FlexiTeam: Flexible Team and Work Organization using Process-Oriented Case-Based Reasoning

Ditty Mathew[1,*], Ralph Bergmann[1,2], Benjamin Weyers[1], Thomas Ellwart[1], Dominique Bohrmann[1] and Ericson Hölzchen[1]

[1]*University of Trier, Germany*

[2]*German Research Center for Artificial Intelligence (DFKI), Germany*

Abstract

During crises such as COVID-19, there is a need to adapt existing work processes and teams to the changing environment in a very flexible and dynamic way in many business and healthcare organizations. In this paper, we conceptualize the advances required for Process-Oriented Case-Based Reasoning to flexibly and dynamically organize human resources in a team and work processes. The novel contributions of this paper include an extended case representation to represent resources, profiles, and key performance indicators (KPIs) of processes, a query definition which covers the "context", and an overall process to flexibly and dynamically organize work processes and human resources. We evaluate the FlexiTeam process using a cooking recipe casebase and analyze the quality of the retrieval using a quality measure. We also derive the research questions that need to be addressed to fully explore this approach and the specific difficulties involved in solving this problem.

Keywords

Process-Oriented Case-Based Reasoning, Process Management, Workflow Management

1. Introduction

In the current COVID-19 pandemic, healthcare organizations, but also many businesses, are having to adapt their existing work processes[1] and teams[2] to the changing environment in a very flexible and dynamic way [1]. Health authorities have to increase staff for contact tracing and train auxiliary staff at short notice. Clinics have to increasingly consolidate their nursing staff in intensive care units, companies have to make work processes suitable for home work and change their team composition so that the entire company does not come to a standstill in the event of infections [2, 3]. The concrete requirements for this redesign are constantly changing due to the parallel process of gaining knowledge about the concrete risk factors in the spread of the pandemic and require an agile redirection, which is associated with great effort and uncertainty [3]. From a process perspective, robustness, flexibility, and efficiency must be weighed against each other in teams [1]. In addition, shifts in business areas can lead to extreme situations of overload and underload, or to situations where employees in the team

ICCBR POCBR'22: Workshop on Process-Oriented Case-based Reasoning at ICCBR-2022, September, 2022, Nancy, France
*Corresponding author.
mathew@@uni-trier.de (D. Mathew)

[1]we use the term *work process* and *task* alternatively to denote the work process in the work
[2]we define the term "team" as a set of human resources who are available to perform the work processes.

are overburdened and under-challenged [2]. Currently, decisions to reorganize teams are often made "on instinct" without a solid basis for decision-making [1]. More sophisticated tools are necessary to deal with these challenges and this is the central focus of the FlexiTeam[3] project. By using the data available in organizations on personnel, resources, and organizational processes, combined with knowledge from cognitive science [4] and industrial psychology [5] as well as concrete experience from comparable issues in earlier crises, new digital tools can be designed to support the flexible organization of work teams.

An important source of knowledge for many decisions and thus also for the organization of the team and work processes is the experiential knowledge. Case-Based Reasoning (CBR) [7] allows the realization of experience-based problem-solving processes, in which the access to previous experiential knowledge as well as its adaptation according to the current problem situation can be (partially) automated. Existing approaches and prototypical systems of the so-called Process-Oriented Case-Based Reasoning (POCBR) [6] are, due to their rich knowledge representation, from their basic orientation also suitable for the representation and processing of experiential knowledge about work processes and the associated team structures. They are also capable of making suggestions for adaptation in the sense of organization of teams and work by analyzing larger stocks of experience. This paper is motivated by the research question - "How can POCBR be applied to flexibly and dynamically organize team and work processes to be prepared for crises like COVID-19 and what are the research challenges that have to be addressed for this purpose?". In this paper, we conceptualize the advances required for POCBR to cater to the flexible organization of the work processes and human resources in the team.

To address the above research question, we recapitulate the current state of the art in POCBR in Section 2, we analyze the potential of POCBR for the FlexiTeam problem and identify the missing part in the case representation of the current state of POCBR using the examples illustrated in Section 3. We propose a general approach in Section 4 to address the current POCBR deficits and the novel contribution of this paper includes i) an extended case representation to represent resources, profiles, and key performance indicators (KPIs) of processes, ii) a query definition which covers the "context", i.e. available resources, iii) an overall FlexiTeam process to flexibly and dynamically organize work processes and human resources. Section 5 presents the evaluation results using a cooking recipe casebase and analyze the quality of the retrieval using a quality measure.. In Section 6, we derive the research questions that need to be addressed to fully explore this approach and the specific difficulties involved in solving this problem.

2. Foundations and Previous Work

POCBR aims at applying CBR methods and principles in process and workflow management. The workflow modeling involves problem solving activities such as identification and creation of a suitable workflow process model for a particular scenario. Therefore, it is important to have a case representation for a workflow and a similarity measure to identify similar workflows.

Bregmann and Gil [8] introduced the NEST graph which is a semantically annotated directed graph to represent the workflow. It is defined as a quadruple $W = (N, E, S, T)$ where N indicates a set of nodes, E denotes a set of edges between nodes in N , S is a function that

[3]we use the abbreviation FlexiTeam to denote the problem of flexibly organizing team and work processes.

assigns a semantic description to each node and each edge from a semantic metadata language like ontology and T is a function that assigns a specific type to each node and each edge. The nodes and edges build the structure of each workflow, whereas the semantic information of the workflow is modeled by the types and the semantic descriptions of nodes and edges.

Consider the example of the sandwich recipe workflow depicted in Figure 1 to demonstrate the NEST graph representation. In this workflow, the cooking steps are represented in the task nodes and the ingredients that belong to the corresponding cooking steps are represented as data nodes. Each workflow graph represents a cooking recipe. The control flow edges connect the task nodes and it defines the order in which the tasks are executed. The data flow edges draw connections between task nodes and data nodes. The task nodes consume the inputs from the data nodes and produce outputs to the data nodes which can be further consumed by other tasks. Semantic descriptions of task nodes and data nodes are used to specify semantic information of the workflow components as attribute-value pairs. The semantic description of the task node Toast is given in the workflow graph. Here, a toaster is needed to execute the task Toast and it is represented by the attribute name Auxiliaries.

In POCBR, a query is a single partial workflow that represents the tasks, data and structural relationships of the desired workflow. The user may also want to express the undesired elements of the workflow. Process-Oriented Query Language (POQL) is the query language developed for representing the queries of POCBR [9]. A POQL query involves a desired workflow and a set of restriction workflows. The desired workflow represents the desired properties of the workflow that is searching for and each restriction workflow represents an undesired situation that can be avoided. More precisely a POQL query $Q = DW \wedge \neg RW_1 \ldots \wedge \neg RW_n$ where DW denotes the desired workflow and each RW_i denotes a restriction workflow. For example, consider the query as searching for a sandwich recipe with tomato and grated cheese, but without grilled meat. Here the workflow for the sandwich recipe with ingredients tomato and grated cheese is the part of desired workflow DW and the workflow for sandwich recipe with grilled meat as ingredient is part of a restriction workflow RW_1.

During case retrieval, similar workflows are identified by finding similarities between the query workflow and a case workflow. The similarity between the desired workflow and the case workflow, and the similarity between the restriction workflow and the case workflow are computed in the same way. Müller and Bergmann [9] adopted the approach from fuzzy logic [10] to deal with the negations and conjunctions in the POQL query when there are restriction workflows. To compute the similarity between two NEST graphs, a similarity measure is required to capture the link structure of nodes and edges as well as the semantic descriptions and types of these workflow elements. Bergmann and Gil [8] propose a semantic similarity measure that determines the similarity based on the local-global principle [11]. The global similarity denotes the similarity between two graphs and it is composed of local similarities which is the pairwise similarity between nodes and edges.

3. FlexiTeam Workflow Representation

In order to apply POCBR to flexibly and dynamically organize human resources and work processes during a crisis, we need to consider human resources allocated to the work processes

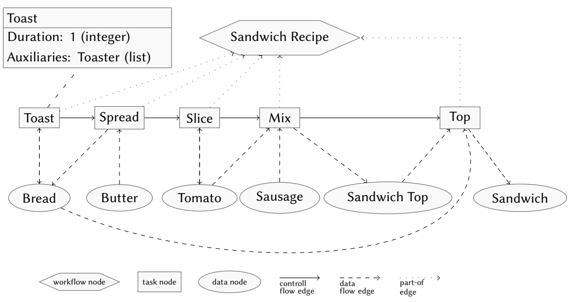

Figure 1: Cooking Recipe Example represented as NEST Graph

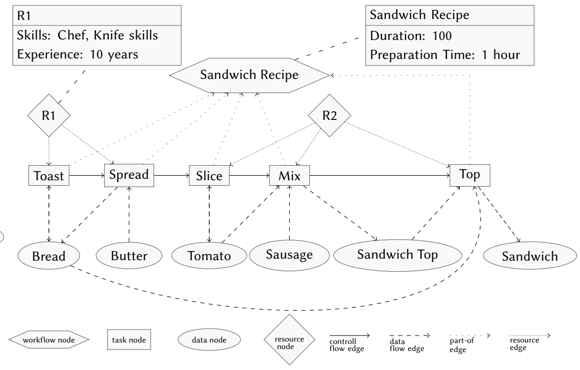

Figure 2: FlexiTeam workflow representation of cooking recipe

in the workflow representation. Here we use an enhanced version of *NEST* graph to represent workflow with the details of allocated human resources to each task in the workflow. A new type of nodes and edges called "resource nodes" and "resource edges" are included to represent the human resources (e.g. cook) allocated to the work processes. The semantic description of each resource node contains the profile of the resource which elaborates the skills, experience, qualification, etc. This profile information can be represented as attribute-value pairs inside the semantic description of a resource node. For example, a list of skills as value for the attribute "skills". A resource edge connects a resource node to the task node to which the resource is allocated. Figure 2 illustrates an example in the cooking recipe domain where the overall goal is to prepare 100 Sandwiches and the preparation time is 1 hour. The workflow node describes the team-related features such as the quantity of work, work duration, etc. as part of its semantic description. In this example, two human resources named R1 and R2 are allocated to complete the tasks. These resources are represented as resource nodes R1 and R2. The tasks such as Toast and Spread are allocated to the resource R1 and the tasks such as Slice, Mix, and Top are allocated to the resource R2. The profile of the resource R2 is represented as attribute-value pairs in the semantic description of the resource node R2. There can be different ways in which the resources and tasks (work processes) can be organized for the same workflow. For example, a sandwich can be prepared by three human resources where two of them are assigned to do the tasks such as Toast and Spread and one resource for the other tasks. Each workflow case represents the structure in which the resources and work processes are organized.

4. FlexiTeam Process Overview

The overall retrieval process to flexibly organize teams and processes may involve the retrieval of multiple workflows. For example, consider the query as a job to allocate human resources and organize work processes or tasks to prepare 100 sandwiches and 100 burgers. The workflow casebase may contain separate workflows for the sandwich recipe and burger recipe. The tasks such as slicing vegetables and spreading butter can be shared for both sandwich and burger.

As there can be various combinations of workflows that can match the query goal, it is not efficient to store all such combinations in the workflow casebase. Hence, there is a need to split the query into different parts. Thus the retrieval process of FlexiTeam includes - i) define query to handle subqueries, ii) retrieve workflow for each subquery, iii) combine the retrieved workflows, and, iv) rank the combined workflows. Section 4.1 discusses the query definition of FlexiTeam query and Section 4.2 describes the workflow retrieval process.

4.1. Query Definition

A POQL query deals with a desired workflow and several restricted workflows as discussed in Section 2. However, in FlexiTeam, team organization may involve processes of multiple workflows. So, a query in FlexiTeam contains multiple desired workflows and restriction workflows for each desired workflow. Though the features like profiles of available human resources for the job, work duration, etc. can be represented in the workflow as described in Section 3, the query may provide overall details of these features instead of individual details for each subquery. We name such overall features as FlexiTeam features, e.g., resource profiles available for the entire processes, total work duration. These features are important to combine the workflows of subqueries and to arrive at an aggregated workflow by matching the resource profiles. More precisely, the FlexiTeam process query Q can be represented as the combination of $<$Query workflows, FlexiTeam features$>$. The Query workflows can be further decomposed into a set of POQL subqueries $\{qw_1, qw_2, \ldots qw_m\}$. Each subquery qw_i is defined as $qw_i = dw_i \wedge \neg rw_{i1} \wedge \ldots \wedge \neg rw_{in}$, where dw_i is the desired workflow of qw_i and each rw_{ij} where $1 \leq j \leq n$ represents a restriction workflow of query qw_i. For example, consider the query "Prepare 50 vegetable sandwiches, 100 vegetable burgers; Available resources: 2 cooking helpers, 2 chefs; and Work duration: 3 hours". In this example, there are two workflow subqueries. One for the vegetable sandwich and another one for the vegetable burger. The FlexiTeam features of this query include i) Available resources, i.e., 2 cooking helpers, 2 chefs and their profiles, ii) Work duration, i.e., 3 hours

4.2. Workflow Retrieval Process

During the retrieval process, each subquery workflow is represented in POQL language and top-k similar workflows can be retrieved from the workflow casebase for each query. In the query example described in Section 4.1, the desired and restriction workflows of each dish can be represented using the POQL language introduced by Müller and Bergmann [9]. The similarities between the subquery workflow and the case workflow can be computed using the ProCAKE framework [12].

We need to combine the workflows retrieved for the subqueries to analyze the FlexiTeam properties of the overall query. An approach to combining the workflows is to choose a workflow from the retrieved results of each subquery and merge them. More specifically, choose one case workflow at a time from the retrieved workflows of a subquery and take m such case workflows for m subqueries. For example, one such combination is the case workflow cw_{1u} from the retrieved results of qw_1, cw_{2v} from qw_2, and so on, i.e., $(cw_{1u}, cw_{2v}, \ldots cw_{mw})$ where $1 \leq u, v, w \leq k$. Such a combination of case workflows can be considered as the combined

Algorithm 1: FlexiTeam Process Algorithm

```
   Input: Q =< QW, TF >
   Output: Best combined workflows based on the ranking
1  Decompose the query workflow as QW = {qw_1, qw_2, ... qw_m}
2  for each workflow subquery qw_i do
3  |  Represent qw_i as a POQL query; dw_i ∧ ¬rw_i1 ... ∧ ¬rw_in
4  |  Retrieve top-k workflows using the POQL query language of qw_i; Say cw_i1, ... cw_ik are the top-k
   |    retrieved workflows for the query qw_i
5  Create candidates for the combined workflow by choosing a retrieved workflow of each workflow subquery
6  for each combined workflow candidate do
7  |  Match the FlexiTeam features of the query with the candidate workflows
8  |  Estimate the CombinedWorkflowScore
9  Rank combined workflow candidates based on the CombinedWorkflowScore
```

workflow candidates for the final result of the overall query.

We estimate a score for each combined workflow candidate based on the extent to which the candidate matches the overall query requirements. Let $\{qw_1, qw_2, \ldots qw_m\}$ be the set of subqueries and the combined workflow candidate of the subqueries be $(cw_{1u}, cw_{2v}, \ldots cw_{mw})$ where $1 \leq u, v, w \leq k$ and k denotes the top-k case workflows that are retrieved for each subquery. The *CombinedWorkflowScore* is computed as given below.

$$\mathit{CombinedWorkflowScore} = \frac{\alpha}{m}\sum_{i=1}^{m} \mathit{sim}(qw_i, cw_i) + \frac{(1-\alpha)}{|\mathit{FlexiTeam}_{feat}|}\sum_{f \in \mathit{FlexiTeam}_{feat}} \mathit{sim}_f(q_f, c_f) \tag{1}$$

The combined workflow score depends on two factors, i) the aggregation of the similarity $\mathit{sim}(qw_i, cw_{ij})$ for each subquery qw_i where cw_{ij} indicates the workflow that is chosen from the top-k retrieved case workflows for the subquery qw_i, and, ii) the similarity between the FlexiTeam features ($\mathit{FlexiTeam}_{feat}$) of the combined workflow candidate and the query.

The overall FlexiTeam process is summarized in Algorithm 1. The lines 1 to 4 describes the query decomposition into POQL subqueries and retrieval of case workflows for the POQL subqueries. Line 5 talks about the creation of combined workflow candidates and lines 6 till 8 discuss the assignment of a score to the combined workflow candidate. Finally, in line 9, combined workflow candidates are ranked based on the assigned score. The best combined workflows can be recommended as output based on the ranking of combined workflow candidates.

5. Experimental Evaluation

We synthetically extend the cooking recipe casebase [13] which contains 40 recipes to evaluate the algorithm proposed in Algorithm 1. We assume that the qualifications of all the resources who perform the cooking tasks are the same. We create multiple cases from a recipe case in the existing casebase by adding different combinations of FlexiTeam features such as the resource nodes, quantity of dishes to be prepared, and the duration to prepare the required quantity of dish. The work duration and the number of resources are randomly generated between the ranges of one to four hours and one to ten resources respectively. The quantity of dishes to be prepared is calculated using the following formula,

$$quantity = \frac{(duration \times no.\ of\ resources)}{preparation\ time\ for\ unit\ quantity} \tag{2}$$

The preparation time for unit quantity is available in the existing case-base. Thus we synthetically generate FlexiTeam features using the preparation time provided in the semantic description of cases. We generate 15 sets of FlexiTeam features and one set of FlexiTeam query features for each case or recipe. As we can use the ProCake framework to retrieve the workflows, we evaluate only the workflow merging process. We use the FlexiTeam query features of each recipe as a subquery and assume that the corresponding cases that are generated for that recipe using FlexiTeam features are the retrieved workflows of that subquery. Hence, the $sim(qw_i, cw_i)$ is always one for all the subqueries. The similarity measure for each FlexiTeam feature, $sim_f(q_f, c_f)$ are given below.
Number of resources and **Duration**: If the number of resources in the combined workflow candidate is less than the number of resources in the query, then there is no need to make use of all available resources. Hence it is a perfect match. If the number of resources in the case exceeds the number of resources in the query, a linear similarity measure represents how much more resources are needed than available. This is the same case for the *duration* feature.

$$sim_f(q_f, c_f) = \begin{cases} 1 & \text{if } c_f \leq q_f \\ LinearSim_f(q_f, c_f) & \text{otherwise} \end{cases} \tag{3}$$

Quantity: We use a symmetric measure for the quantity feature ($sim_f(q_f, c_f) = LinearSim_f(q_f, c_f)$). If the quantity of the case is lesser than that of the query, need to include more resources based on the quantity difference, and vice versa. $LinearSim$ is estimated as,

$$LinearSim_f(q_f, c_f) = 1 - \frac{|q_f - c_f|}{MaxValue(f) - MinValue(f)} \tag{4}$$

MaxValue(f) and *MinValue(f)* are the maximum and minimum values for the feature f in the casebase respectively.

We consider the top-ranked combined workflow candidate as the retrieved workflow and performed the experiments with the number of subqueries $m = 2$ and $m = 3$ where each subquery corresponds to a recipe. The FlexiTeam features are generated for each subquery and they are combined by adding the values of features such as the number of resources and duration. The quantity of the item of each subquery is considered as a separate feature for the combined FlexiTeam features. The FlexiTeam features of the combined workflow candidates are also generated in the same way. The retrieved combined workflow candidate is evaluated across various retrieval sizes of subqueries (k). We analyze the time complexity with the different number of subqueries (m) and the different number of cases retrieved for subqueries (k) and it is illustrated in Figure 3. The complexity of the combination (lines 6-9 in Algorithm 1) depends on the number of cases retrieved for each subquery, exponentially $O(k^m)$. Thus k should be limited. We also evaluate the average quality of all the top-ranked combined workflows for all the queries with the values of each combination of m and k. The quality depends on the

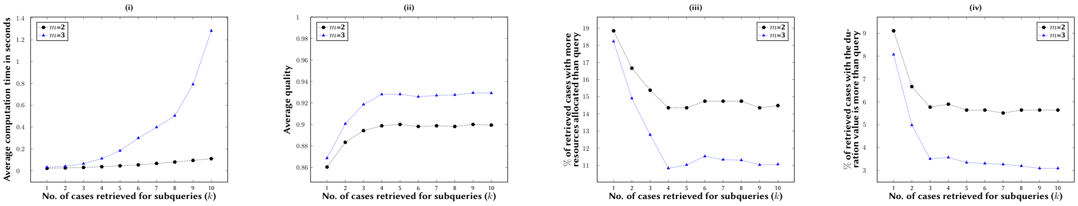

Figure 3: (i) Time complexity analysis, (ii) Quality analysis, (iii) Percentage of retrieved cases with more resources allocated than query, (iv) Percentage of retrieved cases with the more duration than query

formation of the retrieved team and we define it based on the FlexiTeam features such as number of resources and the duration. The average quality of given values of k and m is defined as,

$$avg\_quality(k,m) = 1 - \frac{1}{2}\left(\frac{\#Cases_{\#resources(case)>\#resources(query)} + \#Cases_{duration(case)>duration(query)}}{\#queries}\right) \tag{5}$$

where $\#Cases_{\#resources(case)>\#resources(query)}$ indicates the number of cases with number of resources allocated more than that of the query and $\#Cases_{duration(case)>duration(query)}$ means the number of cases with the duration value is more than that of the query. We also analyze the percentage of retrieved cases with more resources allocated than that of the query and the percentage of retrieved cases with the duration value is more than that of the query. These analyses are depicted in Figure 3. We can observe that the quality increases with the increase in the value of the number of cases retrieved for subqueries (k) till $k = 4$ and then it is almost saturated. The percentage of retrieved cases with more resources allocated than query decreases drastically till $k = 4$ and then there is not much decrease happen with $m = 2$ and $m = 3$. The analysis of the duration feature shows that the values are decreasing till $k = 3$. These analyses show that the value of k can be limited to 4.

6. Future Research Directions

We derive the following research questions that need to be addressed to fully explore the proposed FlexiTeam approach.

1. One future research direction would be to optimize the aggregated workflows by identifying the shared tasks between case workflows in the combined workflow candidate. It would be interesting to study such optimization methods while merging the workflows and devise similarity measures to identify shared tasks. There are studies in Human Computer Interaction (HCI) to meaningfully decompose tasks during task modelling for interactive system design [15].
2. There are many similarity measures in the literature to match the skill profiles. In this paper, we represent the profiles of human resources as part of the workflow [14]. To the best of our knowledge, this is the first approach to extend the workflow representation to add resource nodes and their profiles. It would be interesting to derive a similarity measure to match the resource profiles between workflows by using the properties of the workflows.
3. It is critical to assign work processes to the human resources based on their capability,

workload, and mental load. It would also be interesting to study the cognitive aspects of team members such as the collaborative utilization of the technical resources available to the team [16] and cognitive architectures such as ACT-R [17] to understand how human resources organize knowledge and produce intelligent behavior during their work.
4. The successful assistance and the integration of AI-based digital systems into existing team structures and work processes also require graphical user interfaces and interactive visual interfaces for the transmission of the necessary information by the AI system. This becomes particularly important in the case of exceptional situations such as the COVID-19 pandemic, in which all those involved in crisis management are moving on unfamiliar terrain and in a highly critical situation. By combining POCBR approaches with the methods from HCI, new possibilities arise for designing experience-based systems as interactive systems that enable cooperative solution finding in a dialogue between humans and machines.
5. From a human factors perspective, human decision-makers need to accept and trust the AI-based system to integrate digital expertise in human resource related decision processes. Research from human-autonomy teaming [18] points out various factors, such as system knowledge, system reliability, and usefulness, or adaptability. The FlexiTeam process presented in this paper could not only provide a data basis for the AI, but at the same time, also integrate the human subsystem in the technical solution and promote user acceptance and development of the decision-makers. For example, workflow visualizations may serve as externalized shared team cognition and foster system knowledge. Involving human decision-makers in the definition of queries and retrieval of workflows will lead to human task reflection and elaboration – both key processes for team development and adaptation. Thus, future research may investigate how task changes and user feedback can be continuously integrated into the data-driven decision-making process to account for team adaptation and stimulate a socio-digital learning process.

7. Conclusion

In this paper, we present an important problem of a flexible team and work organization and explore POCBR towards solving this problem. We propose a general approach to address the current POCBR deficits and evaluate the proposed algorithm using a cooking recipe casebase. The successful assistance and the integration of such AI-based systems into existing team structures and work processes require not only the analysis of team structures and processes but also the graphical user interfaces and interactive, visual interfaces for the transmission of the necessary information by the AI system. It would be interesting to complement AI methods with methods from HCI and Psychology research to enable successful communication between human users and the AI system. In this combination, a new form of organizational and business intelligence can be created to increase resilience in crises, which can create benefits in pandemic crises as well as in other challenging situations.

8. Acknowledgement

This work has been funded by the Ministry of Science and Health of the German Federal State of Rhineland-Palatinate.

References

[1] Boiral, O., Brotherton, M.C., Rivaud, L. and Guillaumie, L.: Organizations' Management of the COVID-19 Pandemic: A Scoping Review of Business Articles. Sustainability, 13(7), p.3993 (2021)
[2] Marin-Garcia, J.A., Garcia-Sabater, J.P., Ruiz, A., Maheut, J. and Garcia-Sabater, J.J.: Operations Management at the Service of Health Care Management: Example of a Proposal for Action Research to Plan and Schedule Health Resources in Scenarios Derived from the COVID-19 Outbreak. Journal of Industrial Engineering and Management, 13(2), pp.213-227 (2020)
[3] Vnoučková, L.: Impact of COVID-19 on Human Resource Management. Rev. Latinoam. Investig. Soc.3, pp. 18–21 (2020).
[4] Stillings, N.A., Chase, C.H., Weisler, S.E., Feinstein, M.H., Garfield, J.L. and Rissland, E.L.: Cognitive Science: An Introduction. MIT press. (1995)
[5] Tiffin, J. and McCormick, E.J.: Industrial Psychology. (1965)
[6] Minor, M., Montani, S., Recio-García, J.A.: Process-Oriented Case-based Reasoning. Inf. Syst. 40, pp. 103–105 (2014).
[7] Riesbeck, C.K. and Schank, R.C.: Inside Case-Based Reasoning. PLawrence Erlbaum, Hillsdale, NJ, (1989)
[8] Bergmann, R. and Gil, Y.: Similarity Assessment and Efficient Retrieval of Semantic Workflows. In: Information Systems, 40, pp.115-127 (2014).
[9] Müller, G. and Bergmann, R.: POQL: A New Query Language for Process-Oriented Case-Based Reasoning. In: Proceedings of the LWA, pp. 247-255 (2015).
[10] Burkhard, H.D. and Richter, M.M.: On the Notion of Similarity in Case Based Reasoning and Fuzzy Theory. In Soft Computing in Case Based Reasoning, pp. 29-45 (2001).
[11] Richter, M.M.: Foundations of Similarity and Utility. In FLAIRS, pp. 30-37 (2007).
[12] Bergmann, R., Grumbach, L., Malburg, L. and Zeyen, C.: ProCAKE: A Process-Oriented Case-Based Reasoning Framework. In Workshop Proceedings of International Conference on Case-Based Reasoning, pp. 156-161 (2019).
[13] Müller, G.: Workflow Modeling Assistance by Case-Based Reasoning. In Springer. pp. I-XVII (2018).
[14] Biesalski, E. and Abecker, A.: Similarity Measures for Skill-Profile Matching in Enterprise Knowledge Management. In ICEIS (2), pp. 11-16 (2006).
[15] Bowen, J., Dittmar, A. and Weyers, B.: Task Modelling for Interactive System Design: A Survey of Historical Trends, Gaps and Future Needs. Proceedings of the ACM on Human-Computer Interaction, 5(EICS), pp.1-22 (2021).
[16] Palmer, T.D. and Fields, N.A.: Computer Supported Cooperative Work. Computer, 27(5), pp.15-17 (1994)
[17] Ritter, F.E., Tehranchi, F. and Oury, J.D.: ACT-R: A Cognitive Architecture for Modeling Cognition. Wiley Interdisciplinary Reviews: Cognitive Science, 10(3), p.e1488 (2019)
[18] O'Neill, T., McNeese, N., Barron, A. and Schelble, B.: Human–Autonomy Teaming: A Review and Analysis of the Empirical Literature. In Human FactorHuman Factors: The Journal of the Human Factors and Ergonomics Society, pp. 1–35 (2020).