed whether the prototype fulfills the set objectives, as well as established evaluation criteria of generality, efficiency, operationality, effectiveness, and usefulness [52]. In summary, Hermann et al. [52] was found to be the only study that specifically evaluated not only the technique, but also the user interface.

3.2.7. Framework

Having reviewed the papers, we followed the research questions to identify the characteristics of the PrPM techniques we identified. Starting from RQ1.1, we clustered the techniques according to their objective. We then followed the remaining research questions to detail the clustering. As a result, the proposed framework characterizes PrPM techniques based on 12 characteristics (see Figure 6 for the objective of optimizing process outcome and both; and Figure 7 for the objective of optimizing process efficiency).

Objective	Target	Rec. perspective	Recommendation	Input perspective	Policy	Rec. frequency	Prescriptive analytics method	Rec. purpose	Detail. algorithm	User eval./UI	Reference	Number ref.
Optimizing process outcome	Temporal outcome	Resource	Resource settings	All available	Exceeding metric limits	Continuous	ML/DM, Logic-Based models	Optimizing (cor.)	Y		Gröger et al. (2014)	[45]
			Resource allocation	R; T; D	Highest predicted metric value	Continuous	ML/DM	Optimizing (cor.)	Y		Sindhgatta et al. (2016)	[135]
		Control flow	Next activity	T; C	Multiple options (time-based, deadline-based, decision-based, goal-based)	Continuous	ML/DM	Optimizing (cor.)	N	*	Huber et al. (2015)	[60]
				C	Exceeding metric limits, maximum metric improvement	Discrete	ML/DM	Optimizing (cor.)	Y		Weinzierl et al. (2020b)	[150]
				C; R	Predicted outcome violation	Continuous	ML/DM	Optimizing (cor.)	Y		Zeltyn et al. (2022)	[157]
				Metric specific	Maximum metric improvement	Continuous	ML/DM, Simulation	Optimizing (cor.)	Y		de Leoni et al. (2020)	[84]
	Categorical outcome	Control flow	Specific activity to perform	T; D; C	Resource availability, urgency of the case, necessity of intervention, and uncertainty in predictions	Discrete	ML/DM, Causal Inference	Optimizing (caus.)	Y		Shoush and Dumas (2024)	[133]
				T; D; C	Decision rules based on outcome prediction, intervention effectiveness, urgency, and resource availability	Discrete	ML/DM, Causal Inference	Optimizing (caus.)	Y		Shoush and Dumas (2025)	[134]
				T; D; C	Optimal policy learned by the agent	Discrete	ML/DM, Causal Inference	Optimizing (caus.)	Y		Bozorgi et al. (2023a)	[14]
		Varies (multiple)	An alarm to trigger an intervention	All available	Probability of a negative outcome above a threshold, cost model, mitigation effectiveness	Discrete	ML/DM	Optimizing (cor.)	Y		Fahrenkrog-Petersen et al. (2022)	[37]
				All available	Probability of a negative outcome above a threshold, cost model, reliability estimate	Discrete	ML/DM	Optimizing (cor.)	Y		Metzger et al. (2020)	[95]
			Various	R; T; D	Optimal value of target metric	Discrete	ML/DM	Optimizing (cor.)	Y	*	Agarwal et al. (2022)	[5]
				R; C; T	Values outside the acceptable range	Continuous	ML/DM, Evolutionary Computation	Optimizing (cor.)	Y		Padella and de Leoni (2023)	[111]
				All available	Probability of a negative outcome above a threshold, intervention cost, resource availability	Discrete	ML/DM, Causal Inference	Optimizing (caus.)	Y		Shoush and Dumas (2021)	[131]
			Various process decisions	D; C	Highest predicted metric value	Continuous	ML/DM, Logic-Based models	Optimizing (cor.)	Y		Ghattas et al. (2014)	[44]
			Temporal relations among activities that have to be preserved or violated	T	Temporal logic-based prioritization of recommendations	Discrete	ML/DM, Logic-Based models	Optimizing (cor.)	Y		Donadello et al. (2023)	[31]
Both	Any	Control flow	Specific activity to perform	T; D; C	User-selected policy based on the provided cost-benefit analysis	Discrete	ML/DM, Causal Inference	Optimizing (caus.)	Y		Bozorgi et al. (2023b)	[15]
			Next activity	C; D	Path scoring based on goal fulfillment, cost considerations, and knowledge worker-defined input configuration	Continuous	ML/DM, Logic-Based models	Optimizing (cor.)	Y	*	Seidel et al. (2024)	[127]

Figure 6. Framework overview: optimizing process outcome objective, and both. [Enlarged version of this figure in Appendix A]

The framework's primary characteristic is the Objective. Our analysis reveals that the identified techniques can be classified into two categories based on the objective (see Section 3.2.2). The first category focuses on reducing the percentage of cases with a negative outcome, thereby optimizing process outcomes. The second category aims to enhance process efficiency, measured using a quantitative performance metric, such as cycle time.

The next characteristic is the Target, which represents the metric used to evaluate whether performance improves after a prescribed recommendation. For techniques targeting process outcomes, the target might involve counts of categorical case outcomes (e.g., customer complaints) or temporal outcomes (e.g., deadline violations). In contrast, quantitative performance targets include metrics (e.g., cycle time, defect rate, cost).

The subsequent two characteristics are Recommendation Perspective and Recommendation. The Recommendation Perspective specifies the process aspect re-

Objective	Target	Rec. perspective	Recommendation	Input perspective	Policy	Rec. frequency	Prescriptive analytics method	Rec. purpose	Detail. algorithm	User eval./ UI	Reference	Number ref.
Optimizing process efficiency	Cycle time	Resource	Resource for next activity	C; T	Highest predicted resource performance	Continuous	Probabilistic Models	Guiding	Y		Wibisono et al. (2015)	[152]
			Resource allocation	C; R	Minimal cost, maximum flow	Continuous	ML/DM	Optimizing (cor.)	Y		Park and Song (2023)	[116]
	Processing time			R; C; T	Scheduling algorithm	Continuous	ML/DM	Optimizing (cor.)	Y		Park and Song (2019)	[115]
			Resource for next activity	R; C	Highest predicted resource performance	Continuous	ML/DM, Logic-Based models	Optimizing (cor.)	Y	*	Kim et al. (2013)	[72]
				T; D; R	Task duration predictions based on resource experience and workload distribution considerations, minimizing the Time-Workload coefficient	Continuous	ML/DM, Logic-Based models	Optimizing (cor.)	Y		Padella et al. (2024)	[112]
	Cycle time	Control flow	Next activity	C	Maximum metric improvement	Continuous	Logic-Based Models	Guiding	N		Nakatumba et al. (2012)	[101]
	Defect rate		Set of activities	D; C	Similarity based	Continuous	Statistical Analysis	Guiding	Y	*	Yang et al. (2017)	[144]
			Next activity	C; R	Risk prediction based on historical data, scheduling algorithm	Discrete	ML/DM, Mathematical Programming	Optimizing (cor.)	Y		Conforti et al. (2015)	[27]
	Cost		Specific activity to perform	C; T; D	Risk prediction based on historical data	Discrete	ML/DM	Optimizing (cor.)	Y	*	Dees et al. (2019)	[30]
	Varies (multiple)		Next activity	C	Set of rules	Discrete	Logic-Based Models	Guiding	Y		Haisjackl et al. (2010)	[48]
				C	Optimal policy computed by using Monte Carlo methods	Continuous	ML/DM	Optimizing (cor.)	Y		Branchi et al. (2022)	[16]
			Best next path	D; C	Maximum metric improvement	Discrete	ML/DM, Probabilistic models	Optimizing (cor.)	Y		Khan et al. (2021)	[71]
		Varies (multiple)	Resouce allocation, additional resources, adjusting case attributes	C; R	Temporal patterns of operational constraints	Discrete	Logic-Based Models	Guiding	Y		Park et al. (2023)	[114]
			Multi-activity recommendations (process schedule)	C; R; T; D	External power price forecasts, real-time data on power consumption, and process-specific constraints	Discrete	Mathematical Programming	Guiding	Y	*	Hermann et al. (2024)	[52]
			Next activity and resource	C; R	Set of rules (optimized plans, resource availability)	Continuous	Mathematical Programming	Guiding	N		Barba et al. (2012)	[8]
	Cycle time	Varies (multiple)	Next activity and resource	R; C; D	Similarity based	Continuous	Probabilistic Models	Guiding	N		Nezhad et al. (2011)	[104]

Figure 7. Framework overview: optimizing process efficiency objective. [Enlarged version of this figure in Appendix A]

lated to the recommendation, such as resources or control flow. Additionally, a "varies" category is included for techniques that produce different recommendations. The Recommendation characteristic lists the specific prescribed recommendations (see Section 3.2.3), such as assigning a resource to an activity.

Three next characteristics define the data inputs (Input Perspective, Detailed Input), and policies (Policy) for triggering interventions. The Input Perspective describes the required input data types for a technique (see Section 3.2.4). The elicited categories are (C)ontrol flow (e.g., activities and their sequences), (R)esources (e.g., activity performers), (T)emporal features (time-related data), and (D)omain-specific features (process- or domain-dependent data). The Detailed Input characteristic explains what the detailed input is, such as resource data is experience, performance, or workload (for space reasons, visible in the full version of the framework in supplementary material).

Policy refers to policies that help to define in which cases to prescribe a recommendation (see Section 3.2.5). For example, if the case is predicted to exceed a metric limit, then a recommendation will be prescribed in it.

The characteristic Recommendation Frequency identifies whether the method prescribes actions continuously (at every step) or discretely (only as needed). Additionally, Recommendation Purpose specifies whether a method is optimizing or guiding and, if optimizing, whether it relies on correlation or causality (see Section 3.2.3). This is in connection to the Prescriptive Analytics Method (described in the same section).

Another characteristic is the level of detail provided about the utilized algorithms, categorized as Detailed Algorithm. Papers offering sufficient detail for reproducing the method or algorithm, including step-by-step explanations, are marked with a “Y”. Conversely, those that only reference methods without providing details are marked with an “N”.

Finally, the User Evaluation characteristic marks those papers that conducted a user study to evaluate the technique or propose a user interface (see Section 3.2.6).

3.3. Discussion

The presented framework provides an overview of existing PrPM techniques. The overall review, however, also unveils several gaps and associated implications for future research.

First, the review highlights that the majority of techniques in PrPM focus on improving processes from a temporal perspective (e.g., cycle time, processing time, deadline violations [45, 115, 72, 150]) (RQ1.1). In contrast, other performance dimensions are considered in only a limited number of studies (e.g., defect rate in [155, 27], cost in [30]). Thus, a potential avenue for future research could involve the exploration of additional performance objectives that could be enhanced through PrPM.

We note that techniques differ in terms of the nature of their prescriptions (RQ1.2). As the review highlights, there exists a body of techniques that focus on providing guidance to users during process execution (e.g., [104, 114, 101]), which are commonly based on the similarity between the ongoing case and previous executions of the same process. Conversely, other techniques are specifically designed to optimize processes according to a performance measure (e.g., [71, 95, 15]). Thus, techniques vary between those that guide users and those that offer specific recommendations for actions to be taken next. Future research can distinguish between these two types of techniques and the corresponding recommendations they aim to develop. In terms of optimizing techniques, there are two further categories. One way is to consider the correlation of a predicted value with the likelihood of it having a positive effect on the running case (e.g., [45, 71]). Another way is to focus on causality, where the impact of the proposed recommendation on the outcome of the case should be estimated, for example, through building a causal estimation model [133, 13].

The review also indicates that the majority of techniques focus on identifying cases in which recommendations should be applied and determining the appropriate moment for triggering a recommendation during the execution of a case. In contrast, there is limited attention given to the problem of identifying *which* recommendations can be applied to optimize a process in relation to a performance objective. Discrete techniques often leave it to the users (stakeholders) to define possible recommendations a priori (e.g., [84, 30]). We observed that techniques utilizing observational event logs from BPI challenges rely on winner reports (i.e., the best report selected among the submissions to the challenge) to identify potential recommendations (e.g., [131]). This means taking the actions that the winner reports suggested to take in the process to improve it after having analyzed it as recommendations, for which the effect of the PrPM technique is to be applied in the case. In contrast, continuous techniques focus on recommending the next task(s) (e.g., [101, 127, 155]) or resource allocation (e.g., [72, 111, 116]). One direction for future research could involve the development of techniques capable of discovering candidate recommendations from business process event logs, textual documentation, or other unstructured or structured process metadata.

We observe that the majority of studies tested their techniques with synthetic and/or real-life event logs (RQ1.3). Therefore, the validation is done using a real-world or synthetic *observational* event log, but crucially it is not done in real-life settings. An attempt to test the effectiveness of recommendations in real-world contexts was made by Dees et al. [30], whose study demonstrated that although predictions were relatively accurate, the recommendations did not yield the desired outcomes. Agarwal et al. [5] are planning to integrate the proposed technique into the IBM tool suite, but the testing with the end-users is yet to be done. Therefore, it is crucial that the proposed techniques be validated in real-life settings to confirm their practical applicability.

Another gap in existing research concerns the design and tuning of policies for PrPM (RQ1.4). Existing optimizing techniques that rely on correlations (e.g., [37, 95]) prescribe recommendations when the probability of a negative outcome exceeds a predefined threshold. However, because predictive models are based on correlation rather than causal relationships, the prescribed recommendations may not address the underlying cause of a negative outcome or poor performance (e.g., the cause of a delay). More attention has been paid to accounting for causality when designing policies in studies published in the past three years than before (e.g., [133, 134, 13, 14]). Therefore, an avenue for future research could involve the development of policy design techniques that incorporate causality.

As discussed in Dees et al. [30] and de Leoni et al. [84], the decision of whether or not to apply a recommendation, or which recommendation to apply, often depends on contextual factors. Some recommendations may prove ineffective or even counterproductive due to second-order effects (i.e., a recommendation improves the target metric but leads to an unintended effect on a case or other cases). For instance, a recommendation involving proactive customer contact to prevent complaints may inadvertently increase the likelihood of a complaint [30]. Similarly, assigning a resource to a case running behind schedule might result in other cases being neglected, thereby causing delays elsewhere and increasing the overall ratio of delayed cases. Detecting such second-order effects requires human judgment and iterative policy validation, such as A/B testing [75]. In this context, it is noteworthy that only a handful of current PrPM techniques address the need for interaction with human decision-makers (e.g., allowing the end-user to select a policy for the technique [134, 15]). Furthermore, there is little to no attention to applicability of the developed techniques in practice. As such, we found a mention of a user evaluation or a UI only in [155, 60, 5, 127, 72] (RQ1.5). This is problematic since without a way to present the outputs of PrPM techniques to the end users, it is challenging to obtain feedback on their usefulness and usability.

Another step toward improving PrPM techniques for the end-user usage involves the ability to explain why a recommendation is given. This explanation encompasses two aspects: first, explaining the prediction underlying a prescription (prediction explanation), and second, explaining the policy used to trigger the prescription (policy explanation). Therefore, one potential direction to improve the practical applicability of PrPM techniques is to integrate mechanisms for explainability. Several techniques have incorporated explainability modules [15, 134, 5]. However, still, most of the current techniques lack mechanisms to clarify why a recommendation is given for a specific case and state. In summary, existing techniques focus on addressing functional requirements (i.e., training models for PrPM and using these models at runtime to generate recommendations). As to the non-functional requirements, almost every study provides a way of comparison with other techniques, for example, using the dimensions of computational efficiency, scalability, accuracy, and optimality (maximizing a gain or utility function). There is, however, relatively little work on user-centricity, including usability (e.g., fulfilling user information needs) and explainability.

Finally, the review underscores the lack of a standardized terminology in the field, which may be attributed to the novelty of PrPM as a research area. Authors use a wide variety of terms to describe similar concepts. For example, terms such as “proactive process adaptation” [95], “process prescriptive analytics” [111], “action engine” [114], and “best-next activities” [16] or “best-next actions” [127] are all used to refer to the development and application of PrPM techniques. This inconsistency in terminology highlights the need for a unified lexicon in the field.

3.3.1. Implications

In summary, the following research directions are identified:

1. Validating techniques in real-life settings to ensure their practical applicability: The majority of PrPM techniques have been tested using synthetic or real-life event logs, but not in real-life settings. It is essential for future studies to validate techniques in actual operational environments to ensure practical applicability.
2. Designing techniques to support the discovery of recommendations from event logs and evaluating their potential effectiveness: Many techniques focus on identifying when recommendations should be applied rather than on discovering which recommendations can optimize processes. Future research should explore techniques that can automatically discover candidate recommendations from business process event logs, textual documentation, or other process metadata.

3. Developing policy design techniques that incorporate causality: Existing correlation-based techniques may prescribe recommendations that do not address the underlying causes of negative outcomes. Research should focus on developing policy design techniques that account for causality to improve the effectiveness of recommendations.
4. User-centricity: Although much focus is placed on improving the accuracy and efficiency of PrPM techniques, there is limited attention to non-functional aspects like usability and usefulness. Future research should emphasize improving these aspects to enhance the overall user experience with PrPM techniques.
5. Improving the explainability of PrPM techniques: Most PrPM techniques do not provide explanations for why specific recommendations are made. Future research should focus on incorporating prediction and policy explanations to enhance the transparency and trustworthiness of the techniques.
6. Investigating additional performance objectives beyond temporal ones: Most techniques primarily focus on improving temporal performance (e.g., cycle time, processing time). Additional performance objectives, such as defect rates or cost reduction, that could benefit from PrPM should be explored.
7. Developing a common terminology in the field: The field lacks a standardized terminology, with authors using a variety of terms to describe similar concepts. There is a need for a unified lexicon to improve clarity and consistency in the field of PrPM.

3.3.2. Limitations

The SLR methodology is subject to certain limitations and threats to validity, as discussed by previous researchers [73]. First, there is a potential risk of overlooking relevant publications during the search process. We mitigated this risk by conducting a two-phase search that included a broad range of key terms, as well as backward referencing. Another potential threat arises from the exclusion of relevant publications during the screening phase. To address this, we employed explicitly defined inclusion and exclusion criteria. Additionally, all cases of uncertainty were examined and discussed with the co-authors. Third, data extraction bias may pose a threat, as this step inherently involves some level of subjectivity. We minimized this risk by discussing the papers in the final list and refining the data extraction process as necessary to ensure accuracy and consistency.

3.4. Summary

In this chapter, we addressed $\mathbf{RQ}_1$ *What are the outputs of PrPM techniques?*. To do that, we employed the systematic literature review methodology [73] to review the field of prescriptive process monitoring. As a result, we introduced a framework for categorizing the techniques in this domain. The framework classifies existing techniques according to their objective, target metric, type of recommendations, technique, data input, policy, and user evaluation aspect used to trigger recommendations.

Our findings reveal that existing PrPM techniques primarily aim to achieve one of two objectives: optimizing process outcomes or optimizing process efficiency. To achieve these objectives, a variety of recommendations can be prescribed, with a predominant focus on the control flow and resource perspectives. Additionally, recommendations can be categorized based on recommendation frequency (i.e., when recommendations are prescribed) and recommendation purpose (i.e., how recommendations are prescribed). Regarding recommendation frequency, techniques can be classified as continuous (prescribing recommendations for multiple or all activities of an ongoing case) or discrete (prescribing recommendations only when a need is detected). With respect to recommendation purpose, recommendations can either be optimizing or guiding. Optimizing techniques provide recommendations to improve an ongoing case with respect to a specific KPI or performance measure. These recommendations may be based on either correlation or causality. In contrast, guiding techniques provide recommendations based on other prescriptive analytics methods, for example, analysis of historical traces.

Our review further demonstrates that PrPM techniques typically rely on control flow, resource, temporal, and domain-specific data as input. The policies used to prescribe recommendations vary based on the type of technique. Guiding technique generally use similarity-based policies, while optimizing techniques employ a range of policies, including sets of rules, maximum metric improvement, and thresholds for the probability of negative outcomes.

We also identified key research gaps and directions for future research to address the evolving landscape of prescriptive process monitoring. In particular, our work highlighted: (i) a lack of *in vivo* validation of the proposed techniques; (ii) a lack of techniques for discovering suitable recommendations and assessing their potential effectiveness; (iii) little emphasis on user-centricity to ensure the usefulness and usability of the techniques, (iv) explainability and feedback loops between a PrPM system and its end-users; and (v) a narrow focus on temporal metrics and comparatively little work on applying PrPM to other performance dimensions.

4. DEVELOPMENT CYCLE I: OBJECTIVES DEFINITION, PRE-DESIGN & EVALUATION

This chapter corresponds to the phases *Problem Identification, Objectives Definition*, and *Development Cycle I* in Figure 1. The purpose of this cycle is to answer $\textbf{RQ}_2$ *What user groups could benefit from working with the outputs of PrPM techniques? What are the users' information needs?* To do so, we analyze the PrPM domain (based on the review in the previous chapter) and existing tools to elicit an initial set of information items for a PrPM interface. We then create a wireframe and conduct an expert evaluation. The purpose of the wireframe is to evaluate the information items. The evaluation helps to elicit end-user groups for PrPM and refine the information items for each of them.

Section 4.1 describes the process of *Objectives Definition*. In it, we elicited the initial set of information items to be included in the interface for PrPM outputs. In Section 4.2, we describe the *Design & Development* and *Evaluation & Refinement* of the wireframe for PrPM outputs. We discuss the results in Section 4.3 and conclude the chapter in Section 4.4.

4.1. Objectives Definition: Information Items Elicitation

In this section, we describe the method applied to elicit the information items. Section 4.1.1 introduces the method, and Section 4.1.2 the results.

4.1.1. Method

To elicit the initial set of information items to be included in PrPM interface, we conducted an analysis of tools and the PrPM domain. By *tools*, we refer to commercial process mining tools and academic papers that include an interface for PrPM functionality. The *domain* analysis was based on the outputs of Chapter 3.

Tools Analysis

We analyzed commercial process mining tools and academic solutions. We started with reviewing the commercial tools surveyed in [144].[6] The survey compared 17 tools across seven categories, e.g., "Operational Support", "Conformance Checking". In the category "Operational Support", the survey listed the criterion "Recommendations (prescriptive analytics)". According to the survey, only Celonis has prescriptive functionality. However, the survey was published in 2020. Therefore, we reviewed the other listed tools manually since new features could have been added since then. Thus, we also added ABBYY Timeline and SAP Signavio to the analysis. ABBYY Timeline reports on providing prescriptive analytics on their platform[7], and SAP Signavio has actions functionality

[6] `http://www.processmining-software.com/`

[7] `https://www.abbyy.com/timeline/process-prediction-and-simulation/`

similar to that of Celonis.[8] As to the academic solutions, we examined the studies identified in Chapter 3. Namely, we included studies that proposed an interface for their developed PrPM technique. As a result, we included [155] and [60].[9] In total, five interfaces with some PrPM components were included in the analysis.

We analyzed the identified tools using visual analytics framework [100]. This framework allows for analysis of visualization across three dimensions. Specifically, we analyzed how each tool answers the questions of "What?" (refers to items and attributes), "Why?" (refers to performed task, usually expressed as a verb and a noun), and "How?" (visualization elements). For example, one tool [155] presented a statistical analysis (what) to explain the calculation (why) using a heatmap (how).

We clustered the results based on the tasks ("whys"), and arrived at four main tasks: (1) Describe case, (2) Describe recommendation, (3) Explain recommendation, and (4) Assign resource. These tasks served as the base for groups of information items to be included in the interface. For example, we refined the first task into a group of information items "Case Description" which includes information items related to describing the ongoing case.[10]

Domain Analysis

In this step, we drew conclusions on the capabilities of PrPM techniques based on the findings of Chapter 3. In order to synthesize the review, we created a UML domain model for prescriptive process monitoring. Thus, the model served as an additional source of input for the information items to be included in the interface.

4.1.2. Results

Tools Analysis

Based on the tools analysis, we elicited four groups of information items. These are (1) Describe case, (2) Describe recommendation, (3) Explain recommendation, and (4) Assign resource. On this basis, we formed information items groups. Respectively, group *(i) Case Description* corresponds to the first task of case description and includes items that describe an ongoing case such as case attributes, e.g., requested amount for a loan, or the Key Performance Indicator (KPI) of the process. Group *(ii) Recommendation Description* refers to information about the prescribed recommendation, such as its type. The third group, *(iii) Recommendation Explanation* includes information items with details on how the recommendation was calculated, e.g., description of the ML model. Finally, group *(iv) Resource Assignment* comprises information items relating to assigning a suitable resource to the prescribed recommendation, such as resource availability.

[8] `https://www.signavio.com/products/process-intelligence/`

[9] Although Chapter 3 brings examples of three more UIs for PrPM, this study was originally conducted before the additional UIs were added to the body of knowledge.

[10] The analysis overview can be found at: `https://doi.org/10.6084/m9.figshare.21629615.v1`

Domain Analysis

Our analysis of existing tools enabled us to understand what data is presented to end-users and how it is presented. To ensure that the information items are complete, we extended them with the domain model of PrPM derived from the review in Chapter 3. In the center of the domain model is the PrPM *technique*. The technique uses an *event log* produced by the *business process*. The PrPM technique has an *objective* which is detailed by the *target* (for example, the objective is to avoid a negative outcome (binary), and the specific target is avoiding the deadline violation [157]; the objective is optimizing the process efficiency, and the target is cost [16]). Utilizing a a *policy* (for example, a recommendation is triggered when the probability of a negative outcome exceeds the defined threshold [37]), the PrPM technique produces a *recommendation*. There are three types of recommendations: guiding, correlation- or causality-based. The recommendation also has two attributes: *frequency* and *perspective*. In terms of frequency, the recommendation can be continuous or discrete (for example, a discrete recommendation to contact the customer with additional offers is given only when the need is detected [15], and a continuous recommendation on each step is given for the best next activity in a case [127]). The recommendation can be given from a resource or control flow perspective, or a combination.

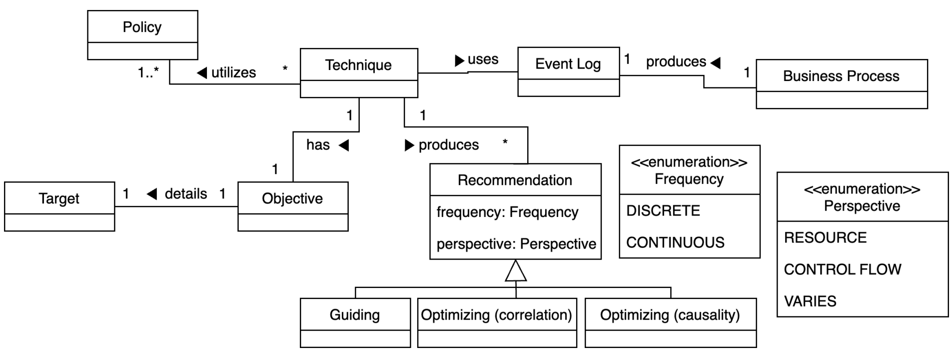

Figure 8. Domain model for prescriptive process monitoring (based on Chapter 3).

The information items based on the domain model were added to groups (i) and (ii). To *(i) Case Description*, we added the Domain-related information (that comes from the input data), and Objectives (based on the objective and target). In the *(ii) Recommendation Description* group, we added Type (guiding, correlation- and causality-based), Recommendation perspective (e.g., control flow), and Recommendation frequency (discrete or continuous). Policy is a technical attribute that describes how the technique works but does not directly relate to the output.

The elicited items are summarized in Figure 9. The final column in Figure 9 (Information Items) presents the outcome of the analysis above, showing how specific elements from both the tools analysis (marked with T) and the domain analysis (marked with D) were synthesized. For each information items group (e.g., Case Description, Recommendation Description), we identified relevant data elements mentioned either in the domain analysis (e.g., objectives, recommendation perspective), and cross-referenced these with the concepts extracted from the tool analysis (e.g., affected KPI, recommendation name). The resulting information items also link to the specific element in the wireframe (number in brackets).

Insights from Tools Analysis	**Insights from Domain Analysis**	**Information Items (Wireframe Mapping)**
(i) Case Description (view #2)		
(T1) Process visualization		Process model (2.2)
(T2) KPI-dependent performance data	*(D1)* Domain-related information	KPI-status, Case-specific attributes (2.1, 1.2)
(T3) Affected KPI	*(D2)* Objectives	
(T4) Case details		
(ii) Recommendation Description (view #3)		
(T5) Recommendation name	*(D3)* Type (recommendation purpose)	Type (3.2)
	(D4) Recommendation perspective	Process aspect (3.3)
	(D5) Recommendation frequency	Frequency (3.1)
(iii) Recommendation Explanation (view #3)		
(T6) Remaining time, impact, statistical analysis		Prediction description (3.4)
(T7) Confusion matrix, model accuracy, attributes contribution		Calculation description (3.5)
(iv) Resource Assignment (view #4)		
(T8) Assignee		Resource availability (4.1), Last update (4.2), Role (4.3)

Figure 9. Information items: initial set. The codes with T stand for tools analysis, and D for domain analysis. View numbers and information items numbers correspond to the wireframe in Figure 10.

4.2. Pre-Design & Evaluation

In this section, we describe the method applied to create the pre-design of a UI for PrPM and evaluate the included information items. Section 4.2.1 introduces the method, and Section 4.2.2 the results.

4.2.1. Method

Wireframe Design

Using Figma[11], we created a simple wireframe. Creating a wireframe provided a way to represent the elicited information items in a context, and it thus served as a pre-design of the PrPM interface. Wireframes allow for a focused exploration of content prioritization and layout without being influenced by visual design elements [49]. By testing and refining the wireframe early, it becomes possible to

[11] We utilized a free wireframe kit `https://bit.ly/3ERovZU` to design the wireframe.

identify gaps in the information provided and ensure that the required information is available for users [49]. We used a loan application process to exemplify included information items in the wireframe since it is one of the better-known and widely applied event logs in the process mining community.

Wireframe Evaluation

Participants. We conducted an evaluation of the wireframe with 13 process mining experts, particularly focusing on assessing the relevancy and usefulness of the information items. We recruited the participants from consultancies and companies that conduct process mining projects (Table 6). Consultants and internal process mining analysts can be expected to have an overview of end users' needs since they work with different stakeholders [98]. Our participants had on average seven years of experience with process mining. Ten participants were from the consulting domain and three from product companies that engage in process mining projects internally.

Table 6. Wireframe evaluation interviews participants.

Code	Domain	Experience
I-01	Real Estate	12 years
I-02	Banking, Finance Services, Insurance	7 years
I-03	Consulting (Process Mining & Data Analytics)	2 years
I-04	Consulting (Process Mining)	8 years
I-05	Consulting (Process Mining & Data Science)	5 years
I-06	Consulting (Process Mining)	7 years
I-07	Consulting (Process Mining)	7 years
I-08	Consulting (Process Mining)	9 years
I-09	Consulting (Process Mining)	18 years
I-10	Consulting (Process Mining & BPM)	2 years
I-11	Online Retail	4 years
I-12	Consulting (Process Mining)	6 years
I-13	Consulting (Process Mining)	6 years

Data Collection. We conducted semi-structured interviews with the participants. We created an interview guide based on the $\mathbf{RQ}_2$ of understanding the information needs of the end users. We also aimed to give the participants the opportunity to discuss their own perspective [50].

At the beginning of the interview, we demonstrated the wireframe to the participant and described the visualized information items using the example of the aforementioned loan application process. During the demonstration, we explained that in the given example, the user would be able to get an overview of ongoing loan applications and recommendations for each application. Using the information displayed, the user would be able to decide whether to follow the recommendations or not.

Then, we asked the participant to think of a recent project where they could use a similar interface. With this, we ensured that the participants could talk about a specific situation instead of providing scattered opinions on different projects. After the demonstration, we asked the participants three questions based on $\mathbf{RQ}_2$. Specifically, the first question dealt with evaluating the relevancy of each information item to the described task of reviewing and deciding on recommendations in ongoing loan applications. To this end, we asked: *"Which information do you find most/least relevant and why?"* Next, we were interested in the usefulness of each information item: *"Which information do you find most/least useful and why?"* In the last question, we asked the participants to reflect on information items which they could imagine could be helpful but were not included in the interface. Specifically, we asked: *"What information do you think is missing?"* Wherever the participant mentioned that an information item was not as they expected (e.g., not relevant, not useful), we also asked whether they had a suggestion on how it could be improved. The full interview guide is available in Appendix C. The interviews lasted between 14 and 25 minutes and were recorded.

Analysis. We transcribed the interviews using an online transcription tool[12] and manually corrected the transcripts. We analyzed the interviews using a combination of deductive and inductive coding.

First, we familiarized ourselves with the transcripts and created the initial set of codes based on our research objective. Thus, the codes related to the elicited information items, for example, "Effect of recommendation" for items in the group Recommendation Explanation, and "Process model is relevant" to an item relating to a process model in Case Description. Upon conducting the first round of coding, we derived additional codes other than those relating to the information items. For instance, interview parts that described information items being relevant for different users or some items being suitable exclusively for one role were marked with "End user". Thus, we expanded the coding scheme. We then reiterated the coding procedure.

Finally, we conducted thematic analysis [17], and clustered the codes into themes. In total, we identified seven themes: four related to the information items groups (e.g., "Case Description", "Recommendation Explanation"), and the remaining three captured other codes that emerged during the first coding round. These three themes were: "End User" (included the codes about information needs of different user groups), "Cases Prioritization" (related to the importance of prioritizing ongoing cases based on a criterion), "Overview (Multiple Cases)" (captured the comments about the page with overview of ongoing cases).[13]

[12] `https://otter.ai/`

[13] The full coding scheme can be found at: `https://doi.org/10.6084/m9.figshare.21629615.v1`

4.2.2. Results

Wireframe Design

We designed a static wireframe based on the elicited information items (Figure 9). As an example for the wireframe, we took a loan application process with attributes such as requested loan amount, purpose, and applicant. Thus, the wireframe shows ongoing cases – loan applications – and recommendations for them.

When designing the wireframe, we followed the visuals present in many existing process mining tools, such as a layout with tabs cases and resources. The wireframe consisted of four views (see Figure 10). View #1 showed an overview of ongoing cases. From this view, the user can gather information about ongoing loan applications, such its case-specific attributes and current status (which corresponds to information items listed in Figure 9). In addition, the user sees an indication of existing recommendations in a case at a given point in time. On view #2, information about an individual case is provided (group Case Description). Thus, from this view, the user can see the details about the case and explore the process model for the case. On view #3, information items included in group Recommendation Description are presented. Namely, the details about the type of recommendation are specified, as well as which process aspect the recommendation relates to (e.g., resources, control flow). Last, on view #4, information items related to group Resource Assignment are included. The user can see information related to availability, last update, and role of resources that could be assigned to act on a recommendation.

Wireframe Evaluation

According to the evaluation findings, the majority of the participants (8/13) found the information items presented in the wireframe suitable for different end-user groups. For example, as commented by an interviewee, *"But what I would really define upfront is, who is going to be your consumer of the information. Because for me, these are different levels of abstraction."* (I-04). Another interviewee similarly mentioned that, *"I was thinking, am I looking at this strategically or operationally? What I'm seeing might change on that basis."* (I-10). The participants highlighted that some wireframe views could be more relevant for one kind of user (e.g., a manager), while others could, suit the needs of users who process cases on daily basis. As such, the evaluation findings indicate that the wireframe could be used by three distinct groups of end users: operational worker, operational manager, and process analyst. Specifically, we define an operational worker as someone who directly processes cases, such as a clerk at an insurance company or a loan application specialist at a bank. Next, an operational manager is responsible for several operational workers and is interested in resource allocation. Last, a process analyst analyzes open and complete cases to find opportunities for improvement in the entire process. In the following, we detail the findings for each of these end-user groups.

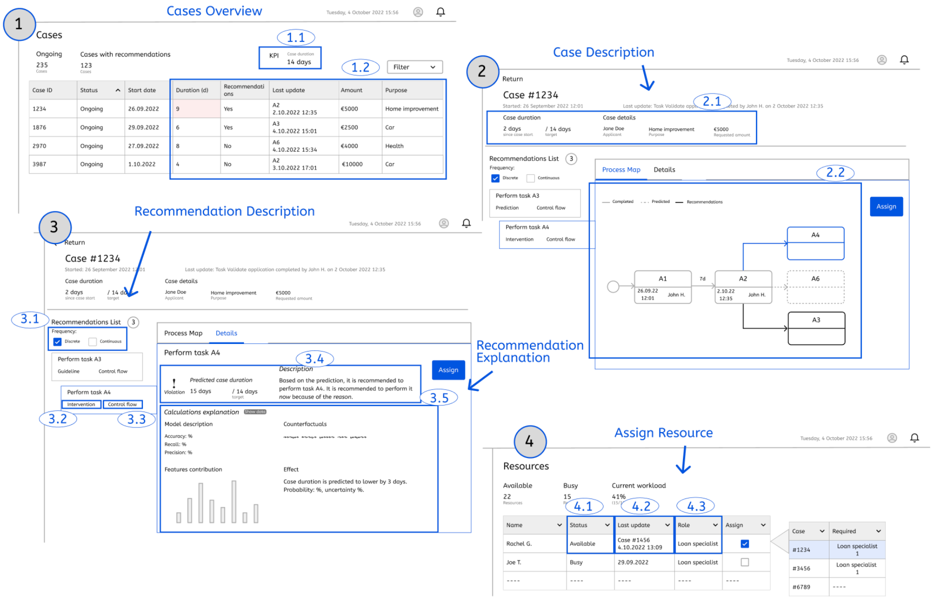

Figure 10. Initial wireframe for prescriptive process monitoring interface. View numbers and information items numbers correspond to Figure 9. [Enlarged version of this figure in Appendix B]

Operational Worker. Interview participants shared an opinion that information items currently presented in the wireframe would be overwhelming for the operational worker and distract them from their main task, i.e., case processing. As one interviewee put it, *"They [operational workers] always need to stay focused, otherwise, they are lost."* (I-06). Another interviewee shared that, from their experience, operational workers have a pace of completing task after task and should concentrate on them: *"something like this [looking at all views] would be getting out of this flow."* (I-01).

As to the relevancy of presented information items, seven participants found the recommendation explanation (Figure 10, view #3) to be too complex for operational workers. Specifically, the interviewees shared that ML model details are important for analysts with technical background but are not necessary for operational workers: *"And a huge level of detail of why such a recommendation is done [...] is interesting for someone that's doing process mining, but for someone that's doing operational work, it can be overwhelming."* (I-11). Another interviewee explained that in their experience, *"...talking about the prediction accuracy, these people [operational workers] don't have any clue about that."* (I-13). Nevertheless, the majority of the participants (9/13) agreed that it would be important for a decision-maker to understand the effect of a decision on the ongoing case. As one participant put it, *"I think it would be much more about knowing the outcome of the action rather than suggesting the action."* (I-08). Another information item the interviewees (4/13) found to be not relevant for the operational worker is the process model. One participant commented that *"you can use it as a kind of a history, but as being the clerk, I'm not sure what that adds to you"* (I-04).

The interviewees also suggested adding additional information items. Half of the participants proposed including statistics on how often the same recommendation had been acted upon in the past. Specifically, an operational worker would be able to examine what outcomes were achieved in previous similar cases. According to one interviewee, such information would help evaluate *"if this is completed, did it help? If I follow the recommendation, did I achieve what was predicted, what was in the recommendation as the effect?"* (I-06). Another suggestion by a half of participants was to make the explanation details available on demand. As one interviewee suggested, *"And if s/he [user] wants to, s/he can drill down to the information about why is this recommendation [proposed]. So drill down should be optional. Not mandatory for the whole process."* (I-02).

The participants also suggested adding indication which cases operational workers should work on based on the KPIs (Figure 10, view #1). One interviewee shared that seeing the cases, a few questions would arise, such as *"How many cases do I need to action on? What are the cases that are most important? Because that gives me an idea of my workload."* (I-11). Another participant commented that having such an indication would help the user understand the tasks at the specific point in time: *"So I would need to have a short list that 'Okay, this is today 9am, what are the most important topics that I need to look at the most."'* (I-06). In addition, the participants found it important to have the cases in the overview prioritized. The majority (8/13) thought that the cases should be prioritized based on the process objective, so that the users would be presented with cases they need to tackle to achieve it. As one participant put it, *"everything which is one day late can maybe still be on time if we pick it up now"* (I-07).

In summary, according to the experts we interviewed, the most relevant information items for operational workers are the predicted effect of a recommendation and evidence of its effectiveness based on past similar cases. Additionally, the explanations about recommendation calculation should be presented on demand. Finally, cases should be prioritized based on the process objectives, and the operational worker should have an overview of cases assigned to them.

Operational Manager. The information about resource assignment (Figure 10, view #4) was found by our interviewees to be most suitable for a operational manager. According to the interviewees, a operational manager is the person interesting in allocation of resources. As explained by one interviewee, *"If I'm in charge of this team, I just want to make sure that these guys [operational workers] are working on the best next case, and taking the next best action."* (I-10).

On information items for resource assignment, four participants shared that resource performance and workload would be required. As per one interviewee, such information would enable the manager to allocate the resources to certain types of cases: *"If I have several loan specialists, how quickly are they typically processing a car loan versus a home improvement? So, can I get a KPI for that* *specific case that tells me: this person is typically better in handling these type of cases."* (I-04). Similarly, another participant shared that such information could also represent different teams to *"see how many open cases per team member or per department, or is there a significant difference between my teams."* (I-08).

According to the participants, a process model would not be relevant for operational managers and operational workers. One interviewee gave the following reasoning: *"I think that this part [process map], although very nice to have for analytics, is not necessary for guys who are relocating resources and doing day-to-day jobs."* (I-02). Another information item that, according to the interviewees, operational managers could struggle with is the characteristics of different recommendations. As one participant put it, *"A business guy would say, 'Discrete or continuous? Good question. Let me mark one of those.' So it's not really business oriented. Unless you are a mathematician, then you know, maybe."* (I-09).

In conclusion, the experts we interviewed found the information related to resource assignment to be the most relevant for the operational manager. In addition, suggestions were made to include information on resources workload and performance. Finally, process model and information on recommendation characteristics were found to be not relevant for the operational manager.

Process Analyst. According to the interviewees, process analysts would be mainly interested in analyzing prescribed recommendations to elicit policies that improve the overall performance of the business process. One participant expressed that *"if the user is improving the processes, and s/he needs some kind of recommendation to improve it, it's the way to go."* (I-01). The participants shared that for this task, it would be useful to understand what different options exist: *"I would want to see what [...] is the prediction that's going to happen? And what should I do instead as a recommendation?"* (I-05).

A process model was found to be useful for process analysts (Figure 10, view #2). As one interviewee commented, *"this one [process map] will be interesting a lot, because I would say, 'Okay, this has been completed, here is predicted. And the recommendation, the blue one is the one assigned.' So I think this one [is] the one which would for me interest a lot, because I would know what is next."* (I-09).

In addition, details about recommendation calculation were perceived to help understand the background and thus, deciding on new policies. Unlike for an operational worker, for a process analyst it is relevant to learn about how the recommendations were given. As one interviewee put it, *"Because what I would also like to see what are the accuracies behind the model? Because one thing is [to say], 'Okay, this is the prediction and this is the recommendation.' But [...] I would also want to see, what is the number behind these?"* (I-05).

In summary, in the context of our study, a process analyst's aim would be to review the recommendations to find potential improvement opportunities in the overall process. Additionally, the most relevant information for a process analyst is a process map and details about recommendation calculation.

Information Items Groups	Information Item	OW	Example Quote	OM	Example Quote	PA	Example Quote
(i) Case Description	KPI status, case-specific attributes	x	*"So assuming I'm the loan specialist at the bank and handling cases. Then I would asked myself, what's my my KPI..."* (I-08)	x	*"So if I would be manager, I will have more high level KPIs, and [...] I would actually be interested a bit more and see how can I work towards overall goals of the company"* (I-08)	x	*"So how much time do I have left to be still able to make my goal. [...] And I need to know it pretty much in advance, or if I was late, that this has already happened, but maybe this you can use for the learning purposes for the models."* (I-06)
	Process model					x	*"this one [process map] will be interesting a lot, because I would say, `Okay, this has been completed, here is predicted. And the recommendation, the blue one is the one assigned."* (I-09)
(ii) Recommendation Description	List of recommendations and characteristics	x	*"I would rather give them [operational worker] three options and the predicted output of each of those options."* (I-04)	x	*"If I'm just in charge of this team, I just want to make sure that these guys are working on the the best next case, and taking the next best action."* (I-10)	x	*"a huge level of detail of why such a recommendation is done or what is the base, it is interesting for someone that's doing process mining, but for [...] operational work, it can be overwhelming."* (I-11)
(iii) Recommendation Explanation	Predicted effect	x	*"I think it would be much more about knowing the outcome of the action rather than suggesting the action."* (I-08)	x	*"So what can I achieve? How do I achieve it? [...] And by whom?"* (I-07)	x	*"I would want to see what are the implication or so what is happening here? What is the prediction that's going to happen? "* (I-05)
	Similar past cases and history of recommendations	x	*"If this is completed, did it help? If I follow the recommendation, did I achieve what was predicted, what was in the recommendation as the effect?"* (I-06)	x	*"I think it would be interesting to have a view on who was assigned for that [recommendation] and what was the real time for it. So my prediction was six days. But in the end, actually, if you look at the log file, it took two days."* (I-09)	x	*"You should be able to group it into clusters, like the one that didn't have this intervention, the one that did have this intervention, with similar timeframes, with similar duration, similar amounts, and what it ultimately meant."* (I-05)
	Model description			x	*"You want to get an understanding as well, how you got to the recommendation and how it came across."* (I-07)	x	*"Because what I would also like to see what are the accuracies behind the model?"* (I-05)
(iv) Resource Assignment	Resources and their performance			x	*"If I have several loan specialists, how quickly are they typically processing a car loan versus a home improvement?"* (I-04)		

Figure 11. Information items: refined after the wireframe evaluation (OW – Operational Worker, OM – Operational Manager, PA – Process Analyst).

We summarized the findings on the information items for each user group in Figure 11. The figure captures information items that were found to be relevant for each user group based on the experts' opinion, and the exemplary quotes.

To further support the understanding of why different information items are needed by each user group, we connect them to the goals that users aim to achieve with the help of PrPM systems. Our approach aligns with the conceptual modeling framework proposed by Nalchigar and Yu [102]. Specifically, the framework helps to connect the actors with their business goals, the decision goals they pursue, the question goals that underpin those decisions, and the insights provided by analytics systems to address those questions. Figure 12 provides such an overview for PrPM context. The overarching business goal is process improvement. As described in Section 2.1, to improve processes radically, one can use process redesign activities. For incremental change, the process is improved continuously [123]. PrPM contributes to the latter by enabling organizations to optimize processes at runtime. The main business goal can be split into business goals for each user group. For example, for the operational manager, the business goal is to ensure resource efficiency. For this, a decision they need to make is assigning cases to the most suitable resource. Their question goal informs that decision.

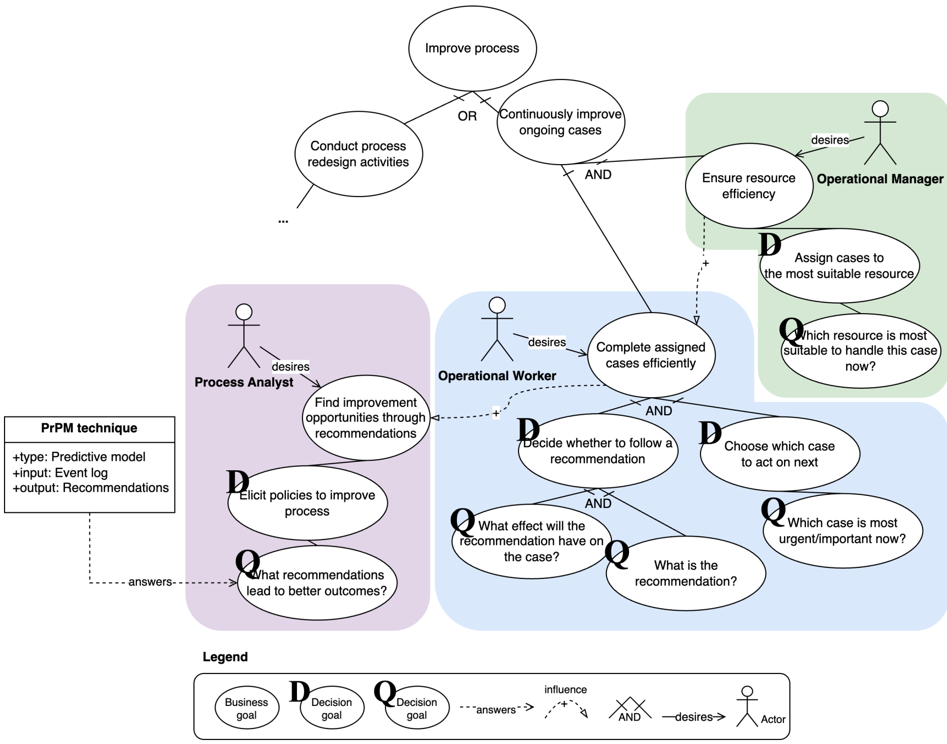

Figure 12. Conceptual overview of actors (users) within PrPM systems with their business, decision, and question goals.

In summary, each user group's business goals are decomposed into decision goals and question goals, which in turn guide their information needs. Table 7 provides the mapping of the information items to the goals.

4.3. Discussion

We identified the user groups of operational worker, operational manager, and process analyst that could benefit from the outputs of PrPM techniques. Each group has specific information needs, some of which are shared across the groups. For instance, understanding how addressing the recommendations influenced the process in the past was found to be useful for each user group. In contrast, some information needs appear to be distinct to one user group, such as resource information for an operational manager. Therefore, our evaluation indicates that the presentation of outputs of PrPM techniques should be adjusted to the information needs of the end users. This is in line with studies from other fields that found that presenting the end users with suitable information leads to quicker technology adoption and better decision-making [4, 26, 57].

Existing studies focus on what can be communicated based on the prescriptive technique rather than which user group to target or their information needs. For instance, in Bozorgi et al. [13], the authors present a tool that allows for discov-

Table 7. Mapping of users' question goals to insights from PrPM system and respective information items groups from Figure 11.

User	Question Goal	Insight from PrPM System	Information Items Group
Operational Worker	What is the recommendation?	Recommended action per case	Recommendation Description
	What effect will the recommendation have on the case?	Predicted outcome if the recommendation is followed	Recommendation Explanation
	Which case is most urgent/important now?	Cases to focus based on KPIs	Case Description
Operational Manager	Which resource is most suitable to handle this case now?	Resource workload and performance	Case Description, Recommendation Description, Recommendation Explanation, Resource Assignment
Process Analyst	What recommendations led to better outcomes	Performance of recommendations across cases	Case Description, Recommendation Description, Recommendation Explanation

ering and visualizing treatment rules that increase the probability of positive case outcomes, but the intended user is not discussed. Likewise, PrPM techniques that prescribe next actions (e.g., [155, 60]), also include information items that could be more suitable for process analysts. Thus, the effectiveness of PrPM techniques might be limited if the needs of the intended user group are not considered.

We also found that prioritizing ongoing cases can be also relevant and useful. In other words, when starting to optimize ongoing cases, it is not feasible to address all cases at once. Therefore, ongoing cases could be assigned a priority to help operational workers determine which case to work on next. Prioritization can be based on process objectives or specific organizational criteria.

Additionally, each user group has an information item of estimating the impact of the recommendation on the outcome. However, most existing techniques rely on correlations between case characteristics and the probability of a given case outcome. In other words, they are correlation-based (see Chapter 3). Thus, future research should focus on developing causality-based methods that can estimate the effect of a proposed intervention using causal models (e.g., [131, 12]).

4.3.1. Implications

Based on the discussion above, we formulate the following implications:

- User group-specific information needs: The presentation PrPM outputs should be tailored to the information needs of the user group (operational workers, operational managers, and process analysts).

- Causal models for estimating effects of recommendations: Many existing PrPM are correlation-based methods, which limits the possibility to estimate how recommendations will affect the case outcome. More attention should be paid to researching causality-based methods to address the issue.
- Prioritization of ongoing cases: For operational workers, prioritizing ongoing cases based on process objectives or organizational criteria can be useful, since it is not realistic to address many cases simultaneously.

4.3.2. Limitations

First, the initial elicitation of information items relied on analyzing existing tools and reviewing outputs of existing methods, which may have resulted in omitting some information items needed by end users. To mitigate this, we conducted an evaluation with 13 experts from diverse domains and backgrounds.

Second, conducting the evaluation with different experts or using examples of other processes in the wireframe might have produced different outcomes. However, the focus was on evaluating the information items rather than domain-specific process attributes.

Third, the evaluation was conducted with experts rather than end users, which poses a limitation. This approach was chosen to gain a broader perspective on the interface for PrPM outputs, but it may not fully capture end-user-specific needs. In addition, future work includes conducting a second evaluation using an interactive prototype in a real-life scenario with end-users.

Fourth, the analysis of qualitative data is subject to risks of bias and subjectivity. To address this, the data and analyses were discussed within research team.

Finally, we abstain from making causal claims and describe the observations and discuss differences in the information items for different end-user groups.

4.4. Summary

In this chapter, we addressed **RQ**$_2$ *What user groups could benefit from working with the outputs of PrPM techniques? What are the users' information needs?*. To do so, we elicited an initial set of information items for a PrPM interface, evaluated them via a wireframe, and refined them. Namely, we analyzed existing tools and the PrPM domain to elicit the initial set of information items. Four main categories of information items were identified: Case Description, Recommendation Description, Recommendation Explanation, and Resource Assignment.

We then developed a wireframe and conducted an evaluation with experts from the field. Findings indicated that the selected information items are relevant. However, the results also revealed three distinct user groups with varying information needs: operational workers, operational managers, and process analysts. In this light, certain information items are more relevant for one user group as compared to another. Following the evaluation results, we formulated the information items for working with outputs of PrPM techniques for each user group.

5. DEVELOPMENT CYCLE II: PROTOTYPING & EVALUATION

The chapter corresponds to *Development Cycle II* of the Figure 1. The purpose of this cycle is to answer **RQ**$_3$ *How to translate these information needs into a user interface design for PrPM systems?* To do so, we apply *Design & Development* and *Evaluation & Refinement* phases of DSRM to create a prototype – a web-based PrPM tool called Kairos. We base the design of the prototype on the information items identified in the previous chapter. To evaluate it, we invite representatives of the end-user groups. The evaluation results help to refine the information items and formulate a set of suggestions for designing PrPM interfaces.

Section 5.1 introduces the method of prototype design and development, evaluation, and refinement, and Section 5.2 – the results. In Section 5.3, we discuss the findings and introduce suggestions for designing PrPM interfaces. Finally, Section 5.4 provides a summary of the chapter.

5.1. Method

In this section, we describe the method to create the prototype. Section 6.2.1 describes requirements elicitation for the prototype based on the findings from the previous chapter, Section 5.1.2 the design and development of the prototype, Section 5.1.3 its evaluation with end-user groups representatives, and Section 5.1.4 the refinement of the prototype.

5.1.1. Requirements

To derive the requirements for Kairos, we synthesized the findings from the analysis of user goals and information items from the previous chapter (Section 4.2.2). This allowed us to abstract individual information items into broader functional and non-functional requirements that reflect what the system should do and how it should support different user groups. The resulting requirements capture both the concrete information items to be presented in the interface (functional) and the overall system expectations (non-functional). The requirements were then categorized by type and linked to the user groups.

For example, one information item for all user groups was "List of recommendations and characteristics", which was turned into requirement FR2 "The system should display the details of each recommendation". The information item "Process model" was suited for process analysts only, and therefore requirement "The system should provide a process visualization that shows current state and recommended actions" is also specified as being for analysts only. The full list is provided in Section 5.2.1.

5.1.2. Design & Development

For functional prototyping, the focus was placed on two user groups: process analysts and operational managers. While operational workers represent a distinct group, the identified information items for them overlap with those of operational managers, with additional items for managers conveying resource-related information. As such, we prioritized prototyping for operational managers and process analysts, as this allowed us to cover the full set of information items identified in the wireframe evaluation. However, we acknowledge that operational workers pursue different goals in PrPM systems and therefore developing and evaluating a tailored interface for them is left as future work.

Based on the information items and elicited requirements, we did functional prototyping and developed a web-based tool for PrPM outputs called Kairos. In doing so, we applied the design principles described in Table 2. The tool consists of the engine that produces the recommendations and a visualization layer. The full description of Kairos architecture and functionality is given in Section 5.2.

5.1.3. Evaluation

We evaluated Kairos with two user groups: process analysts and operational managers. In describing the evaluation approach, we first give an overview and then elaborate on the specific details for process analysts and for operational managers.

The focus of the evaluation was on usability of the developed prototype and usefulness of information items included in it. We conducted the evaluation through semi-structured interviews. Each interview was divided into two phases: prototype demonstration followed by participant interviews based on questions from the interview guide (with examples for each user group provided in the subsequent sections). We used an online tool[14] for transcribing the interview recordings. Then, we manually corrected the transcripts. For analysis, we used a combination of deductive and inductive coding. First, we coded the transcripts using an initial set of codes related to $\mathbf{RQ}_3$ and the information items for a specific role (process analyst or operational manager) (Figure 11). During the coding, new codes emerged which were discussed within the research team, resulting in the coding scheme being modified. These new codes, primarily reflecting participants' suggestions for tool improvements, were added. The following subsections outline the evaluation details for each user group.

Process Analysts

Participants. We selected 12 process analysts who conduct process analysis in their daily jobs. Among those, we invited seven interviewees that had participated in the wireframe evaluation study (see Section 4.2.1). This allowed us to assess whether we interpreted their suggestions from the wireframe evaluation study correctly. The other five participants were new. In Table 8, interviewees I-02 to I-13

[14] `https://otter.ai/`

have the same codes as in the wireframe evaluation study (see Table 6), and the new interviewees are marked with codes I-14 to I-18. The process analysts we interviewed had on average six years of relevant work experience.

Table 8. Kairos evaluation interviews participants (process analysts).

Code	Domain	Experience
I-02	Banking, Finance Services, Insurance	7 years
I-03	Consulting (Process Mining & Data Analytics)	2 years
I-04	Consulting (Process Mining)	8 years
I-05	Consulting (Process Mining & Data Science)	5 years
I-06	Consulting (Process Mining)	7 years
I-12	Consulting (Process Mining)	6 years
I-13	Consulting (Process Mining)	6 years
I-14	IT Industry	4 years
I-15	Banking, Finance Services, Insurance	6 years
I-16	Consulting (Process Mining & AI)	8 years
I-17	Consulting (Process Mining)	3 years
I-18	Consulting (Process Mining)	10 years

Data Collection. At the beginning of the interview, we demonstrated Kairos and its functionality. Then, we shared a link to the web-version of the tool with the participant so they could use it themselves. An event log (loan application in a fictional bank) was pre-uploaded. We asked the participant to review the recommendations for two cases of their choice and share their thoughts while interacting with the tool.

In the second phase, we followed the interview guide. Following the aim of the evaluation, we asked about the usefulness of information items included in Kairos, e.g., *"Which information did you find most useful and why?"* and *"Which information did you find least useful and why?"*. The next set of question related to the participant's perception of the interface, e.g., *"Which aspects of the interface did you have trouble with?"* and *"What worked in the way you expected it to and what did not?"*. Another question related to understanding how Kairos could be used by process analysts in their daily jobs: (*"Would this tool have been helpful in the project? What tasks could you imagine using it for?"*). Last, we asked the participants whether there was information missing or how existing information could be improved, (*"What information is missing in the interface?"* and *"What could be improved about the tool?"*). The full interview guide is available in Appendix C. The interviews lasted between 16 and 34 minutes.

Coding. As mentioned above, we combined deductive and inductive coding. Namely, we first started with a set of codes that related to **RQ**$_3$ and the information items for process analysts. For example, a code "Purpose" related to describing the tasks that process analysts could use Kairos for. Then, codes such as "Process model" and "Recommendation characteristics" related to the specific information items (see Figure 11). The first set of codes also included codes to describe the usability of the tool, specifically "UI" for comments about the design of user interface, and "Interaction" for comments about the interaction with the tool. Last, we included codes to mark suggestions from the interviewees for improvements in the tool, such as "Suggestions for UI", "Suggestions for features", and "Other suggestions". During the coding procedure, new codes emerged. For example, the modified coding scheme included a code for each of the three types of recommendations (guiding, correlation- or causality-based, cf. Section 4.1.2). Another code that was added related to the participants' perception of displaying the three recommendation types in the interface separately ("Type needed" or "Type not needed") and the understandability of the types ("Type clear" or "Type not clear"). We also split the previous code "Purpose" into two codes: "Tasks" that described the tasks that process analysts could use Kairos for (refers to process analysts' responsibilities, e.g., to analyze a root cause of the issue in a process), and "Use cases" which specifically described use cases for Kairos in ongoing or completed projects of the participants (refers to specific examples of projects). There were also new codes related to the information the participants suggested to add to the tool. For instance, the interviewed process analysts suggested adding "Aggregated information" about the process and recommendations before going into detailed analysis. We then conducted another round of coding utilizing the modified coding scheme.[15]

Operational Managers

Participants. In the evaluation with process analysts, we used the loan application process event log. For consistency, we used this log for the evaluation with operational managers. Another reason is that we had access to potential participants (banking sector employees). We selected eight operational managers who worked in a banking sector as managers of a team of loan officers. Table 9 gives an overview of the participants' experience in banking, experience in a managerial position, and size of the team. The operational managers included in this study did not participate in the wireframe evaluation (Section 4.2.1) as the scope of that study focused on process analysts. Therefore, while some of the process analysts participated in the previous study, the operational managers did not.

Data Collection. Due to privacy and security reasons of the participants' organizations, they were not able to use Kairos on their own. Therefore, the demonstration of the tool was done by the interviewer. Thus, in the first part of the interview, we showed the participant the functionality pages for the manager utilizing the uploaded loan application event log.

Similarly to the evaluation for process analysts, the interview guide for operational managers drew from the $\mathbf{RQ}_3$ and the information items for them (Figure 11). In this regard, the first question was *"Think about your last project.*

[15] The full coding scheme can be found at: `https://figshare.com/s/b5180a06a2f4239e6bd3`

Table 9. Kairos evaluation interviews participants (operational managers).

Code	Experience in Banking	Managerial Experience	Team Size
M-01	25 years	8 years	17 people
M-02	17 years	10 years	6 people
M-03	6 years	1.5 years	13 people
M-04	19 years	12 years	12 people
M-05	20 years	15 years	33 people
M-06	20 years	3 years	50 people
M-07	13 years	6 years	14 people
M-08	22 years	8 years	16 people

Describe it a bit: what was it about? Would this tool have been helpful in the project? What tasks could you imagine using it for?" It aimed at understanding the general usefulness of Kairos for their daily jobs. Then, we inquired about the specific functionality in Kairos and the manager's approach to it: *"How would you sort the table of ongoing loan applications (cases) to prioritize them?"* and *"How would you assign the recommendations to your team members?"*. At the end, to identify possible additional information items, we asked *"What information is missing in the interface?"* The full interview guide is available in Appendix C. The interviews lasted between 12 and 23 minutes.

Coding. As in the process analysts' coding approach, we started with a set of pre-defined codes. The only difference in the starting set of codes was the specific information items (e.g., the codes for the managers did not include "Process model" but included "Resources"). New codes also emerged by coding. For example, we split the initial code "Purpose" into three. Thus, "Use cases" described the general use cases that the participants suggested using Kairos for, e.g., upselling. "Cases prioritization" and "Resource assignment" referred to the comments about these specific responsibilities that operational managers had.[16]

5.1.4. Refinement

We summarized the findings from the evaluation, sorting them according to the views in the tool that they related to. Findings were prioritized based on their importance (determined by the frequency of participant references during interviews) and the feasibility of implementation (evaluated in terms of required time and effort). Additionally, the information items for end-users were refined based on these findings. The results are detailed in Section 5.2.4.

[16] The full coding scheme can be found at: `https://figshare.com/s/b5180a06a2f4239e6bd3`

5.2. Results

In this section, we present the requirements for the prototype in Section 5.2.1, its design and development in Section 5.2.2, describing its architecture and functionality for the two end-user groups. In Section 5.2.3, we describe the evaluation results, also structuring them according to the user group. Section 5.2.4 summarizes the refinements in the prototype based on the findings and outlines the refined information items.

5.2.1. Requirements

Table 10 lists the requirements for the PrPM prototype, elicited based on the information items (Figure 11) and user goals (Figure 12) from the previous chapter. We divided them into functional and non-functional requirements. FR1 refers to the main function of the PrPM prototype, which is providing real-time recommendations in ongoing cases. This is to be based on the PrPM techniques that provide different recommendation types, including resources, which are relevant to operational managers and represented as part of information items group (*iv*) Resource Assignment. On the other hand, FR2 corresponds to group (*ii*) Recommendation Description. Next, FR3 refers to group (*iii*) Recommendation Explanation, specifically similar past cases and history of recommendations. Meanwhile, FR4 is on (*i*) Case Description, specifically KPI status and case-specific attributes. FR5 is on the same group, but it refers to the process model which was only found to be needed for process analysts. FR6 is on the model description from group (*iii*).

Table 10. Functional and non-functional requirements for PrPM prototype, based on information items and user goals (Section 4.2.2).

Req. ID	Requirement	User Group(s)
	Functional	
FR1	The system should provide real-time recommendations for ongoing cases.	All
FR2	The system should display the details of each recommendation.	All
FR3	The system should support viewing the history of past recommendations and their outcomes.	Operational Managers, Process Analysts
FR4	The system should allow users to view relevant case details.	All
FR5	The system should provide a process visualization that shows current state and recommended actions.	Process Analysts
FR6	The system should provide model description where applicable.	Process Analysts
	Non-Functional	
NR1	The system should tailor information representation to specific user groups.	All
NR2	The system should support extensibility to add or modify recommendation types.	All

Regarding the non-functional requirements, NR1 ensures that the way the information is presented is tailored to the users' needs. NR2 emphasizes that there exist many PrPM techniques that output different recommendation types. Since and new techniques that capture new recommendation dimensions appear constantly, we need to ensure that the prototype can be extended to support them.

5.2.2. Design & Development

As a result of prototyping, we developed a web-based interface for PrPM outputs called Kairos. Kairos is a web-based tool for prescriptive process monitoring. The tool can be accessed at: `https://kairos.cloud.ut.ee/`[17] In the following, we describe the architecture of Kairos, and then its functionality for both user groups.

Architecture

Kairos consists of two layers: visualization layer (graphical user interface) and PrCore (engine that produces the recommendations). The overview of the tool architecture is given in Figure 13.

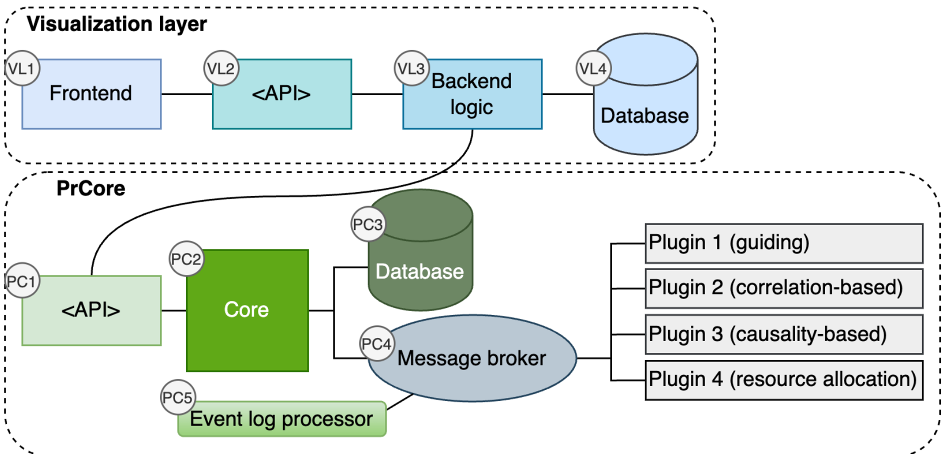

Figure 13. Overview of Kairos architecture: visualization layer (VL) and PrCore (PC). The numbers correspond to the components described in the text.

PrCore. PrCore comprises five components: API (PC1), core (PC2), event log processor (PC5), database (PC3), and message broker (PC4) (Figure 13). The core (PC2) coordinates all the components.

The four plugins in PrCore return recommendations based on different prescriptive techniques (as described in Chapter 3). The plugins extend the core application's capabilities.[18]

[17] See the video demonstration of the tool here: `https://youtu.be/51zgZw40ZzA`. Event logs to download and try, with descriptions of parameters, can be found here: `https://github.com/AutomatedProcessImprovement/kairos-frontend`

[18] Full documentation of PrCore can be accessed here: `https://github.com/AutomatedProcessImprovement/kairos-engine`

The plugins are as follows:

- Plugin #1 provides guiding recommendations. In our implementation, it returns the predicted next best activity in the case based on a KNN-algorithm.
- Plugin #2 implements a technique that returns an alarm when a user-specified probability threshold is used (based on [142]). Specifically, the technique predicts an outcome of the ongoing case, and triggers an alarm when the probability of negative outcome reaches the threshold. An alarm informs the user than an action is required but does not specify the action.
- Plugin #3 provides a causality-based recommendation. Specifically, the technique calculates the treatment effect of a recommendation and prescribes it when the effect is positive (above zero) (based on [15]).
- Plugin #4 provides resource allocation (based on [131]). The outputs of this plugin are relevant for operational mangers.

With the given architecture, other plugins containing additional techniques can be added to Kairos. Thus, this addresses requirement NR2. For example, a recent technique proposes to not only include the potential effect of a recommendation, but also incorporate its urgency [134]. Thus, this technique could be added as a new plugin to Kairos.

Visualization layer. The backend logic (VL3) in the visualization layer coordinates the connection between the database (VL4), PrCore and the frontend (VL1) (Figure 13). The visualization layer receives data from PrCore using API endpoints (VL3-PC1). The data is processed by the backend logic (VL3) and recorded in the database (VL4), and sent to the frontend (VL1) when requested.[19]

Functionality

The functionality of Kairos is based on the information items for the different user groups (Figure 11) and the requirements (Table 10). Figure 14 schematically depicts the functionality of Kairos in regard to its views. The views above the dotted line represent the setup functionality, and below – the functionality for the different user groups. Each view is described in detail below.

The input to Kairos is an event log. We specify that the user can upload the event log in .csv, .xes, or .zip format, and the max file size is 100 MB (Figure 15, a). In a real-world scenario, cases would be received directly from a company's information system. However, for the purpose of this study, we simulated cases streaming within the tool. This supports FR1 about representing the recommendations for ongoing cases in real time.

To start, the user uploads an event log. The user can choose to either upload one or two event logs. In the first option, the event log is split into train (80%) and test sets (20%). In the second option, the user uploads one event log that

[19]The source code of the visualization layer can be accessed here: frontend `https://github.com/AutomatedProcessImprovement/kairos-frontend`, and backend `https://github.com/AutomatedProcessImprovement/kairos-backend`.

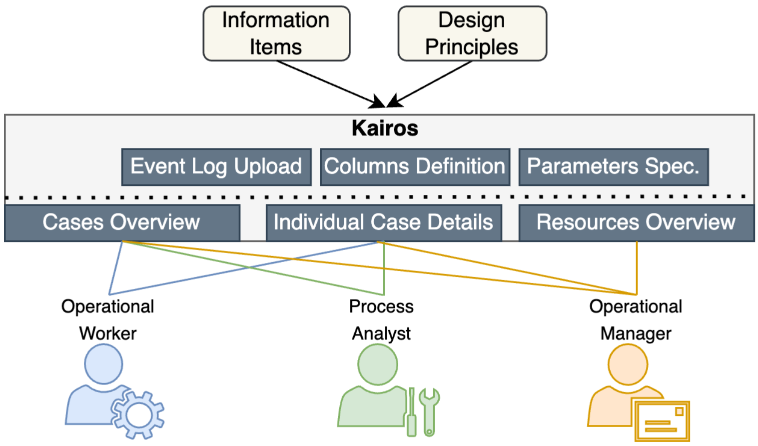

Figure 14. An overview of Kairos functionality for the three user groups.

contains only completed cases, which is used for training, and one more to be used for testing (that contains not completed cases). In the first option, the data is streamed, and in the second, it is delivered as a single response.

Then, the user defines data columns (Figure 15, b). In the example of a loan application event log, defined columns are timestamp, activity, case ID, as well as case attributes, such as cost (numerical), loan goal (text), credit score (numerical). After submitting the defined columns, the user then specifies the parameters (Figure 15, c). These parameters are: 1) The activity that marks the completion of a case. 2) A condition marking the positive outcome of a case. The outcome types are extracted from the event log by PrCore. The outcome types can be one of the columns in the event log (such as cost, specific activity, resource) or the duration of a case. For example, if a positive outcome is specified as equal to "duration less than or equal to 10 days", all cases lasting 10 or fewer days are marked to have a positive outcome, while the rest are marked as negative. 3) An intervention for plugin #3 [15], where the list of options is also provided by PrCore. For example, if the intervention is specified as "Activity equals [Contact_Customer]", then an algorithm estimates the causal effect of performing the activity [Contact_Customer] at a given point in time. 4) The threshold for the probability of a negative outcome, used to trigger an alarm for plugin #2 [142]. 5) The resources for plugin #4 [131]: names (have to be typed in), and the duration of time they will be busy carrying out a recommendation.

If at any point during the event log upload and parameters specification errors appear in PrCore engine, the user is notified about them with a clear message (e.g., "Incorrect file format error."). Similarly, if the user tries to proceed without specifying data in a mandatory field, they receive a notification to correct it.

When an event log is uploaded and the parameters are specified, the event log appears on the dashboard. There, the event log's status is shown (e.g., "training").

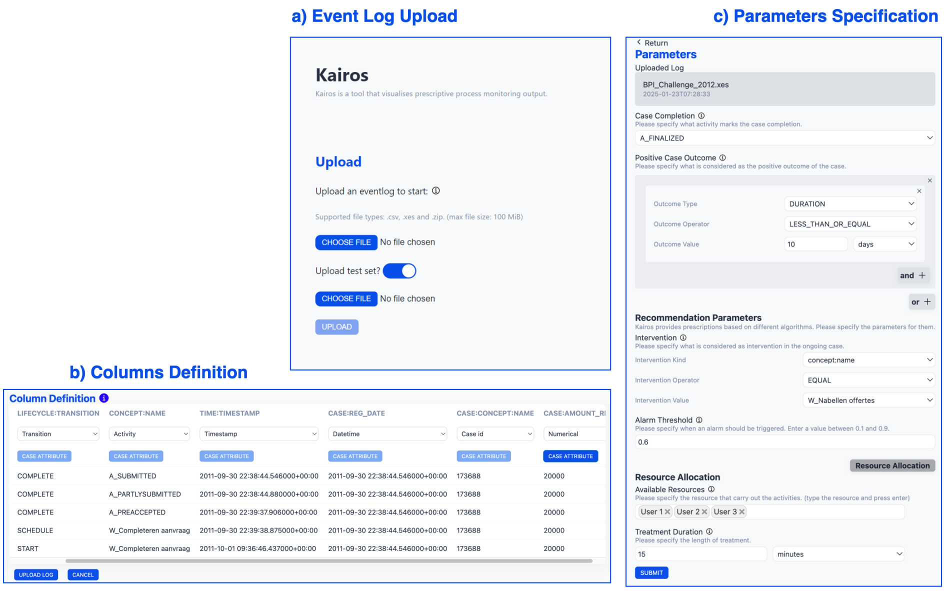

Figure 15. Kairos setup functionality: event log upload, columns definition, and recommendation parameters specification.

When the status is set to “trained”, the user can start the simulation by clicking on the respective button. After this, the log status is changed to “streaming”, and the user can view the respective pages based on their role (e.g., process analyst, operational worker, or operational manager).

The tool consists of three main tabs: Cases, Recommendations, and Resources. The Resources tab is targeted at the operational manager, while the other two contain information items relevant for all user groups.

Functionality for Process Analyst

In view #1, a process analyst is presented with **Cases Overview** (Figure 16, top). In line with the information items, the analyst can familiarize themselves with ongoing and completed cases. In each list, case-specific attributes and the KPI are displayed. For example, for the loan application process, the user can see the requested amount in the case. The selected KPI in this process is case duration. Therefore, the duration for each case is specified. These information items were seen as required by the interviewees in the wireframe evaluation. The table also shows whether recommendations are available in the case, and whether an operational worker has already intervened in the case, i.e., accepted a previous recommendation. In the completed cases overview, a pie chart is displayed with information on the successful outcome of the past cases where recommendations were addressed. When selecting a specific case, the analyst is directed to the case page where more details about the case are shown (view #2).

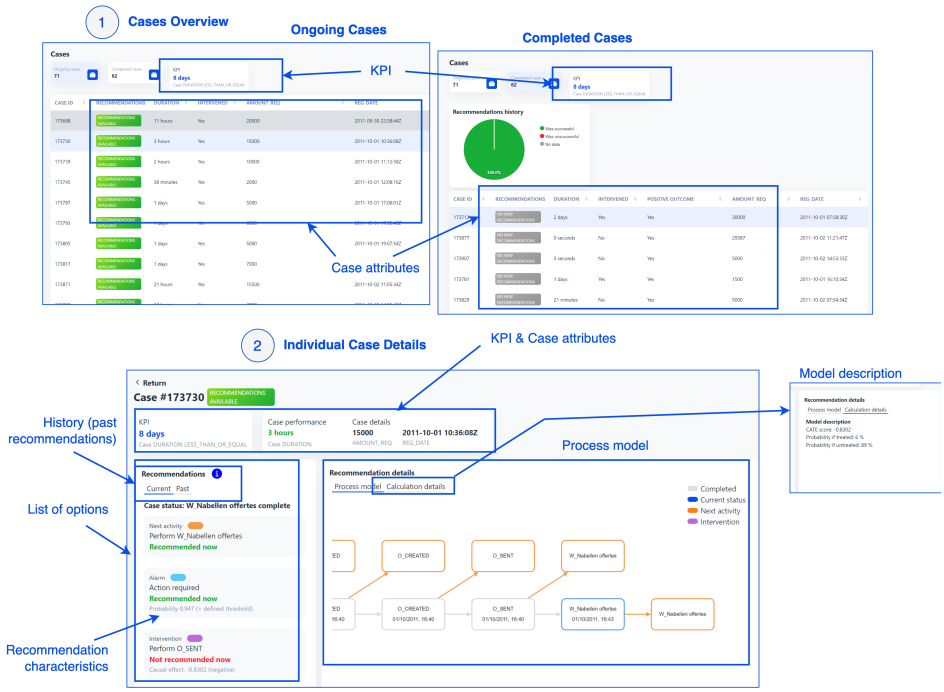

Figure 16. Kairos functionality for process analysts: view #1 Cases Overview and view #2 Individual Case Details. [Enlarged version of this figure in Appendix B]

View #2 shows **Individual Case Details** (Figure 16, bottom). The process analyst is presented with case-specific attributes and KPI status in the case details at the top of the page (requirement FR4).

The list of options displays currently prescribed recommendations in the case. For each recommendation, **Recommendation Details** are provided. The details include the recommendation characteristics, i.e., the type according to the prescriptive technique that generated it. The guiding technique's output is marked as "next activity" (orange), correlation-based as "alarm" (blue), and causality-based as "intervention" (purple). The user can read about the types when clicking on the info button above the recommendations. This addresses requirement FR2.

Apart from the current recommendations, the user can also see the ones that were prescribed in the past. For the recommendations that were prescribed in the past, there is also an indication of whether the recommendation was accepted or discarded by an operational worker (simulated). This addresses requirement FR3.

On the right, the process model is displayed (requirement FR5). The colors for each recommendation type are the same in the process model and the list of options. Last, the basic description of the ML models (accuracy, precision, recall) can be seen when clicking on calculation details (requirement FR6).

Figure 17. Kairos functionality for operational managers: view #1 Cases Overview, view #2 Individual Case Details, and view #3 Resources Overview. [Enlarged version of this figure in Appendix B]

Functionality for Operational Manager

For operational managers, the **Cases Overview** page (view #1, Fig. 17) contains the same information items as the page for process analysts described above. View #2, **Individual Case Details** consists of four elements: KPI & Case attributes pane (same as for process analysts), the case status, recommendations, and resources. The case status pane displays the activity that was completed last and all completed activities in a sequence. The tab with recommendations does not display the calculation details directly. Instead, they are available when the user clicks on the info button on the recommendation. With this, we ensure requirement NR1. Last, among the information items for managers are the resources. Thus, the element with Resources displays the resource that is recommended for this case (if any), and a list of all resources available. If the user wants to assign a recommendation to a resource, they are prompted to do so in a window that opens.

On view #3, **Resources Overview**, information on available and busy resources is displayed, as well as the current workload (busy vs total resources), and the current or the most recent case the resource has worked on.

Design

We followed the design principles described in Table 2 to design the interface. The general application of the design principles in Kairos is depicted in Table 11.

In the following, we describe Kairos based on the Tiramisù framework which consists of four components: backdrop (or base), dimension layers, visual mappings, and visualization configuration [7] (see Chapter 2.4.1 for detail). We use the view #2 Individual Case Details to exemplify the application of the framework (Figure 16, bottom). The framework application is summarized in Table 12.

Backdrop. The backdrop provides contextual elements not directly related to data but essential for interpretation [7]. Such elements include the case attributes, e.g., case performance and details. In addition, the sidebar navigation assists with navigating between the dashboard, cases, and views by role (e.g., process analyst). The tabs: current vs. past recommendations help the users frame the current context vs. historical decisions. These elements anchor the interface in the business process setting and user workflow context. The layout of the interface follows the matrix approach [11]. A matrix is well suited to content with multiple related categories with relations between them [11]. In this example, there are separate sections for case details, recommendations, and the process model. The recommendations are also displayed on the process model, and are related by color.

Dimension layers. These are the primary data abstractions, such as process activities, events, timestamps, etc. [7]. The process activities are represented in the process model. In creating the process model visualization, we followed the visual processing guidelines for diagrams [88]. For example, we harmonize the elements in the model by representing the past and current recommendations in the same

Table 11. Application of design principles in Kairos interface (based on Table 2).

Design principle	Example of application in Kairos	View (Element)
Offer informative feedback	Event log status is shown when it is training ("Training"), trained ("Trained"), and when the user started the simulation ("Simulating")	Dashboard (event log details)
Strive for consistency	The same elements have the same look in different views for user groups	Cases overview (ongoing and completed cases tables)
Prevent errors	Accepted file formats (.csv, .xes and .zip) and file size (max: 100 MB) are displayed	Setup (event log upload)
Minimize user's memory load	Outcomes are extracted from the event log and displayed to the user when selecting the positive outcome	Setup (positive outcome selection)
Simple and natural dialogue	Only title, type, and parameters of recommendation are displayed, with additional text placed in an info button	Individual case details (recommendations)
Good error messages	Displaying a notification to specify the data in the mandatory field if the user tries to proceed without doing that	Setup (columns definition)
Provide shortcuts (customizations)	Possibility to hide or show the menu	All
Provide clearly marked exits	A "return" button on top of the page	Individual case details
Speak the user's language	For probability, a textual indication is displayed for what the number means (e.g., 0.27 is "low" probability)	Individual case details (recommendations)

way. We ensure that the diagram is simple by not adding any other details, rather than the past current, and recommended activities. The colors of the recommendations correspond to the ones in the recommendation panel which emphasizes the key elements and ensures that the different elements can be distinguished. In summary, this layer encodes process execution, and recommendations.

Visual mappings. This refers to how data values are mapped to visual features like position, color, and shape [7]. Color is used for categories labeling, but also indicate status changes. In the given example for process analysts, we use colors as nominal codes for different recommendation types. According to the visual guidelines for color coding, applying color in such cases does not need to be orderable, but recognizable and matched to a legend [147]. Therefore, the colors used for the different recommendation types (orange for "next activity", blue for "alarm", and purple for "intervention") are also displayed on the process model and its legend. In addition, we apply colors in view #1 Cases Overview for process analysts (Figure 16, top). Namely, we apply the green color for the status changes, marking that a recommendation is available. If none is available, the color is changed to gray. In terms of elements, the recommendation panel groups current and past recommendations. Each recommendation is represented in text. With the usage of color, it is marked if an action is recommended (green) or not (red). The color is also used to distinguish the recommendation types, as described above.

Visualization configuration. The final component refers to user control and customization features [7]. For example, toggling for past recommendations enables inclusion of past decision points. Similarly, tabs for process model vs. calculation details enable deeper inspection of recommendations.

Table 12. Description of Kairos based on Tiramisù framework [7] using view #2 for process analysts (Individual Case Details).

Component	Example of application in Kairos
Backdrop	Case details, navigation sidebar
Dimension layers	Process activities, timestamps, recommendations
Visual mappings	Color-coded activity boxes and arrows, current activity highlights, status indicators
Visualization configuration	Toggle for past recommendations, tabs for detail level

5.2.3. Evaluation

We present the evaluation findings according to the user group. First, we describe the findings for the process analysts and then, for operational managers. For each user group, the results are structured based on the views presented in the previous section. A summary of the findings, along with refined information items for the two user groups is given at the end of the section.

Process Analysts

Cases Overview. Process analysts we interviewed shared a view that in the context of prescriptive process monitoring, *"completed cases will be the main focus"* (I-13) for a process analyst. The main reason is that this helps the analysts to identify patterns in recommendations and find improvement opportunities. One interviewee commented, *"As a process analyst, usually, I'm looking back at the process. So I'm not really used to looking at what should happen right now."* (I-15). Therefore, in Kairos, the views for process analysts should be shifted towards reflecting what has already happened in the process. One important information to display here is the accuracy of the PrPM techniques: *"As an analyst, I would be probably interested to see how this prescriptive models actually are working"* (I-17). Our interviewees also highlighted the importance of displaying aggregated information about cases and their recommendations, rather than only focusing on separate cases. In this regard, what could be displayed is the numbers of prescribed recommendations, as well as their success rate. As one participant put it, *"some insight into [...] which kinds of acted recommendations [...] result in successful versus unsuccessful outcomes?"* (I-18).

For the ongoing cases, our interviewees had a similar perception that they would start with a high-level overview before reviewing separate cases: *"I would prefer [to] start with the top and go with drill downs to cases"* (I-16). One interviewee explained that looking into individual cases might be useful for specific use cases, *"[...] unless I look for audit or compliance [...], I'm not very interested in individual cases. I am more interested in optimizable sizes and chunks."* (I-04). In this context, aggregated data on the currently prescribed recommendations should be displayed, such as the number of similar cases receiving the same recommendation. For the analysis of ongoing cases, an automated prioritization of cases requiring immediate attention would be useful for the process analyst, for instance, based on target performance measures. The prioritization would help to identify cases requiring urgent attention faster, since the need to *"go through case by case to see whether there is something I can actually improve on"* (I-17) would be eliminated. Another suggestion from the participants on how to simplify working through the list of cases was to introduce filtering options. As one interviewee put it, *"Maybe I want to filter on which cases have predicted probability of violating KPI as high, or which one has a recommendation?"* (I-03).

In conclusion, the findings indicated that the process analyst's primary focus is on aggregated information regarding completed cases and recommendations. Access to this information would enable analysts to assess the effectiveness of past recommendations and identify improvement opportunities. Although ongoing cases are also relevant, the analysis of these cases should initially emphasize aggregated information before delving into detailed views of individual cases.

Individual Case Details. Process analysts found it useful to visually see the recommendations overlaid with the process model. Majority of the participants commented that the process model was *"easy for me to interact with"* (I-14).

However, they also shared a number of suggestions on how to make it more usable. For instance, recommendation types now have different colors in Kairos. At the same time, current and past recommendations of the same type have the same color (e.g., next activity is orange both in the current and the past view). In this regard, several participants shared that it would be beneficial to distinguish the current and past recommendations, e.g., also by color: *"I would probably segregate historical colors from current colors because it can be very misleading. I don't know where I am in the process."* (I-05). Similarly, some participants suggested introducing a toggle overlay for recommendations on the process model. In this way, the user could study the process model in its current state, and switch on an overlay with past recommendations when needed.

Some interviewees found it challenging to understand what the label "KPI" in an individual case related to. As one interviewee put it, *"I don't understand what we want to optimize in this case. What's the end point of the recommendation for this credit application? Is it to optimize the fact that the credit application would be fast? [...] KPI is just a measurement."* (I-13). Therefore, some process analysts suggested the need for a clear indication of the optimization target which could be addressed by simply modifying the interface labels.

Similar observations were made regarding the case-specific attributes displayed on the interface. Thus, some participants expressed concerns about presenting attributes from more complex event logs due to space constraints. Another concern involved the clarity of attribute presentation, such as specifying the currency for the loan amount attribute. These are existing limitations that could be addressed in the next design iteration.

In conclusion, the analysts we interviewed found it useful to see current and past recommendations in the process model. At the same time, the participants had suggestions on how to improve the process model, mainly focusing on the use of colors for the different recommendation types. Several suggestions were also made for naming of elements in the tool, such as for the target performance measure and the case-specific attributes.

Recommendation Details. In its current version, the interface specifically shows the three different types of recommendation with respective labels and colors. Our findings indicate that the participants had a mixed understanding of the recommendation types. Two interviewees reported that it was clear for them what the differences were. Other six interviewees said it was not clear at first, of which three understood it after reading the info box and further interacting with the tool. Additionally, two interviewees shared they were not clear why the distinction is needed. As one participant expressed, *"I guess [...] I would expect something that's closer to the action that's required from me or the information I want to pull out of it."* (I-17). This comment describes that the focus should be on the action from the end-user rather than the recommendation type stemming from a PrPM technique. The difference in opinions could be explained by the interviewees' background. For instance, I-05 reported that they prefer to see the different types because *"I have data rather than business background, and I don't like having one option."* At the same time, interviewee I-12, who has a business background, said *"as a user, I don't care. You know, what I just want to know [is] what should I do next."*

From the recommendation types, some process analysts struggled specifically with understanding the type produced by the correlation-based plugin. Unlike the other two types, this one is not reflected in the process model since it is an alarm and not a specific activity to show on the model. On this, one interviewee commented, *"So the next activity that I can perform is activity 'Cancel', then I have the blue thing there but I don't have it in legend. So I don't know how to interpret it."* (I-14). This recommendation prompts the user to pay a closer attention to the process instead of providing a clear next step. Therefore, some process analysts thought it would be beneficial to present this type differently from the other two, such as show it as an alert in the case description rather than display it in the list of recommendations.

The current version of Kairos provides limited information on how the PrPM techniques work. Process analysts shared an opinion that more explanations are required to better understand why the recommendations are prescribed. For example, explanations could be provided with regard to the case attributes, and they should be presented in a way that is understood by users with limited experience of technical terms. As one interviewee put it, *"So [is it] risky or [...] it would be a bad debt because this client didn't pay back? [...] Some sort of business reason that explains that we need to cancel it [is needed]."* (I-13).

Furthermore, in the evaluation, the participants saw the basic description of machine learning models (accuracy, precision, and recall metrics) that produce the recommendations. In addition, probability is displayed for plugin #2, and causal effect for plugin #3 (Figure 13). Our findings suggest that there should also be more explanation specifically towards the outputs displayed in the interface. For example, as one interviewee said, *"'Casual effect'. Nobody will know what it means. 'The causal effect is negative.' Is it good? Is it bad? Nobody will know."* (I-05). This implies that the explanations should also provide insight towards the applicability of specific technical terms in practice.

In summary, the interviewed process analysts expressed contrasting views regarding the display of different recommendation types, with some considering it necessary and others finding it unnecessary. These differences may be related to the different backgrounds of the interviewees, specifically whether they are more business- or technically-oriented. A shared opinion among participants was the need for clearer explanations of the nature of recommendations, regardless of type, as well the details of the outputs (e.g., the specific terms). For example, this could include describing the model descriptions in more detail, as well as detailing how the attributes of the current case contribute to the recommendation. Such explanations should also be understandable by users without a technical background.

Operational Managers

Cases Overview. We observed mixed opinions about Kairos' usefulness when put in a specific use case. For example, a participant shared that such an interface would be useful to gain an overview of the state of the process: *"I think that is quite good view [...] how the process right now is doing basically, is it [...] everything is good or not."* (M-01). At the same time, several participants highlighted that using an interface like Kairos might be challenging in regulated environments, such as banking, due to policies that pre-define steps in certain processes. In this regard, M-04 commented *"[...] in banking there are strict processes applied. So pretty much all the applications need to go through a strict process."* This means that it could be challenging to follow a specific action prescribed by an interface like Kairos at a specific time, if the strict regulation requires to do something else. However, the interviewees voiced use cases where following the recommendations would be theoretically possible, as well as useful, e.g., sales. For example, M-03 commented, *"this could help with upsell process because this is something that the agents [...] forget about it."*, and M-04 said *"for example, making a phone call or something regarding the sales process of the loan [could be useful]."*

Another question that we asked in the interview was related to the cases prioritization from the manager's perspective. Here, the operational managers we interviewed highlighted several approaches to prioritization within their team, e.g., case duration and customer status. For example, M-03 commented, *"[It is] mostly the time how much the application is spending in our dashboards. So we start with the oldest."* As for the customer status approach, M-05 shared that *"the most prioritized customers are our private banking customers or our own customers [...]"* One interviewee also shared that they use a half-automated approach to prioritize and assign incoming applications: *"We have robots for the application sharing. So previously it was like purely manual, but now it's like it's not full fully automated, but 70% of the applications are distributed by the robot and like 30% manual."* (M-06). In Kairos currently, only manual prioritization is available. Thus, an improvement for future is to add automatic prioritization.

In addition, a question of integration was used, i.e., how an interface like Kairos would be used in existing daily operations. Namely, M-06 gave an example, *"let's say on top of that [our system], [we] work with the dashboard. So like it would be additional work for all the users?"* This could be resolved by ensuring that such interfaces could be integrated into companies' existing systems or seamlessly connected with them.

In conclusion, participants expressed concerns about the use of Kairos for implementing recommended steps in regulated processes but recognized its potential value in other use cases. Additionally, integrating automatic prioritization techniques into Kairos was suggested. Consideration should also be given to how Kairos could be utilized on a daily basis within organizations, taking into account their existing technical infrastructures.

Individual Case Details. A common question among the participants was whether the recommendations in individual cases were based on case-specific attributes. For instance, M-02 asked *"if a client has a salary which is too small, what will be the recommendations? What to do with this application?"*. Similarly, M-01 shared that *"I as [a] manager want to see all data for some project, for some customer, is it for example application data and income data and so on."* This could be addressed by providing explanations on how the techniques generate the recommendations (e.g., feature importance).

The operational managers we interviewed found resource recommendations useful. They particularly described several use cases where such recommendations would be useful. For instance, re-assigning a recommendation from one resource to another in case of resource unavailability was highlighted: *"And where I see the use of this [is] in case of vacations or sick leave. [...] A manager could connect and see what is now next steps and assign to another employee."* (M-07).

In summary, interviewees indicated that recommendations for operational managers are most effective when linked to specific case attributes. Also, recommendations for resource allocation for particular case steps were considered valuable.

Resources Overview. The previous paragraph referred to case-level resource recommendations. As to view where an overview of resources workload is provided (Figure 17, view #3), the participants suggested that it could be extended to incorporate resource workload and case statuses for each resource. This would help the operational manager to understand where adjustments might need to be made. As M-01 put it, *"For me it's more important that [for] all my 17 people, [...] that I have how many different statuses [of] project[s] we have. And also is there some person who is in the trouble or we have lots of troubles."* Another interviewee brought up understanding resource performance in terms of case processing time: *"if we could measure the time it takes for only the client manager [loan officer] to run through process, then it would definitely have some value."* (M-04). Similarly, M-02 shared that resource performance also includes the number of applications that lead to concluded contracts: *"if it's [number of contracts] very low, like 10%, then of course we have to look inside what this employee is doing differently, that he or she doesn't get these contracts."*

In summary, the resource overview could be enhanced by incorporating additional features for operational managers, such as a summary of case statuses for each resource and an overview of resource performance.

Findings Summary

The summary is provided in Figure 18. The table only concerns functionality improvements. In addition to them, there were a number of suggestions related to the design and interactions, e.g., differentiating the colors of current and past recommendations, and adding interactive zooming to the process model.[20]

#	Finding	User	Example Quote	View	Priority
F1	The focus of PAs was on completed cases. An aggregated view of completed cases should be added to assess the recommendations in the completed cases.	**PA**	*"As an analyst, [...] I would be really interested into seeing statistics and dashboards about how was it applied by operational people. How was it used, what recommendations were taken, which were just seen, but never really used, because those are probably then bad ones."* (I-11)	Cases Overview	High
F2	An aggregated view should also be added to the ongoing cases with a possibility to view individual case details.	**PA**	*"I would say [...] from my perspective to be able to analyze the process, I first need to understand it or have an overview of what's happening and then I can drill down to specific things"* (I-10)	Cases Overview	High
F3	Manual filtering options should be added to the cases list to simplify working through the cases.	**PA**	*"Maybe I want to filter on which cases have predicted probability of violating KPI as high, or which one has a recommendation?"* (I-03)	Cases Overview	Medium
F4	There should be more explanation about recommendations. This relates to understanding how the different recommendations are generated, as well as what the different recommendation types and other terms mean in practice. It should be presented in a way that would be understood by business users.	**PA**	*"As a process analyst, [...] [I] understand the meaning of accuracy recall precision, but I think this is too high level for if you're trying to explain it to the client. [...] Like I will have to probably come up with a bit more details to help them understand what do I mean by those? [...]"* (I-03)	Individual Case Details	Medium
		OM	*"if a client has a salary which is too small, what will be the recommendations? What to do with this application?"* (M-02)		
F5	Automatic prioritization techniques should be integrated to help the users focus on the most important cases.	**PA**	*"So what I would be interested in seeing is some sort of prioritization. So for me to instantly know, what is priority and what isn't a priority."* (I-04)	Cases Overview	Medium
		OM	*"if you get the application, we really want to answer it within one day. So if someone feels that it's not possible because I have too many work, then we can share these applications and we always have someone who can do it today. So this is our purpose."* (M-02)		
F6	A functionality to display specific resource's case statuses should be added.	**OM**	*"For me it's more important that [for] all my 17 people, [...] that I have how many different statuses [of] project [s] we have. And also is there some person who is in the trouble or we have lots of troubles."* (M-01)	Resources Overview	High
F7	Resource performance (e.g., case processing time, cases leading to a positive outcome) should be incorporated in the overview.	**OM**	*"if we could measure the time it takes for only the client manager [loan officer] to run through process, then it would definitely have some value."* (M-04)	Resources Overview	Medium
F8	It should be considered how PrPM interfaces can be integrated with companies' existing systems.	**OM**	*"let's say on top of that [our system], [we] work with the dashboard. So like it would be additional work for all the users?"* (M-06)	All	NA

Figure 18. Summary of the prototype evaluation: functionality improvements (PA – Process Analyst, OM – Operational Manager).

5.2.4. Refinement

Based on the evaluation findings, we refined the information items for both user groups (Table 13). We have also improved the grouping of the refined information items compared to those stemming from the wireframe evaluation (Figure 11). The updated grouping aims to align with users' interactions with the information items, as informed by the evaluation. Specifically, the Cases Overview group addresses the general understanding of ongoing and completed cases. This includes information items such as process KPIs, a list of cases, and aggregated data on cases and recommendations (e.g., acceptance vs rejection rates, success rates). The Individual Case group focuses on information related to specific cases, in-

[20] In the currently available web-version of Kairos, the functionality improvements for process analysts are already implemented. The functionality for operational managers will be implemented in the next releases.

cluding the process model (for process analysts), a list of recommendations, and explanations. Finally, the Resources group provides information for managers, covering an overview of resources and their performance. Together, these information items address end-user needs and serve as a foundation for designing interfaces for PrPM.

Table 13. Information items: refined after prototype evaluation (PA – Process Analyst, OM – Operational Manager).

Information Items Groups	Information Item	PA	OM
Cases Overview	* Clearly indicated KPI and metrics for the process.	x	x
	* List of cases with case-attributes. Possibility of manual prioritization and filtering. Automated prioritization of cases based on the KPI.	x	x
	* Aggregated data on cases and recommendations available for them.	x	x
Individual Case	* Process model with overlay of current and past recommendations.	x	
	* List of recommendations with level of detail based on user's needs.	x	x
	* Explanation for a recommendation generation (e.g., case attributes impact) and recommendation output (e.g., to detail the recommendation type and other attributes).	x	x
Resources	* Overview of resources with workload and statuses of cases they are assigned to.		x
	* Resource performance (e.g., case processing time).		x

5.3. Discussion

Kairos is based on information items for process analysts and operational managers from the wireframe evaluation (Figure 11). Upon evaluating the tool with process analysts, we confirmed that a process model is useful for them. This finding adds to previous works [74, 77] who reported that a process model is used when working across all use cases (e.g., discovery, prediction, drift detection, etc.). According to our findings, the interviewed process analysts found a process model with an indication of prescribed recommendations to be a necessary element in a PrPM interface. The evaluation also confirmed that information on resources is the most valuable output of PrPM techniques for operational managers. Therefore, if a PrPM technique is targeted at managers, it is important to incorporate resource aspects into it (e.g., [115, 131]). At the same time, while in the previous development cycle we also identified the end-user group of operational workers, we did not develop and evaluate an interface with them. Therefore, we did not confirm or refine the previously identified information items for this end-user group. Future work should conduct a targeted evaluation of an interface for operational workers.

Our findings indicate that incorporating automatic prioritization techniques would be beneficial for both user groups. Such techniques can assist users in identifying and focusing on cases requiring urgent attention. For instance, cases that remain open but are approaching a deadline could be automatically prioritized and presented to users as a priority. Future research could draw inspiration from the literature on recommender systems to incorporate such prioritization techniques into PrPM systems [23].

The interviewed process analysts found it valuable to understand the currently prescribed recommendations. However, as also indicated by the interviewees, the initial step in analyzing recommendations and their impact on the process involves examining completed cases. This can be expected given that analyzing the historical data about the process is one of the main process analysts' tasks [159]. However, PrPM deals with the current state of the process and improving the ongoing cases. Therefore, PrPM interfaces should enable process analysts to review recommendations and their configurations for completed cases. This functionality allows analysts to better understand the techniques behind the recommendations and to identify potential improvements in their configuration, ultimately enhancing the support provided to other user groups.

The interviewed process analysts found that the outputs of PrPM techniques in Kairos should be explained better. Additionally, operational managers highlighted the need to understand how recommendations connect to specific case attributes. Works on explainable prescriptive process analytics have started to appear [112] but their scope is yet to be extended. Inspiration could be drawn from the neighboring fields, such as predictive process monitoring [153, 41]. However, as our findings suggest, process analysts do not always understand the technical explanations, such as numbers and graphs describing the machine learning models used to produce the recommendations. This finding is in line with Rizzi et al. [122], who found that process analysts without ML background struggle with similar explanations in predictive process monitoring. Therefore, one avenue for future work is developing explanations that could be understood by the end users with less technical experience, e.g., explanations in connection to the business outcomes.

The evaluation with operational managers revealed that the application of PrPM techniques may be constrained in certain contexts, such as highly regulated environments. This is due to strict process rules that do not allow for deviations. A direction for future research in PrPM is to investigate both the benefits and limitations of PrPM techniques in such settings and explore potential strategies to address these challenges.

Kairos included recommendations produced by different types of PrPM techniques (guiding, correlation- and causality-based [66]). The process analysts we interviewed had no unified perception of the understandability of specific differences between the three types. In this regard, another potential future research area is an A/B study of two interfaces, where one clearly defines the type of the recommendation, and the other one only specifies the next action.

5.3.1. Implications

Based on the evaluation and the discussion above, we propose several suggestions for designing PrPM interfaces:

1. Personalized interfaces: Interfaces should be tailored to the specific information needs of each user group. For instance, in the case of process analysts, the inclusion of a process model for visualizing process states and recommendations, as demonstrated in Kairos, is essential.
2. Explanations: A PrPM interface should provide explanations for the outputs, such as to help understand the recommendation type better, as well as provide some technical details about the calculation of recommendations. The evaluation of Kairos revealed that end users sought more contextual explanations for the prescribed recommendations.
3. Interactive data exploration: PrPM interfaces should support the visual exploration of data. For process analysts, this could involve an interactive process model. Kairos evaluation indicated that process analysts rely on the process model to understand possible case trajectories. Visualizing recommendations within the process model enhances the comprehensibility of recommendations and their types, addressing challenges faced by some process analysts.
4. Decision-making support: A PrPM interface should assist end users in decision-making by providing role-specific aggregated information. For process analysts, this includes historical data on past recommendations, while for operational managers, resource-related information would be most relevant. Additionally, automatic case prioritization should be implemented to help users focus on the most urgent cases.

5.3.2. Limitations

First, the evaluation utilized only one event log as an example, which may limit the generalizability of the findings. Using additional event logs could potentially yield different results. Additionally, the study focused exclusively on the financial sector. However, the objective was to evaluate the information items rather than domain-specific process attributes.

Second, there is a threat of selection bias, as only a subset of participants was interviewed. To address this, participants were selected based on specific criteria, but involving other participants may have produced different outcomes.

Third, the interpretation of qualitative data poses a risk of bias or subjectivity. This risk was mitigated by collaboratively discussing the data and coding schemes within the research team. Furthermore, no causal claims are made; instead, the observations from the two evaluation iterations are described and discussed.

Fourth, the tool has functional limitations, as it only includes the initially prioritized features. Future evaluations will be conducted as the tool is further developed and improved.

5.4. Summary

In this chapter, we addressed **RQ**$_3$ *How to translate these information needs into a user interface design for PrPM systems?*. First, following the information items elicited in the previous chapter, we did functional prototyping and designed and developed a web-based tool for PrPM called Kairos. Then, we conducted an evaluation of Kairos with process analysts and operational managers. The tool was generally found to be useful and usable, but a number of improvement opportunities emerged. The improvements suggested including aggregated overviews of completed and ongoing cases for process analysts, adding automatic prioritization techniques and explanations for both user groups, and information on resource performance and resource's case statuses for the operational managers.

Based on the evaluation, we refined the information items for the interviewed end-user groups. As such, for both user groups it is important to include a clearly indicated KPI to be optimized, along with an aggregated data on cases and recommendations available for them. For individual cases, the level of detail for the recommendations should be tailored to the specific user group. Further, a process model with an overlay of recommendations is essential for a process analyst. Explanations for PrPM techniques' outputs should be presented to both user groups in such a way that it is understood by users with less technical insight. Additionally, an overview of resources with workload, performance, and case statuses is important for the operational managers.

Finally, we elicited a set of suggestions for designing prescriptive process monitoring interfaces, referring to personalized interfaces, the need for explanations, interactive data exploration, and decision-making support.

6. DEVELOPMENT CYCLE III: OUTPUT EXPLAINABILITY & EVALUATION

This chapter describes the *Development Cycle III* of Figure 1. The purpose of this cycle is to answer **RQ**$_4$ *How to enhance the understandability of PrPM outputs?* To do so, we apply *Design & Development* and *Evaluation & Refinement* phases of DSRM. First, we scope the problem, i.e., enhancing the understandability of PrPM outputs. Second, we elicit requirements by defining a set of contextualized questions from the eXplainable AI Question Bank [87]. Based on these contextualized questions, we design and develop our artifact: a prompting method that enables an LLM to elaborate on and explain PrPM recommendations. To evaluate the artifact, we implement an LLM-based chat on top of Kairos.

Section 6.1 introduces the necessary background for this Chapter, namely, explanations in AI systems, and LLMs & prompt engineering. Section 6.2 describes the method: artifact requirements elicitation, artifact design and development, and artifact evaluation with end-users. Following the same structure, Section 6.3 describes the results. We discuss the findings of the evaluation in Section 6.4, and conclude the Chapter in Section 6.5.

6.1. Background

In this section, we introduce the necessary concepts for understanding the rest of this chapter. Namely, Section 6.1.1 describes the different aspects of explanations (explanation focus, categorization, and ways to explain). In Section 6.1.1, we outline the concepts of LLMs and prompt engineering.

6.1.1. Explanations in AI systems

In this section, we describe the differences between explaining the AI models and explaining the outputs. First, we discuss the explanation focus, and then the categorization of explanations based on "what" and "how" to explain.

Explanation Focus

Explanation focus in the context of recommender systems (RS) is based on the part they try to explain: recommendation *input*, recommendation *process*, and recommendation *output* [47, 161, 118]. First, explaining the input aims to clarify the user model which represents the system's understanding of the user's preferences and interests [47, 118]. In other words, this relates to explaining what data is used as input and how the data is collected. Second, explanations of the process entail uncovering the underlying algorithmic logic driving recommendations. Specifically, they clarify how the system processes data to generate recommendations [47, 118]. Last, explaining the output relates to describing why the specific output is given, without focusing on the logic of the system [47, 118].

Understanding the output is often overlooked in algorithmic work within explainable AI (XAI) [87]. However, such understanding helps users evaluate the system's capabilities and utilize it more effectively, particularly in the early stages of using the system [87]. An effective explanation should not only clarify the recommendation but also be meaningful to the intended user [118]. The XAI principle of "Explanation" states that AI systems should deliver supporting evidence for their outputs. Furthermore, the "Meaningful" principle describes that explanations must be understandable to the target audience, highlighting the importance of tailoring information to different user roles and levels of expertise [118]. As was found during the evaluation of the developed PrPM interface (see Section 5.2.3), end-users particularly asked for more explanations of the output. Therefore, in this chapter, we focus on explaining the output.

Table 14. Questions from XAI Question Bank (adapted from [87]), with mapping to explanation focus categories (based on [47, 161, 118]).

Category	Questions (examples)	Explanation focus
Data	* What kind of data does the system learn from? * What is the source of the data?	Input
Output	* What kind of output does the system give? * What does the system output mean?	Output
Performance	* How accurate/precise/reliable are the predictions? * How often does the system make mistakes	Output
How (global)	* How does the system make predictions? * What is the system's overall logic?	Process
Why	* Why is this instance given this prediction? * What feature(s) of this instance lead to the system's prediction?	Process
Why not	* Why is this instance not predicted?	Process
What if	* What would the system predict if this instance changes to ...?	Process
How to be that	* How should this instance change to get a different prediction?	Process
How to still be this	* What is the scope of change permitted to still get the same prediction?	Process
Other	* How to improve the system? (change)	Process
	* What are the results of other people using the system? (social)	Output

When constructing explanations for end users, two questions need to be answered: **what** and **how** to explain [99].

"What" to Explain. Explanations can be categorized into different groups that require a different way of explaining. eXplainable AI Question Bank (XAIQB) is a categorization that provides an overview of such groups based on what questions the users might ask [87, 86]. The groups are *Data, How, Why, Why-not, What-if, How-to, What-else, Output*, and *Performance* explanations. Using XAIQB helps to understand the users' needs for explainability in the specific context (PrPM, in our case), and thus, develop explanations for a broad range of scenarios.

Table 14 provides an overview of XAIQB categories, sample questions, and a matching explanation focus (input, process, or output) for each category.

"How" to Explain. Research on explainability distinguishes between *explanation* and *message* generation [21]. On the one hand, *explanation* generation refers to the different explanator types (e.g., decision trees, salient masks) and structures for an explanation (e.g., feature importance, direct answers). On the other hand, *message* generation concerns the presentation of generated explanations to the user.

For *explanation* generation, XAIQB also provides suggestions on how to explain depending on the explanation category. Table 15 depicts an overview of the categories, sample ways of explanation, and the explanation focus (same as above). The guidance of these ways to explain can be used when designing explanations for a specific purpose.

From the categories, ways to explain, and the explanation focus, it can be seen that explanation generation for input, process, and output is different (as highlighted previously by Guesmi et al. [47]). When providing explanations on input, the target is the user model, as well as the underlying data. For explanations on the recommendation process, the aim is to provide insights into how the algorithm functions (e.g., categories How, Why, Why not, etc.). Generation of such explanations relates to the topic of explainability in PPM and PrPM (see Section 2.3.1). In contrast, explainability of the recommendation output centers around the recommended items themselves (e.g., categories Output, Performance). In the rest of this chapter, we focus on explanations for the recommendation output.

As to the *message* generation, existing presentation forms include graphics, reports, and texts. Texts can be represented as rules or natural language explanations [21]. Natural language explanations are an essential component of future intelligent systems since they are suitable for a wider range of users that posses different domain knowledge [90]. The text presentation can be fixed (e.g., plain text) or interactive (e.g., dialogue system) [21, 24]. In most literature, explanations are presented in a fixed form. In comparison to dialogues, fixed form explanations are not as user-centric since they consist of information that is suitable for a specific type of user [21]. Interactive presentations enable the users to ask free-format questions, and thus, are reflective of their expectations on the explanation [107, 79]. Therefore, interactive presentations make explainable user interfaces more accessible [24].

6.1.2. Large Language Models and Prompt Engineering

Integrating LLMs into prescriptive systems can improve usability by enabling natural language communication and presenting recommendations in plain text, making them more accessible and easier for users to understand [105]. LLMs refers to transformer models pre-trained on large-scale datasets of text that are capable of performing different natural language processing tasks, e.g., text generation [38].

Table 15. Ways to explain the categories from XAI Question Bank (adapted from [87]), with mapping to explanation focus categories (based on [47, 161, 118]).

Category	**Ways to explain (examples)**	**Explanation focus**
Data	* Document comprehensive information about the training data, including the source, type, size, etc.	Input
Output	* Describe the scope of output or system functions * Suggest how the output should be used for downstream tasks or user workflow	Output
Performance	* Provide performance metrics of the model * Describe potential strengths and limitations of the model	Output
How (global)	* Describe the general model logic as feature impact, rules, etc. * If the user is only interested in a high-level view, describe what are the top features or rules considered	Process
Why	* Describe what key features of the instance determine the model's prediction of it * Show similar examples with the same predicted outcome to justify the model's prediction	Process
Why not	* Describe what changes are required for the instance to get the alternative prediction	Process
What if	* Show how the prediction changes corresponding to the inquired change	Process
How to be that	* Highlight features that if changed could alter the prediction	Process
How to still be this	* Describe features that could guarantee the same prediction	Process
Other	No specific guidance, depends on the question	Various

In the field of BPM and process mining, LLMs have been applied to a variety of tasks in the different stages of the BPM lifecycle. As such, a review on LLM applications in BPM [35] brings examples of using LLMs in the discovery phase to generate textual process models from textual descriptions and event logs. Furthermore, for the analysis phase, LLMs are utilized to identify bottlenecks, and in the monitoring phase, to collect information on the process execution and its subsequent analysis [35].

Commonly, adapting a large language model (LLM) to a specific task involves fine-tuning, which requires a substantial dataset of labeled examples tailored to the task [20]. However, this approach has limitations, as the LLM performance depends heavily on the quantity and quality of the examples. Additionally, fine-tuning requires storing a separate copy of the model for each task due to changes in its parameters [20]. An alternative method, known as prompt engineering, has gained popularity. This approach allows task-specific instructions to be provided in a prompt, eliminating the need to modify the LLM itself [20]. Prompt engineering is a process of finding an optimal prompt – a natural language description of instructions – for the LLM to perform the desired task [9].

In the context of process mining, a prompt commonly consists of contextual knowledge, examples, and a task [9, 20]. In the field of process mining, several researchers have already experimented with prompt engineering [46, 61, 20, 9, 81]. To this end, contextual knowledge has been further specified as process mining knowledge (e.g., how an event log is built, definitions of "case"), domain knowledge (e.g., definition of a bottleneck in an event log), data description (e.g., structure, specific calculations) [46, 61]. The output format instructions can also be added to the prompt [46, 61]. This is especially useful when the output is further used by e.g. parsing algorithms [46].

Various strategies for prompt engineering have been discussed in the literature. One approach involves using either a zero-shot or few-shot setting. In a zero-shot setting, the prompt includes only a description of the task, whereas in a few-shot setting, the prompt also provides one or more examples to guide the model [20]. Other research highlights conversational strategies to enhance the style and structure of LLM outputs. For instance, assigning the LLM a specific identity, such as "You are a data scientist", has been shown to improve response quality [148]. Another example is prompting the LLM to use simple language or ask one question at a time [128]. These strategies are also used later in this chapter (referred to in Section 6.3.2).

6.2. Method

In this section, we describe the method to create and evaluate the prompting method. Section 6.2.1 introduces the requirements approach, Section 6.2.2 the design and development, and Section 6.2.3 its evaluation with end-users.

6.2.1. Requirements

To identify requirements for the solution, we started by analyzing the end users' for output explanations in PrPM. With this aim, we took the findings from Kairos evaluation as basis (see Section 5.2.3), and consulted XAIQB to elicit questions the users might have for PrPM context and how to answer them (ways to explain). For example, in the category *Output*, an example question is "What kind of output does the system give?" (Table 14). We contextualized it into "What are the recommendations prescribed by the techniques?". Such a question would help with understanding the differences in recommendations prescribed by Kairos.

We also contextualized the ways to explain. For instance, to answer questions from the *"Output"* category, XAIQB suggests to "describe the scope of output or system functions" and, if applicable, "to suggest how the output should be used for downstream tasks or user workflow" (Table 15). Thus, in the scope of PrPM, a description of recommendation types and their differences could be given. With this in mind, we mapped each category with the possible explanations that could be given as responses. Then, for each question and possible explanations, we formulated a prototypical output that could later be used as examples in the prompt.

6.2.2. Design & Development

We developed a prompting method informed by the requirements elicited in the previous subsection. The initial prompt was constructed based on insights from the literature (see Section 6.1.2) and included several components: context, data description, general conversational rules, task definition, and examples. The examples were derived from the mapping of questions and explanation strategies discussed in the previous subsection.

Then, we submitted the prompt to ChatGPT and asked whether there was any other information required to fulfill the task. From the response, we edited the prompt, added examples, and iteratively tested and improved it based on the feedback. The outcomes of this process are described in Section 6.3.2.

To evaluate the prompting method, we implemented an LLM-based chat on top of Kairos. The design and implementation of this chat are described in detail in Section 6.3.2.

6.2.3. Evaluation

Setting. Participants interacted with the developed chat by asking questions during the evaluation phase. The evaluation used a synthetic event log representing a claim management process. This log comprised 600 cases, 91 case variants, and 4,900 activity instances.

To ensure consistency across interviews, the same case was reviewed each time. This case was ongoing and had a duration of 4 days. The parameters in the PrPM tool were configured to define a positive outcome as completing the process in less than 14 days. In case's current state, the activity "Prepare Claim Settlement" was completed. As a reminder, Kairos produces three different recommendation types (see Section 5.2.2 for detail). The guiding recommendation was to "Approve Claim Settlement", the causality-based was to "Amend Claim Settlement". A correlation-based recommendation was not prescribed in this case (i.e., no alarm raised). The focus of the evaluation was on the explanations of the recommendations provided by the chat and not the recommendations themselves.

Participants. The evaluation had two objectives: (1) assess users' perception of generated explanations based on the prompting method, (2) assess users' interaction with the chat. To address these objectives, a mixed-methods approach was employed, combining contextual interviews and a survey. Contextual interviews, a method derived from contextual design, allow researchers to study the use of technology in its intended context [56]. In this study, the interviews were used to observe participants as they interacted with the chat, aligning with the second objective. This approach provided insights into both the usability of the chat interface and the effectiveness of the explanations during real-time interactions.

The participants in the study were process analysts, either working internally within companies or as consultants. This group was selected because in Kairos evaluation, we saw that operational managers were more interested in the explanation of *how* the recommendation is generated (i.e., explainability techniques). At the same time, process analysts were also asking about explanations of the *output* (see Section 5.2.3). The criteria for the participants were to work with process analysis on a regular basis. Additionally, selecting participants with varying levels of experience and from different domains ensured diverse perceptions. A total of 12 individuals were selected. On average, participants had 6.5 years of experience in process analysis (see Table 16 for an overview). The interviews lasted an average of 21 minutes.

Table 16. Kairos-chat evaluation participants.

Code	Domain	Experience
P-01	Consulting (Finance & Data Analytics)	0.5 years
P-02	Consulting (Process Mining)	7 years
P-03	Software Development (Process Mining)	3 years
P-04	Financial Services	18 years
P-05	Insurance	5 years
P-06	Consulting (Process Mining)	6 years
P-07	Consulting (Process Mining)	1 year
P-08	Software Development (Process Mining)	7 years
P-09	Consumer Goods	7 years
P-10	Consumer Goods	3 years
P-11	Consulting (Process Mining)	1 year
P-12	Consulting (Process Mining)	20 years

Data Collection. During the interview, first, participants were briefed on the goals of the study. They were then provided with a link to the chat to be able to review the case with the provided recommendations on their own. Participants were asked to imagine that the system was newly deployed in an insurance company, and their task was to use the chat to ask questions about the recommendations.

During the interaction, participants were encouraged to think aloud, explaining why they were asking each question and whether the answers provided by the chat were satisfactory. After completing the task, participants were asked to complete a survey to assess their satisfaction with the explanations. The survey was based on the explanation satisfaction scale developed by Hoffman et al. [55]. The survey questions can be found in Appendix C.

Analysis. For each interview, we analyzed two components: (1) the spoken interaction between the participant and the interviewer, and (2) the conversation between the participant and the chat. The participant-chat conversations were saved as separate documents, with annotations capturing our observations for each message exchange. These observations included participants' comments on the chat's responses, such as whether they found them confusing or unhelpful. These annotations informed the coding of the chat's responses for further analysis.

The participants' questions and the chat's responses were analyzed using a coding scheme. For the participants' questions, the coding was based on the categories defined from XAIQB (see Section 6.2.1) (for example, Output, Why, How, etc.). The goal was to investigate what questions the participants asked compared to the questions in XAIQB.

For the chat's responses, we employed a combination of deductive and inductive coding. It is common practice to develop a set of metrics for the explanations that is applicable to the aims of the study [55]. We first reviewed literature on evaluating textual explanations, including systematic reviews on explanation characteristics [55] and their evaluation methods [158, 103]. Then, we analyzed the chat's responses to identify additional codes emerging from the data.

By integrating insights from the literature with our analysis, we defined the following categories for evaluating the chat's explanations: *"Coherency"* – whether the explanation is internally coherent (how well the parts of it fit together), *"Relevancy to the question"* – whether the explanation answers the question, *"Completeness"* – whether there are gaps in the explanation, *"Correctness"* – whether the data in the explanation is correct, and *"Compactness"* – whether the explanation is repetitive or redundant.

Coding was performed independently by the author and another researcher from the paper's authors team (see [69]). They each coded a portion of the dataset, and through multiple rounds arrived at an acceptable agreement score. Cohen's Kappa for questions coding was 0.65 (substantial) and for explanations coding between 0.47-0.5 (moderate).[21] Unlike in the previous evaluations that were rather exploratory, here we involved a second coder due to the more evaluative nature of the study. We also analyzed the survey responses. For each question, we calculated the number of responses in each of the Likert-scale categories. We treated the result as an additional qualitative data point.

6.3. Results

In this section, we describe the results of creating and evaluating the prompting method. Section 6.3.1 introduces the requirements approach, Section 6.3.2 – design and development of the method and its integration into Kairos, and Section 6.3.3 – its evaluation with potential end-users.

6.3.1. Requirements

Following XAIQB (see Section 6.1.1), we contextualized the questions and ways to explain for those for PrPM context. Table 17 presents an excerpt of this mapping.[22] Following the ways to explain guidance, we formulated prototypical output for the textual explanation. In later stages of artifact development, these prototypical outputs are used as examples for the LLM (see Section 6.3.2).

[21] The full coding scheme can be found at: `https://doi.org/10.6084/m9.figshare.25415290.v1`

[22] For full mapping, see supplementary material at link above.

Table 17. [Excerpt] Mapping of XAIQB questions and ways to explain for PrPM context.

Category	Questions	Ways to explain	Prototypical output
Data	What is the size of the event log?	Number of cases in the event log	The event log consists of [number] of cases.
Performance	Why should I believe that the predictions are correct?	Provide performance metrics for the models (accuracy, precision, recall)	The accuracy of recommendations is on average [number].
How	How does the system make predictions?	Describe how the three different techniques work	The tool provides three different recommendation types: next best activity, alarm and intervention. [...] The intervention is produced using Uplift Modeling package CasualLift to get the CATE and probability of outcome if the intervention is applied or not.
Output	What do the different recommendation types mean?	Describe the differences between the techniques	Next activity: A next activity is a type of a recommendation that is prescribed by an algorithm that predicts the next best activity [...]
Further categories and questions are described in supplementary material.			

The mapping was done for all categories of questions in XAIQB. However, as described before in Section 6.1.1, some of the categories require explainability techniques to generate meaningful explanations. Therefore, we formulate general prototypical output also for categories with explanation focus other than "Output", to make sure that the chat can provide the participants with responses if they ask questions from those categories. For example, in Table 17, an example of a question from category *"How"* is brought ("How does the system make predictions?"). As per XAIQB (Table 15), ways to explain for this category are: describe the general model logic as feature impact, rules, etc., or if the user is only interested in a high-level view, describe what the top rules that are considered. Both of these would require explainability techniques. However, to give the user a general idea, a simple description can be provided of how the three different techniques in Kairos work. Similarly, for the category *"Why"*, key features that determine the model's prediction for the instance should be provided, or alternatively, similar examples with the same output. In this case, we implemented a function in the chat that is able to look up similar cases with similar recommendations to be used by the chat as the answer (Section 6.3.2 provides details).

Based on the mapping, we elicited the following functional (FR) and non-functional (NR) requirements:

- FR1: The chat's answers should contain correct data from Kairos outputs.
- FR2: The chat's answers should contain relevant content based on recommendation type.

- NR1: The chat should always provide a response.
- NR2: The chat should respond to the user's question within near-real time.

FR1 refers to the need for the LLM to query correct data from the database to include the relevant data in the prototypical output (e.g., the number of cases). FR2 relates to giving correct information about the techniques that prescribe the different recommendation types (cf. the example of the prototypical output in the category "Output" in Table 17). Thus, the LLM would have to correctly match the information.

6.3.2. Design & Development

Prompting Method

Building on the mapping and requirements outlined in the previous subsection, we developed the prompting method. The initial prompt, drawing from the literature (Section 6.1.2), included the following components: context, data description, general conversational rules, task definition, and examples. Context specified the domain (process mining) and provided details about Kairos, including the techniques employed, the workflow, and the input parameters. Data description outline the structure of the database connected to Kairos (described in the subsequent subsection). The task for to the LLM was to answer questions about Kairos' recommendations and query the database to obtain the required data for the answers. The initial prompt was then tested by directly asking ChatGPT what additional information it needed to serve as an effective explainer module. This feedback led to modifications, such as specifying the conditions under which recommendations were generated, adding information on system configuration, and clarifying user interactions with the recommendations.

To answer users' questions, the LLM needs to query the database where case data and recommendations are stored (FR1). Therefore, subsequent prompt refinement iterations focused on optimizing examples included in the prompt. To determine the optimal representation of query examples in the prompt, we conducted three tests using different variations: #1 no examples, #2 example question and steps for making the query, #3 example question, steps for making the query, and the query itself. We asked the question: *"How often does Kairos make mistakes?"* The results showed that Variations #1 and #2 produced incorrect queries. For example, Variation #1 returned the entire case data. The response from the chat in this variation was *"The accuracy of predictions varies and can be assessed using past performance metrics."*, which was vague and inconsistent. Variation #2 sometimes returned an empty response. Additionally, in both cases, the LLM occasionally queried the wrong collection (e.g., querying the files collection instead of the cases collection). Variation #3, however, produced correct responses (*"The average accuracy over the past instances is 63.00%."*) Based on these findings, we incorporated a question, steps, and a query into the prompting method, since this variation proved to work correctly.

Table 18. [Excerpt] Components of the prompt with text excerpts of each component.

Component	Text (excerpt)
Context	Kairos uses three algorithms to generate prescriptions for business processes [...]. Kairos workflow involves: Uploading an event log. Defining column types. Setting parameters [...]. The key parameters are: Case Completion: An activity that marks the end of a case, e.g., 'Application completed' [...]
Data description	- Description of MongoDB files collection - Description of cases collection
General conversational rules	When answering, use simple language for the explanations. Do not mention the database or show raw data in your responses. [...]

Examples	QUESTION: What is the size of the event log?
	ANSWER: The event log consists of <nr_of_cases> of cases.
	QUERY:
	`collection:'cases',`
	`aggregate:['$match':`
	'event_log_id':<EVENT_LOG_ID>,
	`'$count': 'number_of_cases']`
	STEPS: Run the query with function `query_db` to find the number of cases in this event log.

Task	Your role is to answer questions about Kairos recommendations and query the database for specific case or event log information.

The final iteration involved implementing the refined prompt within an OpenAI assistant, and integrating database query functions for data retrieval. A key improvement was enforcing a conversational rule that prevented the assistant from exposing raw database queries or technical details to users. Table 18 outlines the overall structure of the prompt.[23]

Kairos Integration

To evaluate the prompting method, we developed an LLM-based chat integrated within Kairos (see Figure 19). As a reminder, Kairos provides recommendations that are stored in the database. Kairos displays the generated recommendations for an ongoing case along with the corresponding case attributes.

We used an OpenAI assistant beta feature to create an assistant. It was supplied with a prompt and configured with two functions: `query_db` for querying the database for data relevant for answering questions, and `find_similar_cases`. When the user asks a question (1), the case ID and the question are sent to the backend (2). It then uses the OpenAI API thread endpoint whenever the user creates a new thread (chat).[24] For each question, the backend creates a new run using OpenAI API endpoint (3). The run is configured to include the thread ID (specific to a case) and assistant ID. The backend also configures the run to overwrite the assistant instructions (the prompt) by appending the event log, case structures,

[23]For full prompt and full prompt protocol, see: `https://doi.org/10.6084/m9.figshare.25415290.v1`

[24]The implementation was done in November-December 2023. We used GPT-4 Turbo (specifically, gpt-4-1106-preview).

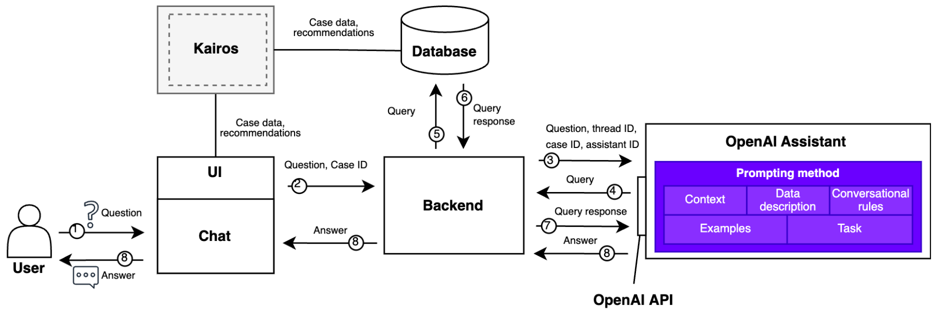

Figure 19. LLM-chat integration into Kairos: the interaction between the user with the chat which displays the answers generated by the LLM based on the prompt.

and their respective IDs to the run instructions. If a question requires querying the database, OpenAI provides the backend with function arguments (4,5) and the backend queries the database (6). The backend then sends the function output to OpenAI (7), which takes the question and function output and produces the answer (8).

Figure 20 shows how the chat was integrated into Kairos. It opens as a chat window at the bottom right of the screen, when the user clicks on the chat icon.

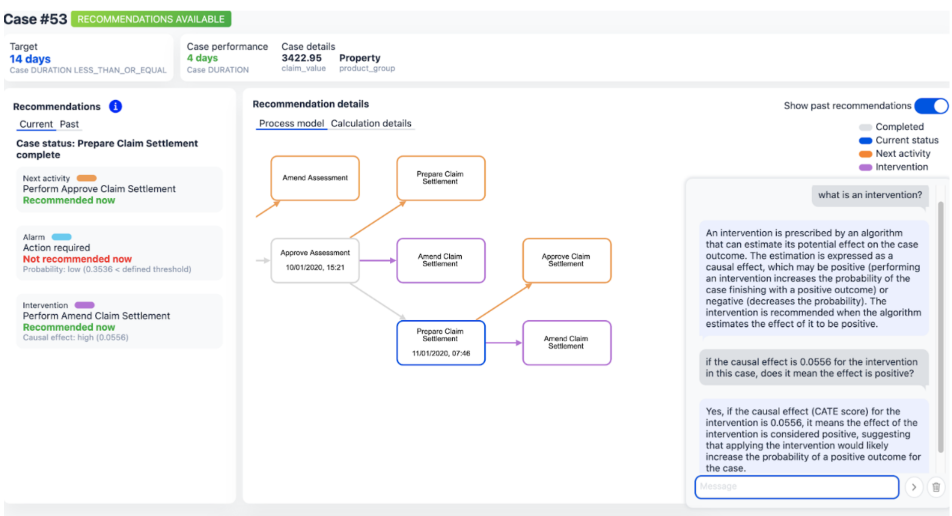

Figure 20. The chat (bottom right) in Kairos interface.

6.3.3. Evaluation

In this section, we introduce the results of the evaluation, divided into three parts: participants' questions, chat's answers, and participant-chat interaction.[25]

[25]Questions asked and answers given can be viewed at: `https://doi.org/10.6084/m9.figshare.25415290.v1`

Participant's Questions

The majority of the participants' questions (55%) focused on the *"Output"* category. These included questions aimed at clarifying PrPM-specific terms, such as "CATE score", "alarm", "positive outcome". Several questions also addressed the recommendation to amend the claim settlement, with participants seeking more precise details on what aspects of the claim settlement should be amended (i.e., not only referring to the activity of changing the claim itself, but also what specifically to change in it). In addition, participants asked questions about case outcomes, such as the last activity in the case and the estimated completion time for the case. Finally, there were questions regarding the potential impact of the recommendations, specifically in terms of measurable outcomes like time savings (e.g., hours) or cost reductions.

The *"Why"* category accounted for 18% of the participants' questions. These questions focused on understanding the rationale behind the prescribed recommendations (thus referring to the explanation focus of recommendation process). Specifically, participants wanted to know why a particular recommendation was suggested in a given case. Given that the case used in the evaluation was displaying two recommendations, participants also inquired about why one recommendation might be preferable to the other.

The category *"How"* recorded 7% of all questions. Here, the participants' questions related to, e.g., how the positive impact of the recommendation can be measured, and whether the recommendations always have a positive effect.

The participants asked comparatively few questions related to the categories of *"What if"* (4%), *"Data"* (2%), and *"How to be that"* (1%). We did not record any questions in categories *"Performance"* (this category refers to the performance of the techniques, e.g., their accuracy), and *"How to still be this."*

Chat's Answers

The chat provided timely responses (NR2), but failed to provide a response once (NR1). However, this was due to an error in the chat's backend. Upon refreshing, the chat worked again.

The analysis of explanation characteristics revealed the following distribution: coherency (98%), relevance to the question (94%), completeness (94%), correctness (75%), and compactness (85%). While coherency, completeness, and relevance were relatively high, correctness and compactness showed room for improvement. Our observations on compactness suggested variability across interviews: in some, most explanations were compact, while in others, they were not. A potential reason for this lies in the chat implementation. Each interaction is recorded as a new message thread, meaning the chat does not have access to responses from earlier interviews. If the chat's first response is long, and the participant does not provide feedback on it, the chat continues to respond similarly.

In regards to correctness, we observed two issues that contributed to incorrect answers. First, in some cases, the chat did not query the database when required but still provided a confident response. For instance, P-06 asked: *"Can you give me statistics about how the duration of the approval of claim settlement correlates with successful and unsuccesful (sic.) claims settlement?"*, to which the chat responded: *"The statistics from the information available show that when the approval of a claim settlement takes 4 to 6 days, the claims are successfully settled. These durations correlate with a positive outcome for the claims that adopted this duration window for settlement approval."* However, the response that it takes 4 to 6 days was not derived from the database. As a reaction, P-06 commented: *"I would still like the more numbers, but I guess [...] it's just what it is."*

Second, sometimes, the chat correctly queried the database but incorrectly matched the retrieved data to the question. For instance, P-09 asked about the probability of the next activity after "Prepare Claim Settlement". The chat responded: *"The probability of the next activity happening after "Prepare Claim Settlement" is 90.14%, as the suggested next activity is "Approve Claim Settlement""*. However, 90.14% related to the accuracy of the prediction, and not the probability. P-09 was confused by the response, because they were not sure whether the probability was for any activity happening or for the activity "Approve Claim Settlement" happening. Therefore, they asked a clarifying question.

The challenges with incorrect answers did not follow specific patterns in terms of the types of queries. While we did not identify specific categories of problematic queries, the collection of queries could be further investigated and refined in future work.

With these results, FR1 is partially fulfilled. For FR2, the chat drew the necessary information from the examples, and its answers were correct.

Participant-Chat Interaction

We observed varying approaches among participants when initiating the conversation with the chat. The majority (8/12) first reviewed the case details and recommendations before formulating questions to clarify specific aspects they found unclear. One participant opened the chat immediately and asked a general question about the case's overall performance before examining the case in detail. Two participants started the conversation by asking about issues in the case they should address. Last, one participant reversed the interaction with the chat by asking how it could help them, basing the next question on its response. Related to the last approach, two other participants suggested adding a feature to the chat interface that would provide pre-defined, clickable questions to streamline the interaction.

On two occasions, the participants disagreed with the chat's response and wrote it in a follow-up message. In this regard, P-01 wrote, *"I think this is not true"*, and P-04 wrote, *"you misled me :-)"*. In both cases, the chat corrected itself and provided the correct data in the response.

Sometimes, the chat had issues in correcting participants when they misinterpreted the data, instead adapting its responses to align with the participants' incorrect assumptions. For example, in the case reviewed in the evaluation, it was displayed that no alarm was raised. However, P-01 misinterpreted the information and asked: *"Why was there an alarm? The probabaility (sic.) was 0.3522."* (The alarm threshold was set at 0.7.) Rather than correcting this misunderstanding, the chat responded: *"An alarm was raised because the probability of a negative outcome calculated by the random forest algorithm was 0.3522. Even though this probability does not exceed the specified threshold that generally triggers an intervention, it still represents a significant risk that may warrant closer attention or a review of the case to prevent an undesirable outcome."* Such chat behavior highlights a need for improvement in the chat's ability to detect and correct misconceptions to ensure that participants receive accurate information.

Participant's Overall Perception of Explanations

The survey was filled out by 10 out of 12 participants, indicating a response rate of 83%. The participants' perception of the explanations was generally positive, with a few being rather neutral (see Figure 21). However, one of the lower-scoring questions was the participants' ability to judge when to trust or not trust Kairos (Q8). This could be related to the correctness of the explanations. Improving the correctness could help enhance trust in the prototype's outputs.

#	The explanations...
Q1	...help me understand how [prototype] works.
Q2	...of how [prototype] works are satisfying.
Q3	...have sufficient detail.
Q4	...seem complete.
Q5	...tell me how to use [prototype].
Q6	...are useful to my goals.
Q7	...show me how accurate [prototype] is.
Q8	...let me judge when I should trust and not trust [prototype].

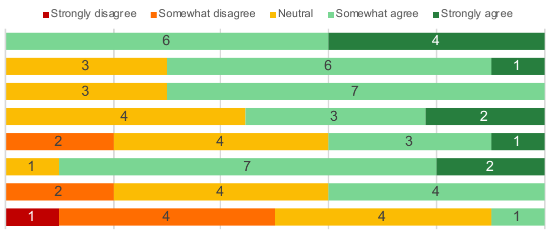

Figure 21. Kairos-chat evaluation: survey results.

6.4. Discussion

Our findings indicate that participants sought detailed information about activity recommendations. For instance, for the recommendation to amend the claim settlement, the participants wanted to know what exactly to amend. This indicates a need for PrPM techniques to take the parameters of the activity, and not only the activity label, into account. Also, several participants asked about the potential impact of a recommendation on the case outcome in temporal or monetary terms. These observations indicate the importance of incorporating causal aspects into PrPM. More specifically, to provide causal recommendations as in e.g., [15, 13, 131]. Thus, future research should improve causal recommendations for PrPM.

Several participants also asked questions about why an action was recommended. This highlights the need to advance explainability in PrPM as well (see Section 2.3.1). One way to address this could be by incorporating causal components into the system. Recent research proposes approaches for generating LLM-based explanations that incorporate causal views, showing execution dependencies among activities in a business process [36]. These causal insights help users understand the underlying reasons for a recommendation. Another approach to addressing "why" questions involves highlighting which case attributes contribute to a given prediction. Techniques such as SHAP values can be used for this purpose, as demonstrated in Galanti et al. [42] and Bozorgi et al. [15]. SHAP provides a clear view of how individual attributes influence the prediction, enhancing interpretability. Counterfactual explanations present yet another method. These allow users to explore hypothetical scenarios by identifying the changes necessary to achieve a desired outcome [58]. Counterfactuals are particularly valuable for actionable insights, as they help users understand the steps required to alter a prediction. Finally, Agarwal et al. [5] show an example of local feature interpretation using the LIME algorithm. With its help, the user receives a textual explanation of why a decision is recommended based on the specific features of the case (e.g., the customer is missing a visa, therefore the travel request should not be approved). Future work could focus on integrating LLMs with these methods for explainability, combining causal views, attribute-based explanations, and counterfactual reasoning.

Prior research indicates that process analysts, even those with a basic understanding of machine learning, may struggle to interpret metric-driven explanations and visualizations effectively [122]. To address this challenge, this work proposes the use of LLM-enhanced explanations as a potential avenue for future research. An experimental design could be developed to compare the understandability of LLM-enhanced explanations with established methods. These established methods could include techniques like SHAP values, which visualize the case attributes influencing predictions [42], tables that show expected KPI values across different scenarios (e.g., no recommendation vs best recommendation) [112], or texts that explain why a decision is not recommended [5]. Furthermore, this experiment could be further extended to compare the effectiveness of "raw" metric-driven visualizations with those supplemented with LLM-based explanations (such as proposed in the previous paragraph).

The findings showed that the participants were mostly satisfied with the explanations and found them useful. However, the evaluation focused on recommendations for individual cases. One future research direction involves expanding the interface to encompass process-level insights. This could include functionalities for viewing aggregated data (e.g., total active recommendations, number of recommended cases) to provide a broader process perspective for analysts, aligning better with their needs as highlighted by [98].

We also observed that many questions were about *Output* and *Why*. Particularly, why a specific recommendation was given in a case. This information could be equally valuable for operational workers who make case-specific decisions [30]. Therefore, another avenue for future research is an evaluation of LLM explanations for case-specific recommendations with operational workers.

Our findings suggest that the correctness of the explanations could be improved. Although the chat corrected itself when being nudged to do that, it is important to secure correct responses in the first place. Specifically, the chat should query the database for each question that requires data to be included in the explanation, ensuring the correct data is retrieved. When designing the prompt, we ran tests with different ways to represent examples to ensure the querying of the database. Additionally, we tested different formulations of the same question (see prompting protocol in the supplementary material in Section 6.3.2). However, we encountered instances where the LLM generated incorrect information, often referred to as "hallucinations." To address this limitation, one potential avenue for future research is the incorporation of a verification layer, as proposed by Ji et al. [62], which could help ensure the accuracy of the chat's responses. Additionally, query validation mechanisms could be implemented to ensure that the generated database queries accurately reflect the user's question. This could involve adding constraints or predefined query templates to prevent misinterpretations. Another factor contributing to these "hallucinations" might be the specific LLM model used. Therefore, another direction for future research would be to conduct an experiment comparing the performance of different LLM models in terms of the correctness of their responses. This could provide valuable insights into model selection and improve the reliability of LLM-based explanations. Another approach would be to investigate criteria that could be used to determine how prompts should be routed to different LLMs based on their strengths and limitations. This would include examining which types of tasks or queries are best suited to specific LLMs. For example, Berti et al. [10] suggest that explaining and interpreting visualizations requires extracting features from images and texts and offering contextual insights. Establishing a structured approach for LLM selection could enhance their overall applicability in PrPM systems.

During the evaluation, individual participants interacted with the chat one at a time. However, in real-world deployments, multiple users may engage with the chat simultaneously, which could impact response times and overall system performance. Future research should investigate how the system handles concurrent interactions. Potential approaches to address performance bottlenecks could include optimizing the database queries, or implementing caching mechanisms for frequently asked questions [59] (i.e., storing frequently requested database query results in memory and retrieving the cached result instead of running a new query).

Several participants suggested adding template questions to the chat, with one asking how the chat could help them. This indicates a wish for guidance on what questions the chat can answer, aligning with [82]. Embedding relevant or frequently asked questions in the chat could be helpful.

To better address questions from the most frequently asked categories, the prompt can be refined to reduce focus on less relevant categories. This adjustment would free up space in the assistant's context window, which, in turn, can be used to provide a more detailed glossary of terms used in the PrPM context.

The importance of incorporating detailed explanations of domain-specific terms in prompts has been emphasized in prior research [61, 81]. Another approach would be to fine-tune the model on examples, and use the prompt solely to describe the context (including a detailed glossary), the data, and to provide the general conversational rules.

We observed that several participants asked questions about case performance, such as cycle time, performance comparison with other cases, and the identification of the “happy path”. This type of data provides contextual information to support understanding of the recommendations. To address this, the prompt could be adapted to include case performance data. However, this would require either enabling the LLM to calculate metrics like cycle time directly or ensuring that the LLM has access to pre-calculated case performance data (e.g., a recent study developed an approach to prompt an LLM to analyze waiting times in an event log and propose redesign options [81]).

6.4.1. Implications

Based on the discussion above, we formulate the following implications:

1. Incorporation of activity parameters in recommendations: PrPM techniques should consider the parameters of an activity, not just the activity label. Detailed recommendations about what needs to be amended, e.g., specific aspects of a claim settlement, should be provided to aid in decision-making.
2. Need for causal aspects in recommendations: Causal aspects should be incorporated into PrPM recommendations since users are interested in understanding the potential impact of a recommendation on case outcomes in terms of temporal or monetary value. Future research should focus on improving causal recommendation capabilities in PrPM.
3. Advancing explainability in PrPM: Users frequently asked “why” a recommendation was made, indicating that PrPM systems should provide explanations from the “process” explanation focus as well. Techniques such as SHAP values and counterfactual reasoning could be applied.
4. Comparing LLM-based explanations and existing methods: Related to above, future work should compare the understandability of LLM enhanced explanations with methods such as SHAP values or counterfactual explanations.
5. Improving the correctness of explanations: A verification layer should be incorporated into the LLM to ensure the correctness of explanations.
6. Adding template questions for user guidance: Adding template questions or frequently asked questions could enhance the user experience by helping users navigate the system more efficiently.
7. Incorporating case performance data: Users asked about case performance, such as cycle time. To address this, the LLM could be adapted to include performance data in its explanations, or be provided with the capability to calculate metrics like cycle time directly.

6.4.2. Limitations

Our study faced several limitations. First, we used a single LLM. Due to its generative nature, LLMs exhibit limited reproducibility. To address this concern, we included participant conversations as supplementary material. Future research could explore applying the same prompt to different LLM models for comparison.

Second, the data used in the evaluation was restricted to a single event log and case. But, the primary focus was on evaluating the quality of the explanations, rather than understanding the recommendations within a broader context.

Third, the selection of interview participants may have introduced recruitment bias. To mitigate this, we recruited participants with diverse experiences and backgrounds. Additionally, to reduce the risk of misinterpreting qualitative data due to bias or subjectivity, we involved two coders and calculated inter-coder reliability.

Finally, the exploratory nature of the study limits the generalizability of findings beyond the specific context investigated. We accepted this limitation, as the study aimed to design and evaluate an approach for LLM-based explanations of the output specifically within the PrPM context.

6.5. Summary

In this chapter, we addressed $\mathbf{RQ}_4$ *How to enhance the understandability of PrPM outputs?* To answer the question, we first elicited user needs for explainability in PrPM using an eXplainable AI Question Bank. Based on these needs, we developed a prompting method for an LLM comprising context, data description, general conversational rules, examples, and task definition. The prompting method was evaluated through an LLM-based chat integrated with a PrPM tool developed in the previous chapter, Kairos.

The implications for research highlight the need for further development of causal recommendations and causal explanations in PrPM. Future studies on explanations in PrPM can leverage the questions identified in our study to better align explanations with end-user needs.

Practically, improvements can be made by incorporating template questions into the chat interface, refining the prompt to focus on the most frequently asked questions, and enabling the system to address queries related to case performance metrics. These enhancements would improve the usability and effectiveness of LLM-based explanations in PrPM.

7. CONCLUSION

In this thesis, we investigate how outputs of PrPM techniques can be made usable for end users. Specifically, we formulate four research questions. In this section, we summarize the answers to each research question and connect them to the contributions.

- **RQ**$_1$ *What are the outputs of PrPM techniques?*

To answer the first research question, we conduct a systematic literature review on PrPM techniques. From this review, we develop a framework that classifies the existing PrPM techniques based on objectives, target metrics, types of recommendations, data inputs, policies to trigger the recommendations, and the user evaluation aspect. As a result, this framework represents *Contribution 1*.

Our analysis reveals that the primary objectives of the existing PrPM techniques are to optimize process outcomes or process efficiency. To achieve these objectives, PrPM techniques provide recommendations that predominantly focus on control flow and resource perspectives. Additionally, recommendations can be categorized based on recommendation frequency (i.e., when recommendations are prescribed) and recommendation purpose (i.e., how recommendations are prescribed). Regarding recommendation frequency, they can be classified as continuous (prescribing recommendations for multiple or all activities of an ongoing case) or discrete (prescribing recommendations only when a need is detected). Regarding the purpose, recommendations are either optimizing or guiding. Optimizing recommendations are aimed at improving performance based on specific KPIs. These recommendations may be based on either correlation or causality. In contrast, guiding techniques provide recommendations based on an analysis of historical traces.

Our review also highlights that PrPM techniques generally rely on inputs from control flow, resource, temporal, and domain-specific data. The policies used to trigger recommendations vary, with guiding techniques typically using similarity-based policies and optimizing techniques employing a diverse range of policies, including rules, maximization of metric improvement, and thresholds for preventing negative outcomes.

Moreover, we identify several research gaps and future directions for advancing PrPM. One of the research gaps is insufficient attention to user-centric design to enhance the usefulness and usability of the outputs of the techniques. We noted that currently, there is little to no attention on applicability of the developed techniques in practice, with only a handful of studies proposing a user interface or evaluating the techniques with potential end-users.

- **RQ**$_2$ *What user groups could benefit from working with the outputs of PrPM techniques? What are the users' information needs?*

To answer the second research question, we identify an initial set of information items for a PrPM interface, evaluate them with experts through a wireframe, and refine the design based on insights from the evaluation. Specifically, we first analyze existing tools and the PrPM domain (based on the systematic literature review) to gather an initial set of information items. For the domain analysis, we use the results of the previous RQ. Combining the analyses, we categorize the initial information items into four main groups: Case Description, Recommendation Description, Recommendation Explanation, and Resource Assignment.

Using these information items, we create a wireframe and conduct an evaluation with experts in the field. The evaluation confirms the relevance of the identified information items. However, it also reveals that end users consist of three distinct groups: operational workers (OW), operational managers (OM), and process analysts (PA). They each have different information needs. Specifically, certain information items are found to be more relevant to some user groups than to others (e.g., process model for process analysts but not for operational workers). Based on the evaluation results, we refine the information items to meet the specific requirements of each user group working with the outputs of PrPM techniques. As such, the refined information items are categorized into four groups: *(i)* Case Description, *(ii)* Recommendation Description, *(iii)* Recommendation Explanation, and *(iv)* Resource Assignment. The information items of group *i* (KPI status, case-specific attributes) are relevant for all end-user groups, while the process model is only relevant for PA. In terms of group *ii*, for all end-user groups it is important to provide the list of possible options, while PAs need recommendation characteristics in addition (type, causal effect or probability). The third group *(iii)* consists of the predicted effect (all), history of past recommendations (all), similar cases (OW, PA), and model description (OM, PA). Group *iv* describes resources and their performance and is relevant to managers.

- **RQ**$_3$ *How to translate these information needs into a user interface design for PrPM systems?*

Next, we use the information items previously identified to design and develop a web-based tool for the output of PrPM techniques called Kairos. Then, we evaluate Kairos with process analysts and operational managers. The feedback reveals that the users generally perceive the tool as useful and usable, but several areas for improvement are highlighted. Suggestions include the addition of aggregated overviews of completed and ongoing cases for PAs, automatic prioritization techniques and their explanations for both end-user groups, and information regarding case statuses for each resource for operational managers.

Based on the evaluation, we refine the information items. For both interviewed groups, it is important to display a clearly indicated KPI to be optimized, alongside aggregated data on cases and available recommendations. The level of detail in recommendations for individual cases should be adjusted according to the user group. Additionally, as highlighted before, PAs require a process model. We refine the information item to specifically include an overlay of recommendations in additions, which allows to visually see the possible next steps. The explanations for the outputs of PrPM techniques should be designed in a way that is accessible to users with limited technical knowledge. For OMs, an overview of resources, including workload, performance in terms of KPIs, and case statuses, is needed.

Finally, we gather a set of recommendations for designing PrPM. Namely, they emphasize the importance of personalized interfaces, the need for clear explanations, interactive data exploration, and support for decision-making. In this way, RQ2 and RQ3 address *Contribution 2*.

- **RQ**$_4$ *How to enhance the understandability of PrPM outputs?*

To address this question, we develop a prompting method for an LLM that enables elaborating and explaining the outputs of PrPM techniques (*Contribution 3*). We first identify user needs for explainability in PrPM by contextualizing the questions from an eXplainable AI Question Bank to PrPM context. The result is a mapping of the questions with possible answers to them (i.e., possible options for an explanation). Based on this mapping, we design and develop a prompting method that enables an LLM to elaborate on and explain PrPM recommendations. The method includes context, data descriptions, general conversational rules, examples, and task definitions. We implement the method into a chat in the previously developed tool Kairos. Then, we conduct an evaluation with end users, where they interact with the chat and ask questions about the case and the recommendations they were seeing in Kairos. We analyze the questions the users asked, the answers the chat gave, and the interaction between the user and the chat. The users found that such conversations help them better understand how to use Kairos, and conducting such conversations is useful to their goals. The evaluation results also show that the correctness of the answers should be improved. The majority of questions the users asked were about the output (e.g., to understand the specific terms better). However, they also wanted to know why a specific recommendation was given in a case.

Based on the results of the evaluation, we formulate research implications. They emphasize the need for further advancement of causal recommendations and causal explanations in PrPM to better answer the questions of why a recommendation is given and how it affects a case. Future research on explainability in PrPM can build upon the questions identified in our study to better tailor explanations to the needs of end users.

The research results presented in this thesis offer several directions for future work. Namely, future work aims to expand the evaluation scope, improve Kairos functionality, enhance the correctness of LLM-based explanations, and incorporate additional explainability methods. Specifically:

- We conduct Kairos evaluation only in one domain (financial services). Future work should expand the evaluation by using event logs from diverse domains. This would allow for a broader understanding of how the information items perform across different contexts and how domain-specific process attributes might influence the effectiveness of the information items.

- Furthermore, as our evaluation results with the operational managers show, the applicability of Kairos in the financial domain might be limited due to compliance requirements. Namely, in certain processes (as loan application), banks have to follow strict procedures and therefore, it might be challenging for them to apply recommendations that diverge from the procedure. Therefore, evaluations in other domains with fewer compliance constraints could provide valuable insights into the broader applicability and effectiveness of PrPM systems.
- Additionally, we evaluate the tool with two out of three user groups. The information items for the operational workers user group should be validated.
- The developed prompting method aims at elaborating on an explaining the outputs of PrPM techniques. However, LLMs could also be applied to make other parts of Kairos more useful and usable. Our current implementation of the prompting method is in a chat, where the user asks a question and gets a response. We could implement a proactive notification approach. One example is to summarize the "most important" (i.e., prioritized) cases for operational managers that need to be addressed by their intervention.
- There is also a clear need to incorporate explainability methods that can explain how the models work. These methods could include approaches such as SHAP values, which visualize case attributes influencing predictions, tables displaying expected KPI values when following or rejecting a recommendation, or textual explanations clarifying why certain decisions are not recommended.
- We also only evaluate the users' perception of the explanations provided by the LLM, without comparing it to other methods. A direction for future work is to design an experiment that compares the understandability of LLM-enhanced explanations with other explanation methods, as exemplified in the point above. Additionally, this experiment could be extended to compare the effectiveness of "raw" metric-driven visualizations with those augmented by LLM-based explanations, to assess whether the inclusion of LLM-driven insights enhances user understanding and decision-making.
- In the prompting method evaluation, we use only one LLM. In future work, we could explore the use of multiple LLM models by applying the same prompt across different models to compare their performance. This would help assess the variability in results due to the generative nature of LLMs and improve the reliability and reproducibility of the findings. Additionally, further research could investigate techniques to enhance the reproducibility of LLM-based interactions in similar contexts.

BIBLIOGRAPHY

[1] Wil M. P. van der Aalst. “Process Mining”. In: *Commun. ACM* 55.8 (Aug. 2012), pp. 76–83. ISSN: 0001-0782.

[2] Wil M. P. van der Aalst, Martin Bichler, and Armin Heinzl. “Responsible Data Science”. In: *Bus. Inf. Syst. Eng.* 59.5 (2017), pp. 311–313.

[3] Chadia Abras, Diane Maloney-Krichmar, Jenny Preece, et al. “User-centered design”. In: *Bainbridge, W. Encyclopedia of Human-Computer Interaction. Thousand Oaks: Sage Publications* 37.4 (2004), pp. 445–456.

[4] Adekunle Oluseyi Afolabi and Pekka Toivanen. “Improving the design of a recommendation system using evaluation criteria and metrics as a guide”. In: *J. Syst. Inf. Technol.* 21.3 (2019), pp. 304–324. DOI: 10.1108/JSIT-01-2019-0019.

[5] Prerna Agarwal et al. “A Process-Aware Decision Support System for Business Processes”. In: *KDD*. ACM, 2022, pp. 2673–2681.

[6] Simone Agostinelli et al. “A User Evaluation of Process Discovery Algorithms in a Software Engineering Company”. In: *EDOC*. IEEE, 2019, pp. 142–150. DOI: 10.1109/EDOC.2019.00026.

[7] Anti Alman et al. “Tiramisù: making sense of multi-faceted process information through time and space”. In: *Journal of Intelligent Information Systems* (2024), pp. 1–23.

[8] Irene Barba, Barbara Weber, and Carmelo Del Valle. “Supporting the Optimized Execution of Business Processes through Recommendations”. In: *Business Process Management Workshops (1)*. Springer, 2011, pp. 135–140.

[9] Patrizio Bellan, Mauro Dragoni, and Chiara Ghidini. “Extracting Business Process Entities and Relations from Text Using Pre-trained Language Models and In-Context Learning”. In: *EDOC*. Vol. 13585. LNCS. Springer, 2022, pp. 182–199.

[10] Alessandro Berti et al. “Evaluating Large Language Models in Process Mining: Capabilities, Benchmarks, and Evaluation Strategies”. In: *BP-MDS/EMMSAD@CAiSE*. Vol. 511. Lecture Notes in Business Information Processing. Springer, 2024, pp. 13–21.

[11] Adream Blair-Early and Mike Zender. “User interface design principles for interaction design”. In: *Design Issues* 24.3 (2008), pp. 85–107.

[12] Zahra Dasht Bozorgi et al. “Prescriptive Process Monitoring for Cost-Aware Cycle Time Reduction”. In: *ICPM*. IEEE, 2021, pp. 96–103. DOI: 10.1109/ICPM53251.2021.9576853.

[13] Zahra Dasht Bozorgi et al. “ProLift: Automated Discovery of Causal Treatment Rules From Event Logs (Extended Abstract)”. In: *ICPM Doctoral Consortium / Demo*. Vol. 3299. CEUR Workshop Proceedings. CEUR-WS.org, 2022, pp. 108–112.

[14] Zahra Dasht Bozorgi et al. "Learning When to Treat Business Processes: Prescriptive Process Monitoring with Causal Inference and Reinforcement Learning". In: *CAiSE*. Vol. 13901. Lecture Notes in Computer Science. Springer, 2023, pp. 364–380. DOI: 10.1007/978-3-031-34560-9_22.

[15] Zahra Dasht Bozorgi et al. "Prescriptive process monitoring based on causal effect estimation". In: *Inf. Syst.* 116 (2023), p. 102198. DOI: 10.1016/j.is.2023.102198.

[16] Stefano Branchi et al. "Learning to Act: A Reinforcement Learning Approach to Recommend the Best Next Activities". In: *BPM (Forum)*. Vol. 458. Lecture Notes in Business Information Processing. Springer, 2022, pp. 137–154.

[17] Virginia Braun and Victoria Clarke. "Using thematic analysis in psychology". In: *Qualitative research in psychology* 3.2 (2006), pp. 77–101.

[18] Andrei Buliga et al. "Nirdizati Light: A Modular Framework for Explainable Predictive Process Monitoring". In: *BPM (Demos / Resources Forum)*. Vol. 3758. CEUR Workshop Proceedings. CEUR-WS.org, 2024, pp. 111–115.

[19] Andrei Buliga et al. "Uncovering Patterns for Local Explanations in Outcome-Based Predictive Process Monitoring". In: *BPM*. Vol. 14940. Lecture Notes in Computer Science. Springer, 2024, pp. 363–380.

[20] Kiran Busch et al. "Just Tell Me: Prompt Engineering in Business Process Management". In: *BPMDS/EMMSAD@CAiSE*. Vol. 479. LNBIP. Springer, 2023, pp. 3–11.

[21] Erik Cambria et al. "A survey on XAI and natural language explanations". In: *Inf. Process. Manag.* 60.1 (2023), p. 103111.

[22] David Chapela-Campa and Marlon Dumas. "From process mining to augmented process execution". In: *Softw. Syst. Model.* 22.6 (2023), pp. 1977–1986.

[23] Li Chen et al. "Human Decision Making and Recommender Systems". In: *ACM Trans. Interact. Intell. Syst.* 3.3 (2013), 17:1–17:7.

[24] Michael Chromik and Andreas Butz. "Human-XAI Interaction: A Review and Design Principles for Explanation User Interfaces". In: *INTERACT (2)*. Vol. 12933. Lecture Notes in Computer Science. Springer, 2021, pp. 619–640. DOI: 10.1007/978-3-030-85616-8_36.

[25] Claudio Di Ciccio et al. "Human in the (Process) Mines (Dagstuhl Seminar 23271)". In: *Dagstuhl Reports* 13.7 (2023), pp. 1–33.

[26] Christian Colceriu et al. "From Task Analysis to Wireframe Design: An Approach to User-Centered Design of a GUI for Mobile HRI at Assembly Workplaces". In: *RO-MAN*. IEEE, 2022, pp. 876–883. DOI: 10.1109/RO-MAN53752.2022.9900679.

[27] Raffaele Conforti et al. "A recommendation system for predicting risks across multiple business process instances". In: *Decis. Support Syst.* 69 (2015), pp. 1–19.

[28] Christine M. Cutillo et al. "Machine intelligence in healthcare - perspectives on trustworthiness, explainability, usability, and transparency". In: *npj Digit. Medicine* 3 (2020).

[29] Fred D Davis. "Perceived usefulness, perceived ease of use, and user acceptance of information technology". In: *MIS quarterly* (1989), pp. 319–340.

[30] Marcus Dees et al. "What if process predictions are not followed by good recommendations?" In: *BPM (Industry Forum)*. Vol. 2428. CEUR Workshop Proceedings. CEUR-WS.org, 2019, pp. 61–72.

[31] Ivan Donadello et al. "Outcome-Oriented Prescriptive Process Monitoring based on Temporal Logic Patterns". In: *Eng. Appl. Artif. Intell.* 126 (2023), p. 106899.

[32] Marlon Dumas et al. *Fundamentals of Business Process Management, Second Edition*. Springer, 2018. DOI: 10.1007/978-3-662-56509-4.

[33] Sebastian Dunzer et al. "Conformance checking: a state-of-the-art literature review". In: *S-BPM ONE*. ACM, 2019, 4:1–4:10.

[34] Gamal Elkoumy et al. "Privacy and Confidentiality in Process Mining: Threats and Research Challenges". In: *ACM Trans. Manag. Inf. Syst.* 13.1 (2022), 11:1–11:17.

[35] Bedilia Estrada-Torres, Adela del-Río-Ortega, and Manuel Resinas. "Mapping the Landscape: Exploring Large Language Model Applications in Business Process Management". In: *BPMDS/EMMSAD@CAiSE*. Vol. 511. Lecture Notes in Business Information Processing. Springer, 2024, pp. 22–31.

[36] Dirk Fahland et al. "How well can a large language model explain business processes as perceived by users?" In: *Data Knowl. Eng.* 157 (2025), p. 102416.

[37] Stephan A. Fahrenkrog-Petersen et al. "Fire now, fire later: alarm-based systems for prescriptive process monitoring". In: *Knowl. Inf. Syst.* 64.2 (2022), pp. 559–587. DOI: 10.1007/s10115-021-01633-w.

[38] Stefan Feuerriegel et al. "Generative AI". In: *Bus Inf Sys Eng* (2023).

[39] Chiara di Francescomarino et al. "Predictive process monitoring methods: Which one suits me best?" In: *International Conference on Business Process Management*. Springer. 2018, pp. 462–479. DOI: 10.1007/978-3-319-98648-7_27.

[40] Chiara Di Francescomarino and Chiara Ghidini. "Predictive Process Monitoring". In: *Process Mining Handbook*. Vol. 448. Lecture Notes in Business Information Processing. Springer, 2022, pp. 320–346.

[41] Riccardo Galanti et al. “Explainable Predictive Process Monitoring”. In: *ICPM*. IEEE, 2020, pp. 1–8.

[42] Riccardo Galanti et al. “An explainable decision support system for predictive process analytics”. In: *Eng. Appl. Artif. Intell.* 120 (2023), p. 105904.

[43] Arash Yadegari Ghahderijani and Dimka Karastoyanova. “Autonomic Process Performance Improvement”. In: *EDOC Workshops*. IEEE, 2021, pp. 299–307.

[44] Johny Ghattas, Pnina Soffer, and Mor Peleg. “Improving business process decision making based on past experience”. In: *Decis. Support Syst.* 59 (2014), pp. 93–107. DOI: 10.1016/j.dss.2013.10.009.

[45] Christoph Gröger, Holger Schwarz, and Bernhard Mitschang. “Prescriptive Analytics for Recommendation-Based Business Process Optimization”. In: *BIS*. Vol. 176. Lecture Notes in Business Information Processing. Springer, 2014, pp. 25–37.

[46] Michael Grohs et al. “Large Language Models Can Accomplish Business Process Management Tasks”. In: *Business Process Management Workshops*. Vol. 492. LNBIP. Springer, 2023, pp. 453–465.

[47] Mouadh Guesmi et al. “Input or Output: Effects of Explanation Focus on the Perception of Explainable Recommendation with Varying Level of Details”. In: *IntRS@RecSys*. Vol. 2948. CEUR Workshop Proceedings. CEUR-WS.org, 2021, pp. 55–72.

[48] Christian Haisjackl and Barbara Weber. “User Assistance during Process Execution - An Experimental Evaluation of Recommendation Strategies”. In: *Business Process Management Workshops*. Vol. 66. Lecture Notes in Business Information Processing. Springer, 2010, pp. 134–145.

[49] Matthew J Hamm. *Wireframing essentials*. Packt Publishing Ltd, 2014. ISBN: 1849698546.

[50] Margaret C Harrell and Melissa A Bradley. *Data collection methods. Semi-structured interviews and focus groups*. Tech. rep. Rand National Defense Research Institute, Santa Monica, USA, 2009.

[51] Lukas-Valentin Herm et al. “A nascent design theory for explainable intelligent systems”. In: *Electron. Mark.* 32.4 (2022), pp. 2185–2205. DOI: 10.1007/s12525-022-00606-3.

[52] Julia Hermann et al. “Watt’s Next? Leveraging Process Flexibility for Power Cost Optimization”. In: *Bus. Inf. Syst. Eng.* 66.5 (2024), pp. 541–563.

[53] Thomas Herrmann and Alexander Nolte. “Promoting the Adoption of AI-Based Recommendations Through Organizational Practices”. In: *Navigating Digital Transformation: Organizational Change, Digital Work, and Individual Behavior*. Springer, 2024, pp. 195–212.

[54] Alan R. Hevner et al. “Design Science in Information Systems Research”. In: *MIS Q.* 28.1 (2004), pp. 75–105.

[55] Robert R. Hoffman et al. “Measures for explainable AI: Explanation goodness, user satisfaction, mental models, curiosity, trust, and human-AI performance”. In: *Frontiers Comput. Sci.* 5 (2023).

[56] Karen Holtzblatt. “Contextual design”. In: *The human-computer interaction handbook*. CRC press, 2007, pp. 975–990.

[57] Jan Horsky et al. “Interface design principles for usable decision support: A targeted review of best practices for clinical prescribing interventions”. In: *J. Biomed. Informatics* 45.6 (2012), pp. 1202–1216. DOI: 10.1016/j.jbi.2012.09.002.

[58] Chihcheng Hsieh, Catarina Moreira, and Chun Ouyang. “DiCE4EL: Interpreting Process Predictions using a Milestone-Aware Counterfactual Approach”. In: *ICPM*. IEEE, 2021, pp. 88–95.

[59] Cunchen Hu et al. “Memserve: Context caching for disaggregated llm serving with elastic memory pool”. In: *arXiv preprint arXiv:2406.17565* (2024).

[60] Sebastian Huber, Marian Fietta, and Sebastian Hof. “Next step recommendation and prediction based on process mining in adaptive case management”. In: *S-BPM ONE*. ACM, 2015, 3:1–3:9.

[61] Urszula Jessen, Michal Sroka, and Dirk Fahland. “Chit-Chat or Deep Talk: Prompt Engineering for Process Mining”. In: *CoRR* abs/2307.09909 (2023).

[62] Ziwei Ji et al. “Towards Mitigating LLM Hallucination via Self Reflection”. In: *Findings of the Assoc. for Comp. Ling.: EMNLP 2023*. Ed. by Houda Bouamor, Juan Pino, and Kalika Bali. Singapore: ACL, 2023, pp. 1827–1843.

[63] Tingting Jiang et al. “Human-AI interaction research agenda: A user-centered perspective”. In: *Data and Information Management* (2024), p. 100078.

[64] Kerwin Jorbina et al. “Nirdizati: A Web-Based Tool for Predictive Process Monitoring”. In: *BPM (Demos)*. Vol. 1920. CEUR Workshop Proceedings. CEUR-WS.org, 2017.

[65] Jay F. Nunamaker Jr., Minder Chen, and Titus D. M. Purdin. “Systems Development in Information Systems Research”. In: *J. Manag. Inf. Syst.* 7.3 (1991), pp. 89–106.

[66] **Kateryna Kubrak** et al. “Prescriptive process monitoring: *Quo vadis*?” In: *PeerJ Comput. Sci.* 8 (2022), e1097. DOI: 10.7717/PEERJ-CS.1097.

[67] **Kateryna Kubrak** et al. “Design and Evaluation of a User Interface Concept for Prescriptive Process Monitoring”. In: *CAiSE*. Vol. 13901. Lecture Notes in Computer Science. Springer, 2023, pp. 347–363. DOI: 10.1007/978-3-031-34560-9_21.

[68] **Kateryna Kubrak** et al. “Kairos: A Tool for Prescriptive Monitoring of Business Processes”. In: *ICPM Doctoral Consortium / Demo*. Vol. 3648. CEUR Workshop Proceedings. CEUR-WS.org, 2023. URL: `https://ceur-ws.org/Vol-3648/paper%5C_214.pdf`.

[69] **Kateryna Kubrak** et al. “Explanatory Capabilities of Large Language Models in Prescriptive Process Monitoring”. In: *BPM*. Vol. 14940. Lecture Notes in Computer Science. Springer, 2024, pp. 403–420. DOI: `10.1007/978-3-031-70396-6\_23`.

[70] **Kateryna Kubrak** et al. “Supporting Analysts and Managers to Utilize Prescriptive Process Monitoring: A User Interface Design and Evaluation”. In: *ACM Transactions on Management Information Systems* (Under Review).

[71] Asjad Khan et al. “DeepProcess: Supporting Business Process Execution Using a MANN-Based Recommender System”. In: *ICSOC*. Vol. 13121. Lecture Notes in Computer Science. Springer, 2021, pp. 19–33.

[72] Aekyung Kim, Josué Obregon, and Jae-Yoon Jung. “Constructing Decision Trees from Process Logs for Performer Recommendation”. In: *Business Process Management Workshops*. Vol. 171. Lecture Notes in Business Information Processing. Springer, 2013, pp. 224–236.

[73] Barbara Kitchenham. “Procedures for performing systematic reviews”. In: *Keele, UK, Keele University* 33.2004 (2004), pp. 1–26.

[74] Christopher Klinkmüller, Richard Müller, and Ingo Weber. “Mining Process Mining Practices: An Exploratory Characterization of Information Needs in Process Analytics”. In: *BPM*. Vol. 11675. Lecture Notes in Computer Science. Springer, 2019, pp. 322–337.

[75] Ron Kohavi and Roger Longbotham. “Online Controlled Experiments and A/B Testing”. In: *Encyclopedia of Machine Learning and Data Mining*. Springer, 2017, pp. 922–929.

[76] Kateryna Kubrak, Fredrik Milani, and Alexander Nolte. “Process Mining for Process Improvement - An Evaluation of Analysis Practices”. In: *RCIS*. Vol. 446. Lecture Notes in Business Information Processing. Springer, 2022, pp. 214–230. DOI: `10.1007/978-3-031-05760-1\_13`.

[77] Kateryna Kubrak, Fredrik Milani, and Alexander Nolte. “A visual approach to support process analysts in working with process improvement opportunities”. In: *Business Process Management Journal* 29.8 (2023), pp. 101–132. DOI: `10.1108/BPMJ-10-2021-0631`.

[78] William Kuechler and Vijay Vaishnavi. “The emergence of design research in information systems in North America”. In: *Journal of Design Research* 7.1 (2008), pp. 1–16.

[79] Samuli Laato et al. “How to explain AI systems to end users: a systematic literature review and research agenda”. In: *Internet Res.* 32.7 (2022), pp. 1–31.

[80] Julius Peter Landwehr et al. "Design Knowledge for Deep-Learning-Enabled Image-Based Decision Support Systems". In: *Bus. Inf. Syst. Eng.* 64.6 (2022), pp. 707–728.

[81] Katsiaryna Lashkevich et al. "LLM-Assisted Optimization of Waiting Time in Business Processes: A Prompting Method". In: *BPM*. Vol. 14940. Lecture Notes in Computer Science. Springer, 2024, pp. 474–492.

[82] Yoonjoo Lee et al. "DAPIE: Interactive Step-by-Step Explanatory Dialogues to Answer Children's Why and How Questions". In: *CHI*. ACM, 2023, 450:1–450:22.

[83] Janna-Liina Leemets et al. "Persuasive Visual Presentation of Prescriptive Business Processes". In: *RCIS*. Vol. 476. Lecture Notes in Business Information Processing. Springer, 2023, pp. 398–414.

[84] Massimiliano de Leoni, Marcus Dees, and Laurens Reulink. "Design and Evaluation of a Process-aware Recommender System based on Prescriptive Analytics". In: *2nd International Conference on Process Mining, ICPM*. IEEE, 2020, pp. 9–16. DOI: 10.1109/ICPM49681.2020.00013.

[85] Katerina Lepenioti et al. "Prescriptive analytics: Literature review and research challenges". In: *Int. J. Inf. Manag.* 50 (2020), pp. 57–70.

[86] Q. Vera Liao. "Question-Driven eXplainable AI: Re-framing the Technical and Design Spaces of XAI". In: *DaSH@KDD*. 2021.

[87] Q. Vera Liao, Daniel M. Gruen, and Sarah Miller. "Questioning the AI: Informing Design Practices for Explainable AI User Experiences". In: *CHI*. ACM, 2020, pp. 1–15.

[88] Monika Malinova Mandelburger and Jan Mendling. "Cognitive diagram understanding and task performance in systems analysis and design". In: *MIS Quarterly* 45.4 (2021), pp. 2101–2157.

[89] Felix Mannhardt. "Responsible Process Mining". In: *Process Mining Handbook*. Vol. 448. Lecture Notes in Business Information Processing. Springer, 2022, pp. 373–401.

[90] Ettore Mariotti, José Maria Alonso, and Albert Gatt. "Towards Harnessing Natural Language Generation to Explain Black-box Models". In: *NL4XAI@INGL*. Association for Computational Linguistics, 2020, pp. 22–27.

[91] Alfonso Eduardo Márquez-Chamorro, Manuel Resinas, and Antonio Ruiz-Cortés. "Predictive Monitoring of Business Processes: A Survey". In: *IEEE Transactions on Services Computing* 11.6 (2018), pp. 962–977. DOI: 10.1109/TSC.2017.2772256.

[92] Nijat Mehdiyev and Peter Fettke. "Explainable Artificial Intelligence for Process Mining: A General Overview and Application of a Novel Local Explanation Approach for Predictive Process Monitoring". In: *CoRR* abs/2009.02098 (2020).

[93] Nijat Mehdiyev, Maxim Majlatow, and Peter Fettke. "Communicating Uncertainty in Machine Learning Explanations: A Visualization Analytics Approach for Predictive Process Monitoring". In: *xAI (3)*. Vol. 2155. Communications in Computer and Information Science. Springer, 2024, pp. 420–438.

[94] Steven Mertens and Frederik Gailly. "Supporting Physicians Through Prescriptive Process Monitoring". In: *ICHI*. IEEE, 2023, pp. 12–20.

[95] Andreas Metzger, Tristan Kley, and Alexander Palm. "Triggering Proactive Business Process Adaptations via Online Reinforcement Learning". In: *BPM*. Vol. 12168. Lecture Notes in Computer Science. Springer, 2020, pp. 273–290.

[96] Andreas Metzger et al. "Automatically reconciling the trade-off between prediction accuracy and earliness in prescriptive business process monitoring". In: *Inf. Syst.* 118 (2023), p. 102254.

[97] Fredrik Milani and Fabrizio Maria Maggi. "A Comparative Evaluation of Log-Based Process Performance Analysis Techniques". In: *BIS*. Vol. 320. Lecture Notes in Business Information Processing. Springer, 2018, pp. 371–383.

[98] Fredrik Milani et al. "Process Mining: A Guide for Practitioners". In: *RCIS*. Vol. 446. Lecture Notes in Business Information Processing. Springer, 2022, pp. 265–282.

[99] Sina Mohseni, Niloofar Zarei, and Eric D. Ragan. "A Multidisciplinary Survey and Framework for Design and Evaluation of Explainable AI Systems". In: *ACM Trans. Interact. Intell. Syst.* 11.3-4 (2021), 24:1–24:45.

[100] Tamara Munzner. *Visualization Analysis and Design*. A.K. Peters visualization series. A K Peters, 2014. ISBN: 978-1-466-50891-0.

[101] Joyce Nakatumba, Michael Westergaard, and Wil MP van der Aalst. "A meta-model for operational support". In: *BPM Center Report BPM-12-05, BPMcenter. org* (2012), pp. 16–32.

[102] Soroosh Nalchigar and Eric S. K. Yu. "Business-driven data analytics: A conceptual modeling framework". In: *Data Knowl. Eng.* 117 (2018), pp. 359–372.

[103] Meike Nauta et al. "From Anecdotal Evidence to Quantitative Evaluation Methods: A Systematic Review on Evaluating Explainable AI". In: *ACM Comp. Surv.* 55.13s (2023), 295:1–295:42.

[104] Hamid R. Motahari Nezhad and Claudio Bartolini. "Next Best Step and Expert Recommendation for Collaborative Processes in IT Service Management". In: *BPM*. Vol. 6896. Lecture Notes in Computer Science. Springer, 2011, pp. 50–61.

[105] Marvin Niederhaus et al. "Technical Readiness of Prescriptive Analytics Platforms: A Survey". In: *35th Conference of Open Innovations Association, FRUCT 2024, Tampere, Finland, April 24-26, 2024*. IEEE, 2024,

pp. 509–519. URL: https://ieeexplore.ieee.org/document/10516367.

[106] Jakob Nielsen. "Enhancing the explanatory power of usability heuristics". In: *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*. 1994, pp. 152–158.

[107] Muhammad Suffian Nizami et al. "Investigating Human-Centered Perspectives in Explainable Artificial Intelligence". In: *XAI.it@AI*IA*. Vol. 3518. CEUR Workshop Proceedings. CEUR-WS.org, 2023, pp. 47–66.

[108] Donald A Norman. "Design principles for human-computer interfaces". In: *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*. 1983, pp. 1–10.

[109] Vimala Nunavath, Andreas Prinz, and Tina Comes. "Identifying First Responders Information Needs: Supporting Search and Rescue Operations for Fire Emergency Response". In: *Int. J. Inf. Syst. Crisis Response Manag.* 8.1 (2016), pp. 25–46. DOI: 10.4018/IJISCRAM.2016010102.

[110] Chitu Okoli and Kira Schabram. "A Guide to Conducting a Systematic Literature Review of Information Systems Research". In: *SSRN Electronic Journal* 10 (2010).

[111] Alessandro Padella and Massimiliano de Leoni. "Resource Allocation in Recommender Systems for Global KPI Improvement". In: *BPM (Forum)*. Vol. 490. Lecture Notes in Business Information Processing. Springer, 2023, pp. 249–266.

[112] Alessandro Padella et al. "Explainable Process Prescriptive Analytics". In: *ICPM*. IEEE, 2022, pp. 16–23.

[113] Alessandro Padella et al. "Experience-Based Resource Allocation for Remaining Time Optimization". In: *BPM*. Vol. 14940. Lecture Notes in Computer Science. Springer, 2024, pp. 345–362.

[114] Gyunam Park, Daniel Schuster, and Wil M. P. van der Aalst. "Pattern-based action engine: Generating process management actions using temporal patterns of process-centric problems". In: *Comput. Ind.* 153 (2023), p. 104020.

[115] Gyunam Park and Minseok Song. "Prediction-based Resource Allocation using LSTM and Minimum Cost and Maximum Flow Algorithm". In: *ICPM*. IEEE, 2019, pp. 121–128.

[116] Gyunam Park and Minseok Song. "Optimizing Resource Allocation Based on Predictive Process Monitoring". In: *IEEE Access* 11 (2023), pp. 38309–38323.

[117] Ken Peffers et al. "A Design Science Research Methodology for Information Systems Research". In: *J. Manag. Inf. Syst.* 24.3 (2008), pp. 45–77. DOI: 10.2753/mis0742-1222240302.

[118] P. Jonathon Phillips et al. “Four principles of explainable artificial intelligence”. In: (2021).

[119] Azzurra Pini, Ross Brown, and Moe Thandar Wynn. “Process Visualization Techniques for Multi-perspective Process Comparisons”. In: *AP-BPM*. Vol. 219. Lecture Notes in Business Information Processing. Springer, 2015, pp. 183–197.

[120] Jennifer Preece. *Interaction Design: beyond human-computer interaction*. 2002.

[121] Williams Rizzi et al. “Nirdizati 2.0: New Features and Redesigned Backend”. In: *BPM (PhD/Demos)*. Vol. 2420. CEUR Workshop Proceedings. CEUR-WS.org, 2019, pp. 154–158.

[122] Williams Rizzi et al. “Explainable predictive process monitoring: a user evaluation”. In: *Process Science* 1.1 (2024), p. 3.

[123] Maximilian Röglinger et al. “Exogenous Shocks and Business Process Management”. In: *Bus. Inf. Syst. Eng.* 64.5 (2022), pp. 669–687.

[124] Emmelien De Roock. “Bringing healthcare professionals to the forefront in process mining: novel methods to identify information needs and visualize outcomes”. In: *ICPM Doctoral Consortium / Demo*. Vol. 3648. CEUR Workshop Proceedings. CEUR-WS.org, 2023.

[125] Jenny Ruiz, Estefanía Serral, and Monique Snoeck. “Unifying Functional User Interface Design Principles”. In: *Int. J. Hum. Comput. Interact.* 37.1 (2021), pp. 47–67. DOI: 10.1080/10447318.2020.1805876.

[126] Helen Schonenberg et al. “Supporting Flexible Processes through Recommendations Based on History”. In: *BPM*. Vol. 5240. Lecture Notes in Computer Science. Springer, 2008, pp. 51–66.

[127] Anjo Seidel, Stephan Haarmann, and Mathias Weske. “Model-Based Recommendations for Next-Best Actions in Knowledge-Intensive Processes”. In: *CAiSE*. Vol. 14663. Lecture Notes in Computer Science. Springer, 2024, pp. 195–211.

[128] Woosuk Seo, Chanmo Yang, and Young-Ho Kim. “ChaCha: Leveraging Large Language Models to Prompt Children to Share Their Emotions about Personal Events”. In: *CHI*. ACM, 2024, 903:1–903:20.

[129] Ben Shneiderman. *Designing the user interface - strategies for effective human-computer interaction, 3rd Edition*. Addison-Wesley, 1997.

[130] Ben Shneiderman. *Human-centered AI*. Oxford University Press, 2022.

[131] Mahmoud Shoush and Marlon Dumas. “Prescriptive Process Monitoring Under Resource Constraints: A Causal Inference Approach”. In: *ICPM Workshops*. Vol. 433. Lecture Notes in Business Information Processing. Springer, 2021, pp. 180–193.

[132] Mahmoud Shoush and Marlon Dumas. “Intervening With Confidence: Conformal Prescriptive Monitoring of Business Processes”. In:

PMAI@IJCAI. Vol. 3569. CEUR Workshop Proceedings. CEUR-WS.org, 2023.

[133] Mahmoud Shoush and Marlon Dumas. “Prescriptive Process Monitoring Under Resource Constraints: A Reinforcement Learning Approach”. In: *KI-Künstliche Intelligenz* (2024), pp. 1–22.

[134] Mahmoud Shoush and Marlon Dumas. “White box specification of intervention policies for prescriptive process monitoring”. In: *Data & Knowledge Engineering* 155 (2025), p. 102379.

[135] Renuka Sindhgatta, Aditya K. Ghose, and Hoa Khanh Dam. “Context-Aware Analysis of Past Process Executions to Aid Resource Allocation Decisions”. In: *CAiSE*. Vol. 9694. Lecture Notes in Computer Science. Springer, 2016, pp. 575–589.

[136] Kanupriya Singh et al. “A Formative Usability Study to Improve Prescriptive Systems for Bioinformatics Big Data”. In: *BIBM*. IEEE, 2020, pp. 735–742.

[137] Marit Sirgmets et al. “Designing Process Diagrams - A Framework for Making Design Choices When Visualizing Process Mining Outputs”. In: *OTM Conferences (1)*. Vol. 11229. Lecture Notes in Computer Science. Springer, 2018, pp. 463–480.

[138] Elizaveta Sorokina et al. “PEM4PPM: A Cognitive Perspective on the Process of Process Mining”. In: *BPM*. Vol. 14159. Lecture Notes in Computer Science. Springer, 2023, pp. 465–481.

[139] International Organization for Standardization/International Electrotechnical Commission. *ISO/IEC 9241-11 Ergonomic requirements for office work with visual display terminals (VDTs) — Part 11: Guidance on usability,* 1998. URL: `https://www.iso.org/standard/16883.html`.

[140] Thomas Süße, Maria Kobert, and Caroline Kries. “Antecedents of Constructive Human-AI Collaboration: An Exploration of Human Actors’ Key Competencies”. In: *PRO-VE*. Vol. 629. IFIP Advances in Information and Communication Technology. Springer, 2021, pp. 113–124.

[141] Irene Teinemaa et al. “Alarm-Based Prescriptive Process Monitoring”. In: *BPM (Forum)*. Springer, 2018, pp. 91–107.

[142] Irene Teinemaa et al. “Alarm-Based Prescriptive Process Monitoring”. In: *BPM (Forum)*. Vol. 329. Lecture Notes in Business Information Processing. Springer, 2018, pp. 91–107. DOI: `10.1007/978-3-319-98651-7\_6`.

[143] Ilya Verenich et al. “Predictive Process Monitoring in Apromore”. In: *CAiSE Forum*. Vol. 317. Lecture Notes in Business Information Processing. Springer, 2018, pp. 244–253.

[144] Daniel Viner, Matthias Stierle, and Martin Matzner. “A Process Mining Software Comparison”. In: *ICPM Doctoral Consortium / Tools*. Vol. 2703. CEUR Workshop Proceedings. CEUR-WS.org, 2020, pp. 19–22.

[145] Jan Vom Brocke, Alan Hevner, and Alexander Maedche. “Introduction to design science research”. In: *Design science research. Cases* (2020), pp. 1–13.

[146] Karel Vredenburg et al. “A survey of user-centered design practice”. In: *CHI*. ACM, 2002, pp. 471–478.

[147] Colin Ware. *Information visualization: perception for design*. Morgan Kaufmann, 2019.

[148] Jing Wei et al. “Leveraging Large Language Models to Power Chatbots for Collecting User Self-Reported Data”. In: *Proc. ACM Hum. Comput. Interact.* 8.CSCW1 (2024), pp. 1–35.

[149] Sven Weinzierl et al. “From predictive to prescriptive process monitoring: Recommending the next best actions instead of calculating the next most likely events”. In: *Wirtschaftsinformatik (Zentrale Tracks)*. GITO Verlag, 2020, pp. 364–368.

[150] Sven Weinzierl et al. “Prescriptive Business Process Monitoring for Recommending Next Best Actions”. In: *BPM (Forum)*. Springer, 2020, pp. 193–209.

[151] Hans Weytjens, Wouter Verbeke, and Jochen De Weerdt. “Timed Process Interventions: Causal Inference vs. Reinforcement Learning”. In: *Business Process Management Workshops*. Vol. 492. Lecture Notes in Business Information Processing. Springer, 2023, pp. 245–258.

[152] Arif Wibisono et al. “On-the-Fly Performance-Aware Human Resource Allocation in the Business Process Management Systems Environment Using Naive Bayes”. In: *AP-BPM*. Springer, 2015, pp. 70–80.

[153] Bemali Wickramanayake et al. “Generating Purpose-Driven Explanations: The Case of Process Predictive Model Inspection”. In: *CAiSE Forum*. Vol. 452. Lecture Notes in Business Information Processing. Springer, 2022, pp. 120–129.

[154] Moe Thandar Wynn et al. “ProcessProfiler3D: A visualisation framework for log-based process performance comparison”. In: *Decis. Support Syst.* 100 (2017), pp. 93–108. DOI: 10.1016/j.dss.2017.04.004.

[155] Sen Yang et al. “A Data-driven Process Recommender Framework”. In: *KDD*. ACM, 2017, pp. 2111–2120.

[156] L Richard Ye and Paul E Johnson. “The impact of explanation facilities on user acceptance of expert systems advice”. In: *Mis Quarterly* (1995), pp. 157–172.

[157] Sergey Zeltyn et al. “Prescriptive Process Monitoring in Intelligent Process Automation with Chatbot Orchestration”. In: *PMAI@IJCAI*. Vol. 3310. CEUR Workshop Proceedings. CEUR-WS.org, 2022, pp. 49–60.

[158] Jeffrey C Zemla et al. “Evaluating everyday explanations”. In: *Psychonomic bulletin & review* 24 (2017), pp. 1488–1500.

[159] Francesca Zerbato, Pnina Soffer, and Barbara Weber. “Process Mining Practices: Evidence from Interviews”. In: *BPM*. Vol. 13420. Lecture Notes in Computer Science. Springer, 2022, pp. 268–285. DOI: 10.1007/978-3-031-16103-2_19.

[160] Francesca Zerbato et al. “ProMiSE: Process Mining Support for End-Users”. In: *CAiSE Research Projects Exhibition*. Vol. 3413. CEUR Workshop Proceedings. CEUR-WS.org, 2023, pp. 60–67.

[161] Ruijing Zhao, Izak Benbasat, and Hasan Cavusoglu. “Transparency in Advice-Giving Systems: A Framework and a Research Model for Transparency Provision”. In: *IUI Workshops*. Vol. 2327. CEUR Workshop Proceedings. CEUR-WS.org, 2019.

Appendix A. FRAMEWORK FOR PRPM OUTPUTS

Figure 22 is an enlarged version of the Figure 6, and Figure 23 is en enlarged version of the Figure 7 (Section 3.2.7).

Objective	Target	Rec. perspective	Recommendation	Input perspective	Policy	Rec. frequency	Prescriptive analytics method	Rec. purpose	Detail. algorithm	User eval./ UI	Reference	Number ref.
Optimizing process outcome	Temporal outcome	Resource	Resource settings	All available	Exceeding metric limits	Continuous	ML/DM, Logic-Based models	Optimizing (cor.)	Y		Gröger et al. (2014)	[45]
			Resource allocation	R; T; D	Highest predicted metric value	Continuous	ML/DM	Optimizing (cor.)	Y		Sindhgatta et al. (2016)	[135]
		Control flow	Next activity	T; C	Multiple options (time-based, deadline-based, decision-based, goal-based)	Continuous	ML/DM	Optimizing (cor.)	N	*	Huber et al. (2015)	[60]
				C	Exceeding metric limits, maximum metric improvement	Discrete	ML/DM	Optimizing (cor.)	Y		Weinzierl et al. (2020b)	[150]
				C; R	Predicted outcome violation	Continuous	ML/DM	Optimizing (cor.)	Y		Zeltyn et al. (2022)	[157]
				Metric specific	Maximum metric improvement	Continuous	ML/DM, Simulation	Optimizing (cor.)	Y		de Leoni et al. (2020)	[84]
	Categorical outcome	Control flow	Specific activity to perform	T; D; C	Resource availability, urgency of the case, necessity of intervention, and uncertainty in predictions	Discrete	ML/DM, Causal Inference	Optimizing (caus.)	Y		Shoush and Dumas (2024)	[133]
				T; D; C	Decision rules based on outcome prediction, intervention effectiveness, urgency, and resource availability	Discrete	ML/DM, Causal Inference	Optimizing (caus.)	Y		Shoush and Dumas (2025)	[134]
				T; D; C	Optimal policy learned by the agent	Discrete	ML/DM, Causal Inference	Optimizing (caus.)	Y		Bozorgi et al. (2023a)	[14]
		Varies (multiple)	An alarm to trigger an intervention	All available	Probability of a negative outcome above a threshold, cost model, mitigation effectiveness	Discrete	ML/DM	Optimizing (cor.)	Y		Fahrenkrog-Petersen et al. (2022)	[37]
				All available	Probability of a negative outcome above a threshold, cost model, reliability estimate	Discrete	ML/DM	Optimizing (cor.)	Y		Metzger et al. (2020)	[95]
			Various	R; T; D	Optimal value of target metric	Discrete	ML/DM	Optimizing (cor.)	Y	*	Agarwal et al. (2022)	[5]
				R; C; T	Values outside the acceptable range	Continuous	ML/DM, Evolutionary Computation	Optimizing (cor.)	Y		Padella and de Leoni (2023)	[111]
				All available	Probability of a negative outcome above a threshold, intervention cost, resource availability	Discrete	ML/DM, Causal Inference	Optimizing (caus.)	Y		Shoush and Dumas (2021)	[131]
			Various process decisions	D; C	Highest predicted metric value	Continuous	ML/DM, Logic-Based models	Optimizing (cor.)	Y		Ghattas et al. (2014)	[44]
			Temporal relations among activities that have to be preserved or violated	T	Temporal logic-based prioritization of recommendations	Discrete	ML/DM, Logic-Based models	Optimizing (cor.)	Y		Donadello et al. (2023)	[31]
Both	Any	Control flow	Specific activity to perform	T; D; C	User-selected policy based on the provided cost-benefit analysis	Discrete	ML/DM, Causal Inference	Optimizing (caus.)	Y		Bozorgi et al. (2023b)	[15]
			Next activity	C; D	Path scoring based on goal fulfillment, cost considerations, and knowledge worker-defined input configuration	Continuous	ML/DM, Logic-Based models	Optimizing (cor.)	Y	*	Seidel et al. (2024)	[127]

Figure 22. [Enlarged] Framework for PrPM outputs (Optimizing process outcome objective).

Objective	Target	Rec. perspective	Recommendation	Input perspective	Policy	Rec. frequency	Prescriptive analytics method	Rec. purpose	Detail. algorithm	User eval./ UI	Reference	Number ref.
Optimizing process efficiency	Cycle time	Resource	Resource for next activity	C; T	Highest predicted resource performance	Continuous	Probabilistic Models	Guiding	Y		Wibisono et al. (2015)	[152]
			Resource allocation	C; R	Minimal cost, maximum flow	Continuous	ML/DM	Optimizing (cor.)	Y		Park and Song (2023)	[116]
	Processing time			R; C; T	Scheduling algorithm	Continuous	ML/DM	Optimizing (cor.)	Y		Park and Song (2019)	[115]
			Resource for next activity	R; C	Highest predicted resource performance	Continuous	ML/DM, Logic-Based models	Optimizing (cor.)	Y	*	Kim et al. (2013)	[72]
				T; D; R	Task duration predictions based on resource experience and workload distribution considerations, minimizing the Time-Workload coefficient	Continuous	ML/DM, Logic-Based models	Optimizing (cor.)	Y		Padella et al. (2024)	[112]
	Cycle time	Control flow	Next activity	C	Maximum metric improvement	Continuous	Logic-Based Models	Guiding	N		Nakatumba et al. (2012)	[101]
	Defect rate		Set of activities	D; C	Similarity based	Continuous	Statistical Analysis	Guiding	Y	*	Yang et al. (2017)	[144]
			Next activity	C; R	Risk prediction based on historical data, scheduling algorithm	Discrete	ML/DM, Mathematical Programming	Optimizing (cor.)	Y		Conforti et al. (2015)	[27]
	Cost		Specific activity to perform	C; T; D	Risk prediction based on historical data	Discrete	ML/DM	Optimizing (cor.)	Y	*	Dees et al. (2019)	[30]
	Varies (multiple)		Next activity	C	Set of rules	Discrete	Logic-Based Models	Guiding	Y		Haisjackl et al. (2010)	[48]
				C	Optimal policy computed by using Monte Carlo methods	Continuous	ML/DM	Optimizing (cor.)	Y		Branchi et al. (2022)	[16]
			Best next path	D; C	Maximum metric improvement	Discrete	ML/DM, Probabilistic models	Optimizing (cor.)	Y		Khan et al. (2021)	[71]
		Varies (multiple)	Resouce allocation, additional resources, adjusting case attributes	C; R	Temporal patterns of operational constraints	Discrete	Logic-Based Models	Guiding	Y		Park et al. (2023)	[114]
			Multi-activity recommendations (process schedule)	C; R; T; D	External power price forecasts, real-time data on power consumption, and process-specific constraints	Discrete	Mathematical Programming	Guiding	Y	*	Hermann et al. (2024)	[52]
			Next activity and resource	C; R	Set of rules (optimized plans, resource availability)	Continuous	Mathematical Programming	Guiding	N		Barba et al. (2012)	[8]
	Cycle time	Varies (multiple)	Next activity and resource	R; C; D	Similarity based	Continuous	Probabilistic Models	Guiding	N		Nezhad et al. (2011)	[104]

Figure 23. [Enlarged] Framework for PrPM outputs (Optimizing process efficiency objective, and both).

Appendix B. PRPM WIREFRAME AND INTERFACE

B.1. Wireframe

Figure 24 is an enlarged version of the Figure 10 (Section 4.2.2). It depicts the information items from groups #1 and #2 (Figure 9).

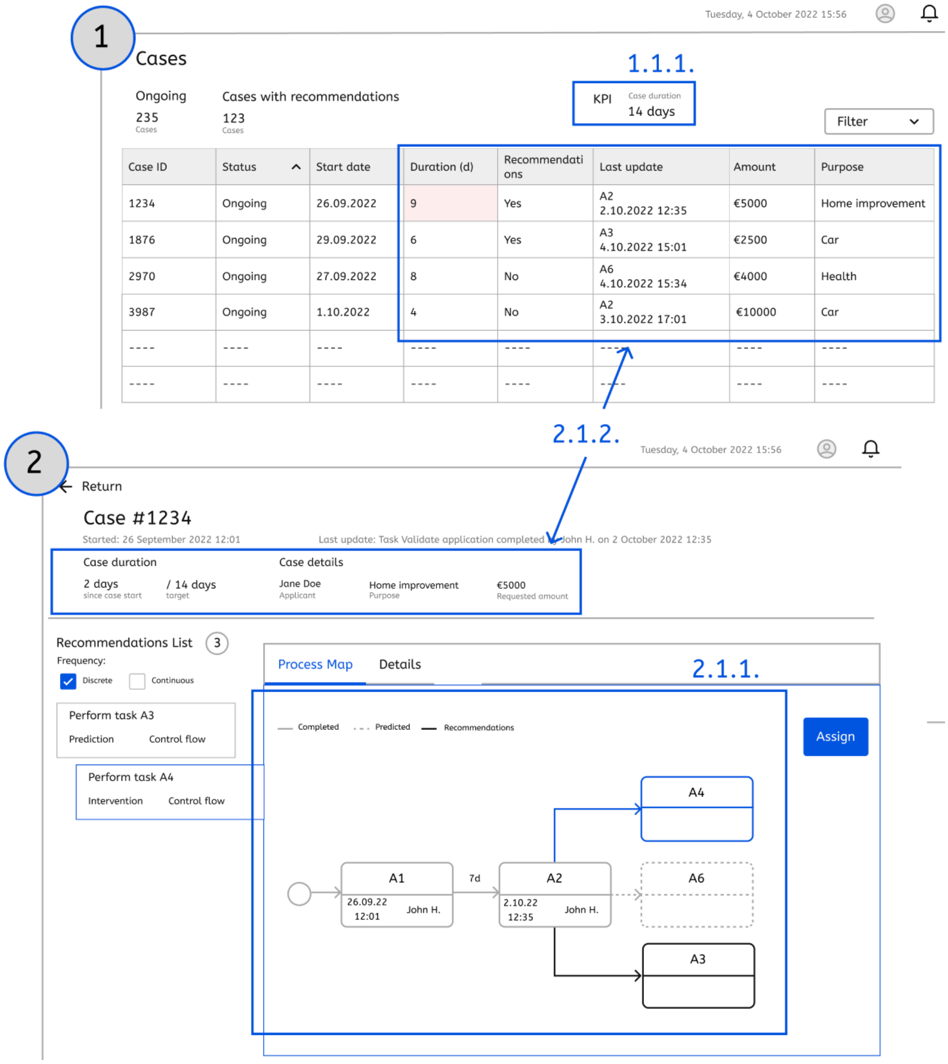

Figure 24. [Enlarged] Wireframe: view #1 Cases Overview, and view #2 Case Description.

Figure 25 is an enlarged version of the Figure 10 (Section 4.2.2). It depicts the information items from groups #3 and #4 (Figure 9).

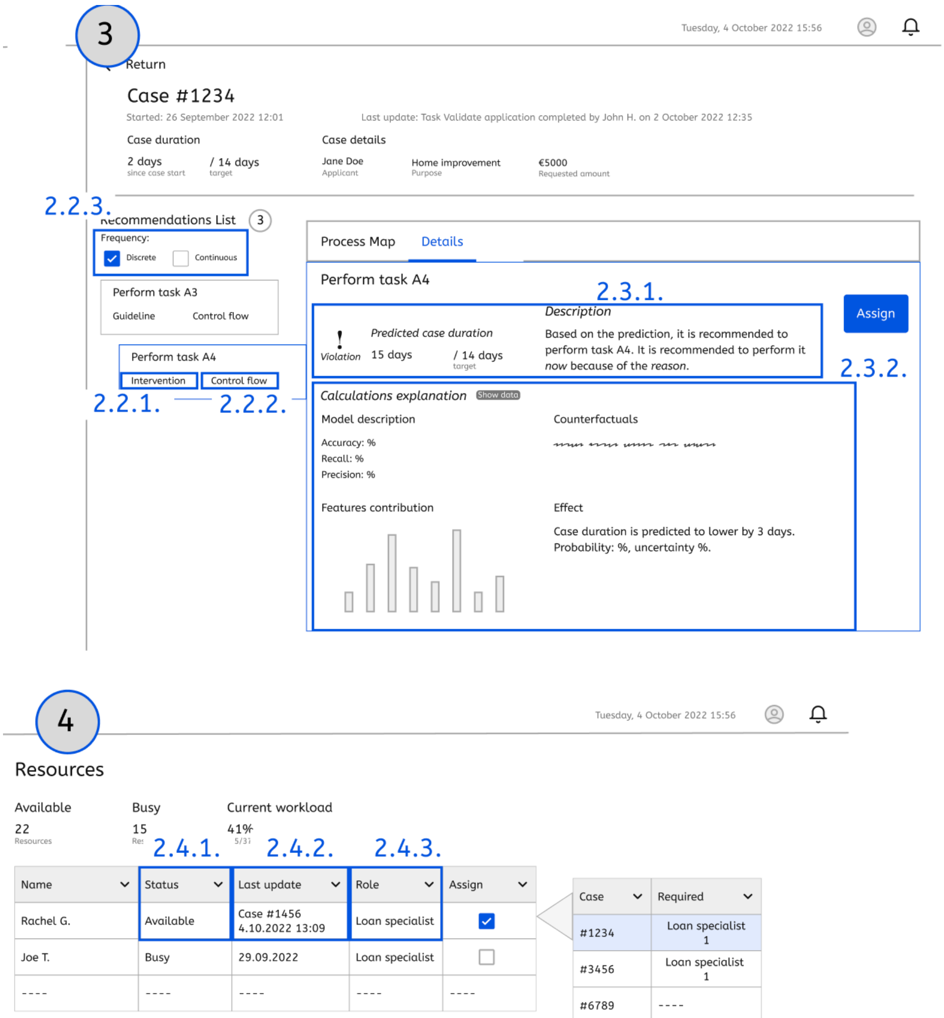

Figure 25. [Enlarged] Wireframe: view #3 Recommendation Description, and view #4 Assign Resource.

B.2. Kairos for Process Analysts

Figure 26 is an enlarged version of the Figure 16 (Section 5.2.2). It depicts the functionality for process analysts based on the identified information items.

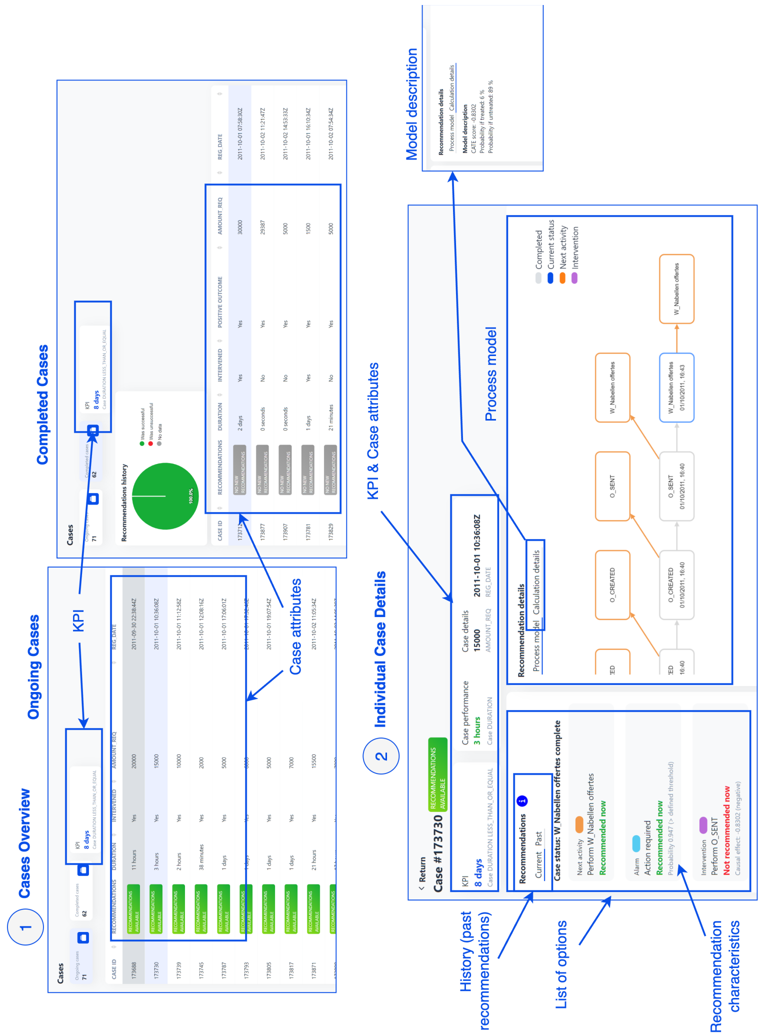

Figure 26. [Enlarged] Kairos functionality for process analysts.

B.3. Kairos for Operational Managers

Figure 27 is an enlarged version of the Figure 17, top half (Section 5.2.2). It depicts the functionality for operational managers (view #1 Cases Overview) based on the identified information items.

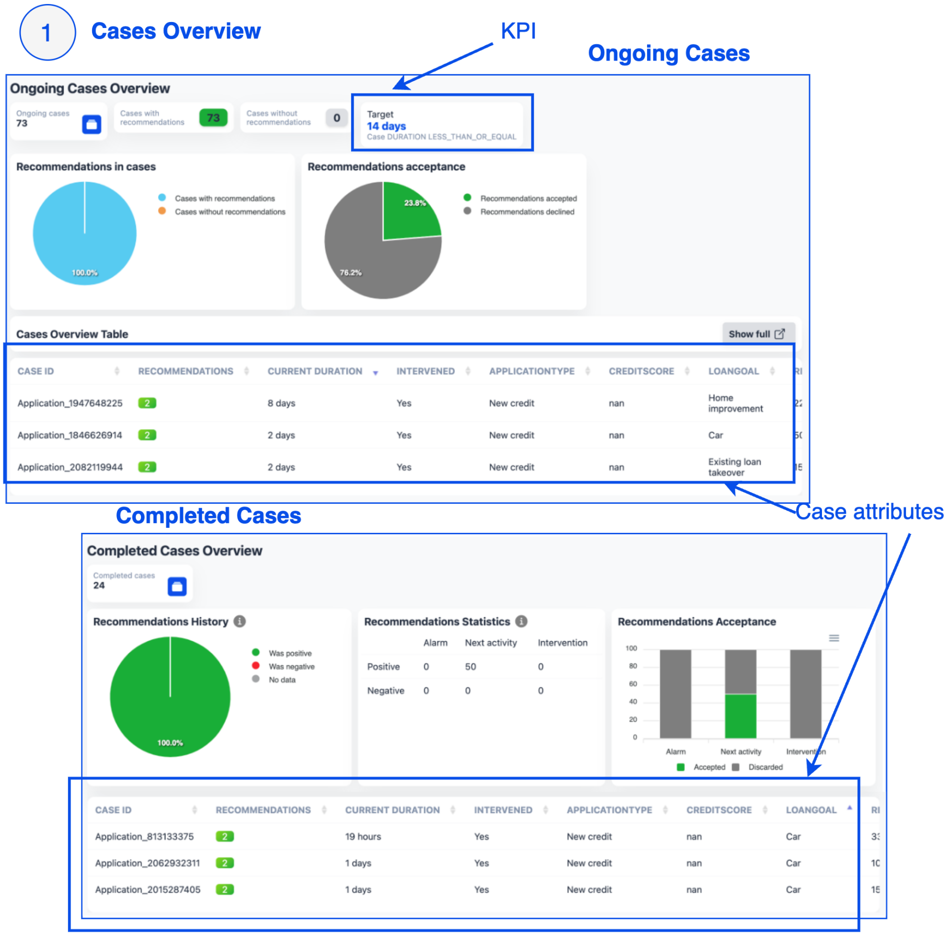

Figure 27. [Enlarged] Kairos functionality for operational managers: view #1 Cases Overview (ongoing and completed cases).

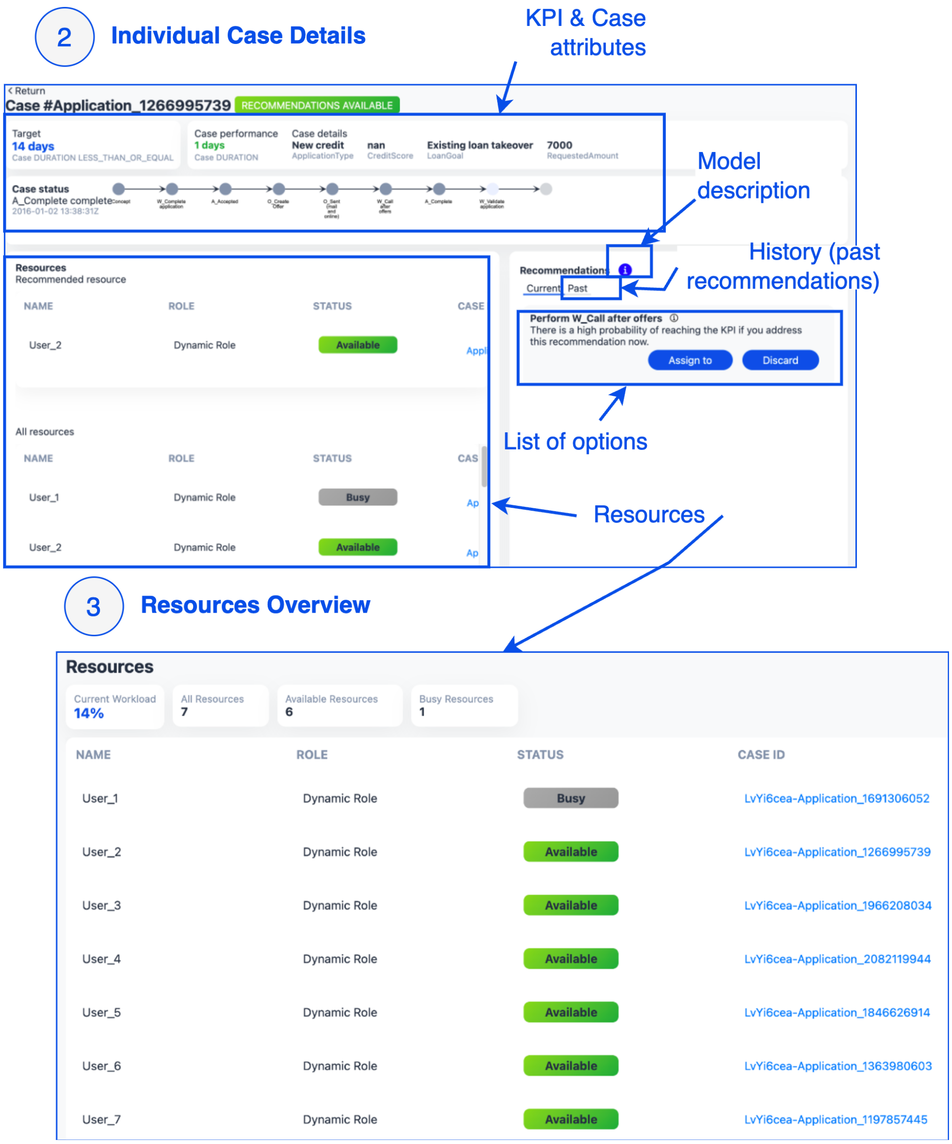

Figure 28. [Enlarged] Kairos functionality for operational managers: view #2 Individual Case Details, and view #3 Resources Overview.

Figure 28 is an enlarged version of the Figure 17, bottom half (Section 5.2.2). It depicts the functionality for operational managers (view #2 Individual Case Details and view #3 Resourcs Overview) based on the identified information items.

Appendix C. EVALUATION STUDIES DATA COLLECTION INSTRUMENTS

Table 19. Development Cycle I: interview guide (Section 4.1.1).

1. Think about a (similar) recent situation that you remember well where this kind of tool could be used (by your process workers). Please describe this situation to me.
2. (For each wireframe view) Based on this situation, which content elements do you find:
   - Most relevant and why? Least relevant and why?
   - Most useful and why? Not useful and why?
   - What could be improved about the content?
3. (At the end) What information do you think is missing?

Table 20. Development Cycle II: interview guide (process analysts) (Section 5.1.3).

1. Think about your last project. Describe it a bit: what was it about? Would this tool have been helpful in the project? What tasks could you imagine using it for?
2. Which information did you find most useful and why?
3. Which information did you find least useful and why?
4. Which aspects of the interface did you have trouble with?
5. What worked in the way you expected it to and what did not?
6. What information is missing in the interface?
7. What could be improved about the tool?

Table 21. Development Cycle II: interview guide (operational managers) (Section 5.1.3).

1. Think about your last project. Describe it a bit: what was it about? Would this tool have been helpful in the project? What tasks could you imagine using it for?
2. Which information did you find most useful and why?
3. Which information did you find least useful and why?
4. How would you sort the table of ongoing loan applications (cases) to prioritize them?
5. How would you assign the recommendations to your team members?
6. What information is missing in the interface?
7. What could be improved about the tool?

Table 22. Development Cycle III: survey questions (Section 6.2.3).

The explanations...	**Strongly disagree**	**Somewhat disagree**	**Neither agree nor disagree**	**Somewhat agree**	**Strongly agree**
...help me understand how Kairos works.	○	○	○	○	○
...of how Kairos works are satisfying.	○	○	○	○	○
...have sufficient detail.	○	○	○	○	○
...seem complete.	○	○	○	○	○
...tell me how to use Kairos.	○	○	○	○	○
...are useful to my goals.	○	○	○	○	○
...show me how accurate Kairos is.	○	○	○	○	○
...let me judge when I should trust and not trust Kairos.	○	○	○	○	○

ACKNOWLEDGMENTS

I imagined the moment of writing these acknowledgments so many times. In my head, I've used all kinds of poetic opening words. Yet, when I'm finally putting them on paper, I don't know where to start. So I'll start at the beginning.

I met Fredrik Milani in the first days of my master's studies. I so clearly remember him standing at the lectern in one of the classrooms of the old faculty building, telling our newly arrived group of students about a program to take part in. Watching him talk about such a mundane topic, and yet with such charisma, made me think, "Wow, I wanna be like him when I grow up." Fredrik, there are so many things I've learned from you. Thank you for inviting me to be a teaching assistant for your course so early on and for taking me in for a master's thesis. I cannot even begin to describe how big that made me feel back then. Eventually, it was you who convinced me to do a PhD; you do always know how to talk. You opened many doors for me, and for that, I am forever grateful. Alexander Nolte joined as a co-supervisor of my master's thesis. Alex, I'm glad you also remained for PhD. A wise man once said to me that a PhD is a marathon and not a race. I think (I know) it was you, and not once, but repeatedly. Thank you for that. Thank you for reminding me to celebrate the small victories, and, most importantly, thank you for all the cat content. Marlon Dumas completed the trio of my supervisors. Marlon, being a part of your research group was an honor and I will always remember it as such. Thank you for all the opportunities you have given me and all the people you introduced me to. Your strategic direction and ambition always made me strive for more. I would be lying if I said that having three supervisors was not challenging sometimes. It was. But it was an experience like no other. Thank you all for giving me the flexibility I needed but also for providing the guidance for my ship to not get lost at sea. My PhD sailing was stormy at times, but I made it thanks to you.

I thank my internal reviewer, Radwa El-Shawi, and external reviewers, Adela del Río Ortega and Pnina Soffer, for taking the time to read this thesis and provide valuable comments. Your suggestions truly have made it better. I must also say how glad I was to have such brilliant women as reviewers. As a bonus, it restored the cosmic balance after a supervisor team of three (equally brilliant) men.

Many thanks to my co-authors, Lana Botchorishvili, Mahmoud Shoush, and Zhaosi Qu, for their contributions. I'm also grateful to Viktoriia Abakumova for quickly jumping on board towards the end. Thanks to Anti Alman and Kristo Raun for translating the texts to Estonian (but for the great companionship at conferences, too).

Many chapters of this thesis relied on the valuable insights of the many interviewees I spoke with throughout. Thank you all, you've helped me tremendously.

I kept realizing that when doing a PhD, it's very important to believe in yourself. But if you are not particularly good at that, it's important to have people around you who do the believing job for you. I'm so fortunate to have my friends.

Sasha and Tanya F., whenever I wanted to quit and become a baker, you told me you'd support me regardless of who I am or want to be. Дякую вам за це. Tanya Sh., thank you for being there since day 1. You listened to my whining during our bachelor's, master's, and now my PhD – дякую. Thank you for introducing me to *Friends* years ago; this show has comforted me so many times that I probably don't have such a number in my calculator. Mariia, we're both doctors now, can you believe it? Sharing this road with you made it much easier. Just so you know, the word "PhD" in both alphabets comes up 389 times in our chat. Дякую тобі. Thanks, Łukasz, for our monthly calls. Thanks, Katrin, for a listening ear and the most random videos. But, really, anybody who was (un)fortunate enough to hear me complain – thanks. It was part of the process.

I owe so much to my parents, Valentyna and Anatolii. They are the ones who taught me the importance of knowledge early on. Дякую, мам, пап (і Льовочка теж). Без вас нічого цього б не було. I dedicate this thesis to my grandfather Hryhorii. He had no formal education, and yet was the brightest person. He diligently solved crossword puzzles every single day. I will never forget the quiet pride in his eyes when I told him about my little achievements. Діду, Ви наснились мені на самому початку мого PhD і сказали: «Все тільки починається». Я дуже часто про Вас думаю. Дякую, що оберігаєте мене.

And finally, Jan. Thank you for listening to my presentations, advising on slide design, cooking all meals, and keeping yourself busy during the weekends and long evenings I spent with this thesis. You really are the most patient person in the world. I don't know how to find the words, so I'll borrow them from *The Grand Budapest Hotel (2014)*: "With respect, adoration, admiration, kisses, gratitude, best wishes, and love from ~~Z to A~~ K to J." Danke, Beeb! You are my rock.

* * *

What a time. There was so much turmoil, but just as many moments of little joy; so much predictability, yet so many surprises; so many things that seemed significant were actually not, and so many that seemed insignificant turned out to be just the opposite. This experience may have equipped me with research skills, but the life lessons I take from it are much more valuable. I will never forget all the disbelief I was planting in myself over the years. But I will also remember the hot and so beautiful streets of Barcelona, my first conference. I will definitely remember the soaking thunderstorm of Rome. I will cherish the good time shared with my academic friends in scorching hot Zaragoza. And I will never ever forget the peaceful falling leaves in Kraków. These sound like nothing more than a travel diary. But those are just the associations I make with the deep memories created in each of those places. And that is what I take with me as I go forward.

> *Зайди в холодну воду/ не бійся намочити ноги/ переплисти кілометри льоду/ з дельфінами на той бік моря.*
> [Скрябін – Дельфіни]

І наостанок – Слава Україні!

SISUKOKKUVÕTE

Kasutajakesksete ettekirjutavate protsessijälgimise süsteemide suunas

Organisatsioonid püüavad pidevalt parendada oma äriprotsesse, et tõsta tõhusust ja olla konkurentidest edukamad. Äriprotsess on kogum tegevustest, mis konkreetses järjekorras sooritatuna annab tulemuseks teenuse või toote. Üks võimalus äriprotsesside parendamiseks on kasutada ettekirjutava protsessijälgimise (ingl *prescriptive process monitoring, PrPM*) süsteeme.

PrPM-süsteem on infosüsteem, mis soovitab äriprotsessi täitmisel jooksvalt täiendavaid tegevusi, eesmärgiga parandada protsessi tulemuslikkust. Olemasolevad lähenemised PrPM-süsteemide arendamiseks erinevad nii optimeerimiseesmärkide, loodud soovituste kui ka algoritmide poolest, mis määravad milliseid soovitusi anda ja millal.

Enamus PrPM-alasest teadustööst keskendub soovituste koostamiseks kasutatavate algoritmide tõhususele ja täpsusele. Samas, nende süsteemide kasutatavusele on seni vähe tähelepanu pööratud, mistõttu on PrPM-süsteemid leidnud seni vaid piiratud kujul praktilist rakendust. Käesoleva doktoritöö eesmärk on selle puuduse käsitlemine, uurides, kuidas kasutajad suhtlevad PrPM-süsteemide poolt pakutavate soovitustega.

Doktoritöö raames on püstitatud neli uurimisküsimust: (1) Millised on PrPM-tehnikate väljundid? (2) Millised kasutajarühmad võiksid PrPM-tehnikate väljunditega töötamisest kasu saada ja mis on nende teabevajadused? (3) Kuidas kajastada neid teabevajadusi PrPM-süsteemide kasutajaliidese disainis? (4) Kuidas suurendada PrPM-väljundite arusaadavust?

Nendele küsimustele vastamiseks viiakse esmalt läbi süstemaatiline kirjanduse ülevaade, mis käsitleb põhjalikult olemasolevaid PrPM-tehnikaid ja nende väljundeid ning millele tuginedes luuakse kontseptuaalne raamistik. See raamistik liigitab PrPM-tehnikad eesmärkide, soovitus-tüüpide, andmesisendite, järgitavate reeglite ja kasutajatestide alusel. Need liigitused aitavad mõista lõppkasutajatele esitatavate PrPM-väljundite valikut. Tuvastame ka senise teadustöö peamised lüngad ja ettekirjutava protsessijälgimise tulevikusuunad. Eelkõige tõuseb esile olemasolevate tehnikate puudulik valideerimine reaalsetes oludes, vähene rõhuasetus kasutajakesksusele tagamaks tehnikate kasulikkust ja kasutatavust ning puudulik seletatavus ja tagasiside PrPM-süsteemi ja selle lõppkasutajate vahel.

Teiseks käsitleb doktoritöö küsimust, milline peaks olema PrPM-süsteemi kasutajaliides. Sellele küsimusele vastamiseks kaardistame kolme erineva kasutajarühma (protsessianalüütikud, töötajad ja operatiivjuhid) teabevajadused tuginedes PrPM-tehnikate väljundite analüüsile ja potentsiaalsete kasutajatega tehtud intervjuudele. Teabevajaduste all viitame konkreetsetele teabeelementidele, mis on erinevatele lõppkasutajatele kasutajaliideses vajalikud (nt ressursid on olulised operatiivjuhtidele, kuid mitte töötajatele). Eelnimetatud kaardistusele tuginedes loome ja analüüsime PrPM tööriista nimega Kairos. Kairos on veebipõhine tööriist, mis annab äriprotsesside sündmuste logide põhjal soovitusi täiendavate tegevuste tegemiseks protsessi käimasolevate juhtumite täitmisel. Kairose kasutajatestimine viidi läbi lõppkasutajatega (protsessianalüütikud ja operatiivjuhid) kes leidsid, et tegemist on kasutatava ja kasuliku tööriistaga, juhtides samas tähelepanu ka mitmetele parendusvõimalustele. Kasutajatestimise tulemuste põhjal täpsustame eelnevalt kaardistatud teabeelemente ning pakume soovitused PrPM-süsteemide kasutajaliideste loomiseks. Soovituste põhjal oleks vaja isikupärastatud kasutajaliideseid (vastavalt teabevajadusele), selgitusi, võimalust interaktiivseks andmete uurimiseks ning otsustustuge.

Kolmandaks, et abistada kasutajaid PrPM-tehnikate väljundite mõistmisel, pakub käesolev doktoritöö välja suurte keelemudelite viipamise meetodi parendamaks PrPM-väljundite seletatavust. Siinkohal eristame tehnika toimimise selgitusi ("miks"konkreetne väljund antakse) tehnika väljundi selgitustest ("midaäntud väljund tähendab). Nimetatud meetodi loomisel kirjeldame esmalt kasutajate vajadused PrPM seletatavuse osas ning seejärel töötame vajadustest lähtuvalt välja vastava viipamise meetodi, mis sisaldab konteksti, andmete kirjeldust, üldisi vestlusreegleid, näiteid ja ülesande kirjeldust. See meetod on rakendatud eelnevalt nimetatud PrPM tööriistas Kairos interaktiivse vestlusliidesena ning hinnatud kasutajatestides lõppkasutajatega. Lisaks analüüsime kasutajatestide käigus kasutajate poolt küsitud küsimusi, meetodi poolt antud vastuseid ja vestlusliidese kasutust. Analüüsi põhjal anname ülevaate võimalikest eelistest ja väljakutsetest suurte keelemudelite kasutamisel PrPM-väljundite seletatavuse parandamiseks.

Kokkuvõttes aitab see doktoritöö kaasa kasutajakesksete PrPM-süsteemide kujundamisele, kus soovitused pole mitte ainult tehniliselt vastupidavad, vaid ka lõppkasutajatele kasutatavad ja rakendatavad. Selle doktoritöö tulemused edendavad nii PrPM-süsteemide teoreetilist arusaama kui ka praktilist rakendamist.

CURRICULUM VITAE

Personal data

Name:	Kateryna Kubrak
Date of Birth:	20.02.1998
Citizenship:	Ukraine
Contact:	kateryna.kubrak@gmail.com

Education

2021–2025	PhD in Computer Science – University of Tartu, Tartu, Estonia
2019–2021	Master of Arts in Innovation and Technology Management *(cum laude)* – University of Tartu, Tartu, Estonia
2015–2019	Bachelor in Computer Science – Taras Shevchenko National University of Kyiv, Kyiv, Ukraine

Employment

2024–...	Process Manager – Nordea, Tallinn, Estonia
2021–2025	Junior Research Fellow – Institute of Computer Science, University of Tartu, Tartu, Estonia
2020–2021	Project Coordinator – Institute of Computer Science, University of Tartu, Tartu, Estonia
2018	Research Intern – University of Ottawa, Ottawa, Canada

Teaching

August 2023	Digital Product Management Course at UniTartu Summer University
Fall 2020–Spring 2024	Software Product Management (Teaching Assistant)

Scientific work

Main fields of interest:

- Business Process Management
- Process Mining
- Human-Centered Research

ELULOOKIRJELDUS

Isikuandmed

Nimi:	Kateryna Kubrak
Sünniaeg:	20.02.1998
Kodakondsus:	Ukraina
Kontakt:	kateryna.kubrak@gmail.com

Haridus

2021–2025	Doktoriõpe informaatika erialal – Tartu Ülikool, Tartu, Eesti
2019–2021	Magistriõpe innovatsiooni ja tehnoloogia juhtimise erialal (*cum laude*) – Tartu Ülikool, Tartu, Eesti
2015–2019	Bakalaureuseõpe informaatika erialal – Kiievi Tarass Šev-tšenko nimeline Rahvusülikool, Kiiev, Ukraina

Teenistuskäik

2024–...	Protsessijuht – Nordea, Tallinn, Eesti
2021–2025	Nooremteadur – Arvutiteaduse instituut, Tartu Ülikool, Tartu, Eesti
2020–2021	Projektide koordinaator – Arvutiteaduse instituut, Tartu Ülikool, Tartu, Eesti
2018	Teadusassistent – Ottawa Ülikool, Ottawa, Kanada

Õppetöö

August 2023	Digitoodete juhtimise kursus UniTartu suveülikoolis
Sügis 2020–Kevad 2024	Tarkvaratoodete juhtimine (õppeassistent)

Teadustegevus

Peamised uurimisvaldkonnad:

- Äriprotsesside juhtimine
- Protsessikaeve
- Inimkesksed uuringud

DISSERTATIONES INFORMATICAE PREVIOUSLY PUBLISHED IN DISSERTATIONES MATHEMATICAE UNIVERSITATIS TARTUENSIS

19. **Helger Lipmaa.** Secure and efficient time-stamping systems. Tartu, 1999, 56 p.
22. **Kaili Müürisep.** Eesti keele arvutigrammatika: süntaks. Tartu, 2000, 107 lk.
23. **Varmo Vene.** Categorical programming with inductive and coinductive types. Tartu, 2000, 116 p.
24. **Olga Sokratova.** Ω-rings, their flat and projective acts with some applications. Tartu, 2000, 120 p.
27. **Tiina Puolakainen.** Eesti keele arvutigrammatika: morfoloogiline ühestamine. Tartu, 2001, 138 lk.
29. **Jan Villemson.** Size-efficient interval time stamps. Tartu, 2002, 82 p.
45. **Kristo Heero.** Path planning and learning strategies for mobile robots in dynamic partially unknown environments. Tartu 2006, 123 p.
49. **Härmel Nestra.** Iteratively defined transfinite trace semantics and program slicing with respect to them. Tartu 2006, 116 p.
53. **Marina Issakova.** Solving of linear equations, linear inequalities and systems of linear equations in interactive learning environment. Tartu 2007, 170 p.
55. **Kaarel Kaljurand.** Attempto controlled English as a Semantic Web language. Tartu 2007, 162 p.
56. **Mart Anton.** Mechanical modeling of IPMC actuators at large deformations. Tartu 2008, 123 p.
59. **Reimo Palm.** Numerical Comparison of Regularization Algorithms for Solving Ill-Posed Problems. Tartu 2010, 105 p.
61. **Jüri Reimand.** Functional analysis of gene lists, networks and regulatory systems. Tartu 2010, 153 p.
62. **Ahti Peder.** Superpositional Graphs and Finding the Description of Structure by Counting Method. Tartu 2010, 87 p.
64. **Vesal Vojdani.** Static Data Race Analysis of Heap-Manipulating C Programs. Tartu 2010, 137 p.
66. **Mark Fišel.** Optimizing Statistical Machine Translation via Input Modification. Tartu 2011, 104 p.
67. **Margus Niitsoo**. Black-box Oracle Separation Techniques with Applications in Time-stamping. Tartu 2011, 174 p.
71. **Siim Karus.** Maintainability of XML Transformations. Tartu 2011, 142 p.
72. **Margus Treumuth.** A Framework for Asynchronous Dialogue Systems: Concepts, Issues and Design Aspects. Tartu 2011, 95 p.
73. **Dmitri Lepp.** Solving simplification problems in the domain of exponents, monomials and polynomials in interactive learning environment T-algebra. Tartu 2011, 202 p.

74. **Meelis Kull.** Statistical enrichment analysis in algorithms for studying gene regulation. Tartu 2011, 151 p.
77. **Bingsheng Zhang.** Efficient cryptographic protocols for secure and private remote databases. Tartu 2011, 206 p.
78. **Reina Uba.** Merging business process models. Tartu 2011, 166 p.
79. **Uuno Puus.** Structural performance as a success factor in software development projects – Estonian experience. Tartu 2012, 106 p.
81. **Georg Singer.** Web search engines and complex information needs. Tartu 2012, 218 p.
83. **Dan Bogdanov.** Sharemind: programmable secure computations with practical applications. Tartu 2013, 191 p.
84. **Jevgeni Kabanov.** Towards a more productive Java EE ecosystem. Tartu 2013, 151 p.
87. **Margus Freudenthal.** Simpl: A toolkit for Domain-Specific Language development in enterprise information systems. Tartu, 2013, 151 p.
90. **Raivo Kolde.** Methods for re-using public gene expression data. Tartu, 2014, 121 p.
91. **Vladimir Šor.** Statistical Approach for Memory Leak Detection in Java Applications. Tartu, 2014, 155 p.
92. **Naved Ahmed.** Deriving Security Requirements from Business Process Models. Tartu, 2014, 171 p.
94. **Liina Kamm.** Privacy-preserving statistical analysis using secure multi-party computation. Tartu, 2015, 201 p.
100. **Abel Armas Cervantes.** Diagnosing Behavioral Differences between Business Process Models. Tartu, 2015, 193 p.
101. **Fredrik Milani.** On Sub-Processes, Process Variation and their Interplay: An Integrated Divide-and-Conquer Method for Modeling Business Processes with Variation. Tartu, 2015, 164 p.
102. **Huber Raul Flores Macario.** Service-Oriented and Evidence-aware Mobile Cloud Computing. Tartu, 2015, 163 p.
103. **Tauno Metsalu.** Statistical analysis of multivariate data in bioinformatics. Tartu, 2016, 197 p.
104. **Riivo Talviste.** Applying Secure Multi-party Computation in Practice. Tartu, 2016, 144 p.
108. **Siim Orasmaa.** Explorations of the Problem of Broad-coverage and General Domain Event Analysis: The Estonian Experience. Tartu, 2016, 186 p.
109. **Prastudy Mungkas Fauzi.** Efficient Non-interactive Zero-knowledge Protocols in the CRS Model. Tartu, 2017, 193 p.
110. **Pelle Jakovits.** Adapting Scientific Computing Algorithms to Distributed Computing Frameworks. Tartu, 2017, 168 p.
111. **Anna Leontjeva.** Using Generative Models to Combine Static and Sequential Features for Classification. Tartu, 2017, 167 p.
112. **Mozhgan Pourmoradnasseri.** Some Problems Related to Extensions of Polytopes. Tartu, 2017, 168 p.

113. **Jaak Randmets.** Programming Languages for Secure Multi-party Computation Application Development. Tartu, 2017, 172 p.
114. **Alisa Pankova.** Efficient Multiparty Computation Secure against Covert and Active Adversaries. Tartu, 2017, 316 p.
116. **Toomas Saarsen.** On the Structure and Use of Process Models and Their Interplay. Tartu, 2017, 123 p.
121. **Kristjan Korjus.** Analyzing EEG Data and Improving Data Partitioning for Machine Learning Algorithms. Tartu, 2017, 106 p.
122. **Eno Tõnisson.** Differences between Expected Answers and the Answers Offered by Computer Algebra Systems to School Mathematics Equations. Tartu, 2017, 195 p.

DISSERTATIONES INFORMATICAE UNIVERSITATIS TARTUENSIS

1. **Abdullah Makkeh**. Applications of Optimization in Some Complex Systems. Tartu 2018, 179 p.
2. **Riivo Kikas**. Analysis of Issue and Dependency Management in Open-Source Software Projects. Tartu 2018, 115 p.
3. **Ehsan Ebrahimi**. Post-Quantum Security in the Presence of Superposition Queries. Tartu 2018, 200 p.
4. **Ilya Verenich**. Explainable Predictive Monitoring of Temporal Measures of Business Processes. Tartu 2019, 151 p.
5. **Yauhen Yakimenka**. Failure Structures of Message-Passing Algorithms in Erasure Decoding and Compressed Sensing. Tartu 2019, 134 p.
6. **Irene Teinemaa**. Predictive and Prescriptive Monitoring of Business Process Outcomes. Tartu 2019, 196 p.
7. **Mohan Liyanage.** A Framework for Mobile Web of Things. Tartu 2019, 131 p.
8. **Toomas Krips.** Improving performance of secure real-number operations. Tartu 2019, 146 p.
9. **Vijayachitra Modhukur.** Profiling of DNA methylation patterns as biomarkers of human disease. Tartu 2019, 134 p.
10. **Elena Sügis.** Integration Methods for Heterogeneous Biological Data. Tartu 2019, 250 p.
11. **Tõnis Tasa.** Bioinformatics Approaches in Personalised Pharmacotherapy. Tartu 2019, 150 p.
12. **Sulev Reisberg.** Developing Computational Solutions for Personalized Medicine. Tartu 2019, 126 p.
13. **Huishi Yin.** Using a Kano-like Model to Facilitate Open Innovation in Requirements Engineering. Tartu 2019, 129 p.
14. **Faiz Ali Shah.** Extracting Information from App Reviews to Facilitate Software Development Activities. Tartu 2020, 149 p.
15. **Adriano Augusto**. Accurate and Efficient Discovery of Process Models from Event Logs. Tartu 2020, 194 p.
16. **Karim Baghery.** Reducing Trust and Improving Security in zk-SNARKs and Commitments. Tartu 2020, 245 p.
17. **Behzad Abdolmaleki.** On Succinct Non-Interactive Zero-Knowledge Protocols Under Weaker Trust Assumptions. Tartu 2020, 209 p.
18. **Janno Siim.** Non-Interactive Shuffle Arguments. Tartu 2020, 154 p.
19. **Ilya Kuzovkin.** Understanding Information Processing in Human Brain by Interpreting Machine Learning Models. Tartu 2020, 149 p.
20. **Orlenys López Pintado.** Collaborative Business Process Execution on the Blockchain: The Caterpillar System. Tartu 2020, 170 p.
21. **Ardi Tampuu.** Neural Networks for Analyzing Biological Data. Tartu 2020, 152 p.

22. **Madis Vasser.** Testing a Computational Theory of Brain Functioning with Virtual Reality. Tartu 2020, 106 p.
23. **Ljubov Jaanuska.** Haar Wavelet Method for Vibration Analysis of Beams and Parameter Quantification. Tartu 2021, 192 p.
24. **Arnis Parsovs.** Estonian Electronic Identity Card and its Security Challenges. Tartu 2021, 214 p.
25. **Kaido Lepik.** Inferring causality between transcriptome and complex traits. Tartu 2021, 224 p.
26. **Tauno Palts.** A Model for Assessing Computational Thinking Skills. Tartu 2021, 134 p.
27. **Liis Kolberg.** Developing and applying bioinformatics tools for gene expression data interpretation. Tartu 2021, 195 p.
28. **Dmytro Fishman.** Developing a data analysis pipeline for automated protein profiling in immunology. Tartu 2021, 155 p.
29. **Ivo Kubjas.** Algebraic Approaches to Problems Arising in Decentralized Systems. Tartu 2021, 120 p.
30. **Hina Anwar.** Towards Greener Software Engineering Using Software Analytics. Tartu 2021, 186 p.
31. **Veronika Plotnikova.** FIN-DM: A Data Mining Process for the Financial Services. Tartu 2021, 197 p.
32. **Manuel Camargo.** Automated Discovery of Business Process Simulation Models From Event Logs: A Hybrid Process Mining and Deep Learning Approach. Tartu 2021, 130 p.
33. **Volodymyr Leno.** Robotic Process Mining: Accelerating the Adoption of Robotic Process Automation. Tartu 2021, 119 p.
34. **Kristjan Krips.** Privacy and Coercion-Resistance in Voting. Tartu 2022, 173 p.
35. **Elizaveta Yankovskaya.** Quality Estimation through Attention. Tartu 2022, 115 p.
36. **Mubashar Iqbal.** Reference Framework for Managing Security Risks Using Blockchain. Tartu 2022, 203 p.
37. **Jakob Mass.** Process Management for Internet of Mobile Things. Tartu 2022, 151 p.
38. **Gamal Elkoumy.** Privacy-Enhancing Technologies for Business Process Mining. Tartu 2022, 135 p.
39. **Lidia Feklistova.** Learners of an Introductory Programming MOOC: Background Variables, Engagement Patterns and Performance. Tartu 2022, 151 p.
40. **Mohamed Ragab.** Bench-Ranking: A Prescriptive Analysis Approach for Large Knowledge Graphs Query Workloads. Tartu 2022, 158 p.
41. **Mohammad Anagreh.** Privacy-Preserving Parallel Computations for Graph Problems. Tartu 2023, 181 p.
42. **Rahul Goel.** Mining Social Well-being Using Mobile Data. Tartu 2023, 104 p.

43. **Anti Ingel.** Algorithms using information theory: classification in brain-computer interfaces and characterising reinforcement-learning agents. Tartu 2023, 142 p.
44. **Shakshi Sharma.** Fighting Misinformation in the Digital Age: A Comprehensive Strategy for Characterizing, Identifying, and Mitigating Misinformation on Online Social Media Platforms. Tartu 2023, 158 p.
45. **Kristiina Rahkema.** Quality Analysis of iOS Applications with Focus on Maintainability and Security Aspects. Tartu 2023, 182 p.
46. **Ivan Slobozhan.** Studying Online Social Media Engagement in CIS Countries during Protests, Mass Demonstrations and War. Tartu 2023, 81 p.
47. **Nurlan Kerimov.** Building a catalogue of molecular quantitative trait loci to interpret complex trait associations. Tartu 2023, 248 p.
48. **Pavlo Tertychnyi.** Machine Learning Methods for Anti-Money Laundering Monitoring. Tartu 2023, 117 p.
49. **Abasi-amefon Obot Affia.** A Framework and Teaching Approach for IoT Security Risk Management. Tartu 2023, 180 p.
50. **Raimond-Hendrik Tunnel.** Video Game Design and Development Bachelor's Curriculum for Estonia. Tartu 2024, 137 p.
51. **Ahto Salumets.** Bioinformatics analysis of various aspects in immunology. Tartu 2024, 198 p.
52. **Mohammed Abdulhameed Shaif Ali.** Deep Learning Methods for Cell Microscopy Image Analysis. Tartu 2024, 143 p.
53. **Pille Pullonen-Raudvere**. Foundations of Efficient and Secure Algorithm Development for Secure Multiparty Computation. Tartu 2024, 265 p.
54. **Marili Rõõm.** Multiple approaches to learners' success and factors affecting it in computer programming MOOCs. Tartu 2024, 170 p.
55. **Shivananda Rangappa Poojara.** Design and Orchestration of Scalable, Event-Driven Serverless Data Pipelines for Internet of Things (IoT) Applications. Tartu 2024, 172 p.
56. **Hassan Abdulgaleel Hassan Salim Eldeeb.** Empowering Machine Learning Pipelines with Automated Feature Engineering. Tartu 2024, 121 p.
57. **Muhammad Uzair.** Soft decision making for agri-food 4.0. Tartu 2024, 158 p.
58. **Kirill Milintsevich.** Estimation of Depression Level from Text: Symptom-Based Approach, External Knowledge, Dataset Validity. Tartu 2024, 130 p.
59. **Maksym Del.** Multilingual and Multi-Domain Representational Patterns Across Trpansformer-Based Models. Tartu 2024, 131 p.
60. **Kristo Raun.** Adaptive Out-of-order Handling in Streaming Conformance Checking. Tartu 2024, 118 p.
61. **Toivo Vajakas.** Towards integration of mobile network data into analyzing human mobility. Tartu 2024, 103 p.
62. **Katsiaryna Lashkevich.** Data-Driven Analysis and Optimization of Waiting Times in Business Processes. Tartu 2024, 169 p.
63. **Alejandra Duque-Torres.** Classifying, Constraining and Ranking Metamorphic Relations. Tartu 2025, 159 p.

64. **Mariia Bakhtina.** A Method for Information Security and Privacy Management in Smart Solutions. Tartu 2025, 199 p.
65. **Andre Tättar.** Multilingual Machine Translation for Under-Resourced Languages. Tartu 2025, 170 p.
66. **Mahmoud Shoush.** Prescriptive Process Monitoring Under Uncertainty and Resource Constraints. Tartu 2025, 178 p.
67. **Alireza Akhavi Zadegan.** A Multimodal approach for refining Mapping and Localization by Integrating Generative AI and Pedestrian-Centric Data. Tartu 2025, 147 p.
68. **Eerik Muuli.** Automating the assessment and feedback processes in IT teaching – improving creation and maintenance from the teaching staff perspective. Tartu 2025, 196 p.