TECHNICAL UNIVERSITY OF CLUJ-NAPOCA

ACTA TECHNICA NAPOCENSIS

Series: Applied Mathematics, Mechanics, and Engineering
Vol. 64, Issue Special I, January, 2021

ALGORITHM AND MODEL FOR KNOWLEDGE CAPITALIZATION IN AUTOMOTIVE RESEARCH AND DEVELOPMENT OFFSHORE BRANCHES

Marius GAL, Claudiu V. KIFOR

Abstract: *In the Research and Development (R&D) offshore branches, employees with different levels of expertise and knowledge can be found in the newly created teams. The orders coming from the headquarters are usually assigned to the team members without such criteria like complexity of the order, or the expertise of the employee required for that order, the orders being assigned based on their priority, to the employees available at that moment. The results can be, in many cases, exceeded deadlines, low quality of the software products, a high rework rate and a high pressure on employees. In our attempt to solving this problem, we propose an algorithm for orders allocation based on the order's priority, complexity and employee's availability and expertise. The algorithm was tested in a real-life scenario and results are also presented in this paper.*
Key words: *management, expertise, resources allocation, software development quality, offshore, automotive.*

1. INTRODUCTION

The success of a company is the effect of a clockwork mechanism based on a set of industry specific procedures and processes. Actions and reactions in different contexts, are managed through the processes for realizing the organization's goals. To warrant the company's advantage or sometimes the survival on the market, the managers must monitor and control all the business processes to apply with success the new strategies [1].

In the last twenty years a deep change happened in the vehicle manufacturing, meaning that the number of Electronic Control Unit's on a car have increased in a remarkable way. Electronics and software control approximately 70% of modern car's functionality and studies predict that this percentage will increase more [6, 7]. This means that if the software companies from automotive industry can provide high quality and performant software to the clients, the premises of growth have a high probability to happen.

The quality of the Original Equipment Manufacturers (OEM's) products will decide their future, and to maintain or have a bigger part of the market, they are focusing on the improvement of practical and managerial processes.

The automotive companies must work more on the following aspects: additional focus on project rather than a pure process-centered approach, better-quality technical management and strong links to established automotive quality frameworks. The objective of a standard is to offer a scheme for software development process capability and a path for the improvement of the company. The process capability is a characterization of the process's ability to meet the business goals [7].

When discussing about software development, the traditional way of development was in one location where the whole team worked at the same office. In the last years the activity of software development had become more expensive at the headquarters, so the companies started to look for options.

One solution which can boost the firms' balance on middle and long term is internationalization or offshoring. Traditionally, the offshoring strategy included mostly manufacturing activities, but in the recent years this strategy started to include also "administrative and technical services as well as advanced services and R&D-related functions" [9, 2, 3, 4]. This means searching the best cost country, and in that country the best cost city for doing business that matches the company's profile. The researchers state that multinationals from very strong markets began to expand their R&D units in China and India, this meaning a new direction for the R&D centers, which were placed until 1980's very close to the headquarters of the companies [11]. Strongly linked with internationalization is offshoring. According to [5], offshoring of production is "*relocation of parts of production to own locations abroad as well as to other suppliers abroad*."

Knowledge management and business process implementation can support the companies in obtaining higher capabilities by using at the maximum potential knowledge resources. This fact should be used also in software development. The software environment is in continuous change, and when implementing the best processes inside the company, the stakeholders should take part when discussing the improvement or the change of a process and contribute with their ideas and vision. This step is very important on the long run and will help in reducing the costs, avoiding problems and help the companies with quality improvement [8].

Knowledge Management (KM) is a very important point for every company in planning, organizing, motivate and control people processes, procedures, routines and structures for making sure that the resources are used efficiently according to the expertise and are continuously improved. KM is a method for helping the company reaching its targets by gathering, generating and distributing information, visions, designs and know-hows, which can impact the company on the long run [8].

Business processes are sets of interacting or interconnected actions that transform assets or inputs into outputs. For adding value to the organization, each process is planned as a part of a workflow that is checked and measured. The business process is the link between employees, and technology for supporting the company in achieving the financial results, vision and project's objectives [1].

A business process must have the following attributes: detailed objectives, input, output, efficient resources usage, multi-stage activity, at least one unit in an organization and quality for costumers [8].

For accomplishing the aims, every organization must maintain and adapt the business processes and management system including the following:

- For every process, the inputs must be defined for meeting the expected outputs.
- Decide the implementation order and connections of the group processes.
- Decide which criteria and methods must be applied for ensuring the software development or testing efficiency and process control.
- Allocating and assigning proper resources for all the processes.
- Passing on tasks, responsibilities and authorizations for the processes.
- Managing risks and opportunities related to all the necessary interactions.
- Ensuring that the performance indicators are reached by updating the processes.
- Integration and continuous improvement of the business process in the organizations' management [1].

The processes must be concentrated on necessary improvements, ease automation, efficient process flow, increase the productivity and decrease the number of employees implicated in the process [10]. In this paper, the knowledge management and business process will be both integrated in our proposed algorithm for both knowledge increase and continuous improvement of the interactions between the staff.

In the Research and Development (R&D) offshore branches, employees with different levels of expertise and knowledge can be found in the newly created teams. The orders coming from the headquarters are usually assigned to the team members without such criteria like complexity of the order, or the expertise of the employee required for that demand, the orders being assigned based on their priority, to the employees available at that moment. The results can be, in many cases, exceeded deadlines, low quality of the software products, a high rework rate and a high pressure on employees.

2. ALGORITHM FOR ASSIGNING ORDERS BASED ON THEIR COMPLEXITY AND THE EMPLOYEES EXPERTISE

To solve the problem, we introduce the $I_{E/C}$ indicator (1), that can help teams in assigning orders to the most qualified employees.

$$I_{E/C} = \frac{Expertise\ of\ the\ employee}{Complexity\ of\ the\ order} \in [0{,}5; 1{,}5]. \quad (1)$$

Both the employees' expertise and orders complexity are ranked from 1 to 3, using the following criteria:

Employee expertise:

1 - Low expertise employees - employees new in the team, or with limited experience. They can solve low complexity tasks.

2 - Medium expertise employees- they can handle easily orders and medium complexity orders.

3 - High expertise employees - these employees can solve with no difficulty orders at any level of complexity.

Similarly, the **order's complexity** is categorized as it follows:

1 - Low complexity orders - orders that are relatively simple and do not require a detailed knowledge and experience to be completed.

2 - Medium complexity orders - orders needed a certain amount of knowledge and expertise to be fulfilled in the specified time and quality.

3 - High complexity orders - deep knowledge and experience are required here in completing the orders. These orders should be not assigned to the employees with low expertise because the probability to complete the order in time is low. Instead, when the priority is low, it is recommended to assign them to an employee with medium expertise.

We suggest the [0,5 – 1,5] for the $I_{E/C}$ indicator. The lower limit was proposed based on the ratio of the lowest expertise and highest complexity recommended of the order: 1/2 = 0,5. If the ratio is smaller than this, the order should be interchanged with another employee or on hold.

The upper limit was established based on the following argument: the employee with the highest expertise (3) should have assigned an order at least equal with medium level (2), otherwise, a waste of resources is made.

There will be for sure no improving the knowledge in the team when an employee with high expertise receives an order with low complexity. For the orders with low complexity, the maximum level of expertise of an employee should be a maximum of two.

Figure 1 presents the concept of the algorithm used for assigning orders based on their complexity and the employee's expertise. And we introduced two terms: interchange and hold, that will be further defined:

Interchange of the orders means to change the orders assignment in direction of achieving higher efficiency. For example, instead of allocating a low complexity order (1) to a high expertise employee (3), we need to search for a lower qualified employee; if the only employee found is already involved in another request, than the orders should be interchanged between the employees for better productivity and progress of the team members.

Hold means that the order is not assigned to an employee, it is postponed until an employee with proper expertise is be available. This state can be applied when $I_{E/C}$ is outside the recommended interval [0,5 - 1,5]. For example, when an order with low priority (1) and high complexity (3) is given by the customer, and the only available employee is an employee with low expertise (1), the recommended action is to hold the process until an employee with medium (2) or high expertise (3) will be available.

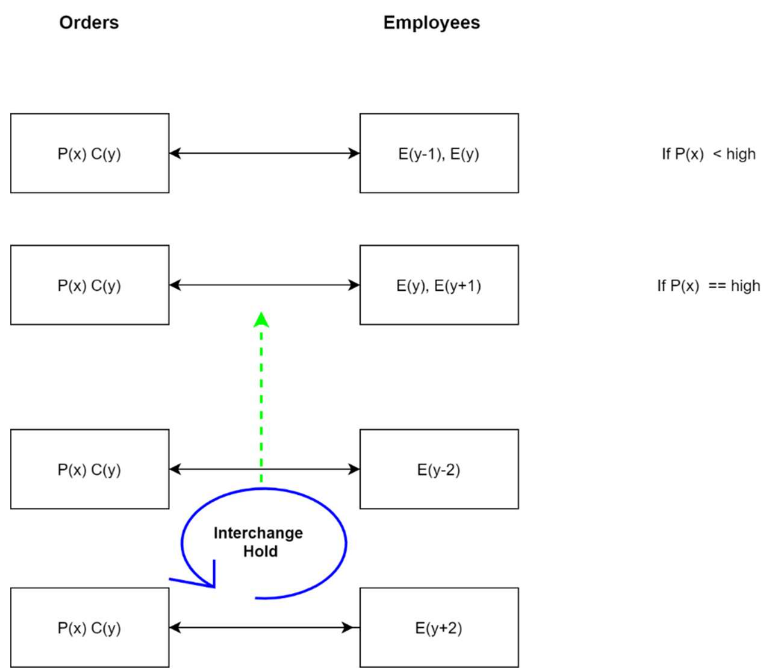

Fig. 1. The algorithm for assigning orders based on their complexity and the employee's expertise

And here is some scenario for $I_{E/C}$:

- If $I_{E/C} \leq 1$, and the priority of the order is medium or high, then the order should be assigned to an employee with higher expertise.

- If the priority of the order is low, is recommended to have $I_{E/C} < 1$, close to the low limit of 0,5. This means that the employee solving that order has a smaller expertise than the complexity of the order. In the area of low priority orders, the learning process should take place.

- If $I_{E/C} > 1{,}5$, it is a indication of waste. This can happen in the case that the expertise of the employee is higher than the complexity of the order.

General principles that can be applied:

1. Assign the order based on its complexity to an employee with the same level of expertise.
2. If the priority is high and an employee with the expertise equal with the order complexity is not available, assign the order to the employee with highest expertise available.
3. If the priority is low, an employee with a lower expertise should be found so the learning process takes place.

Next, the algorithm is described in detail for every possible situation and the actions that must be taken, to reach the declared objectives:

Function orders_arrangement (order_1, order_2, ..., order_n)
{The orders should be arranged descending based on their priority.

In the case that there are orders which have the same priority, the next criteria that must be taken in consideration is their complexity. If the complexity is higher and the orders have the same priority, the order with higher priority will be the order with higher complexity.

In the case that the orders have the same priority and same complexity, their order will remain the same.}

Function employees_arrangement(Employee_1, Employee_2,..., Employee_n)
{The employees will be arranged in a descending order based on their expertise. }

Function order_allocation (employees, orders)
{The order should be assigned based on its priority and complexity to an employee that has the necessary level of expertise to solve the order. For this, based on the complexity of the

order that must be allocated, an employee will be selected from the list.

If order priority == 3 and order complexity == 3 and expertise == 3 available then:
Assign order.
If order priority == 3 and order complexity == 3 and expertise == 2 available then:
Assign order.
If order priority == 3 and order complexity == 3 and expertise == 1 available then:
Check which report of all the employees assigned already with orders is the biggest. Take that $I_{E/C}$ and check if it is possible to interchange the orders between the employees. For example, if an order with priority high and complexity equal with 1, the order can be changed with an employee that has the expertise equal with 1 and the complexity of the order equal with 3 and priority equal with 1. Like this, $I_{E/C}$ is closer than the established limits [0,5; 1,5].
In the case that interchange is not possible, check if an employee with high or medium expertise will finish its order soon. If yes, put the order on hold.
In the last case scenario the order is assigned to the available employee with the mention that when another employee with higher expertise is free of his/her order, the order should be assigned to him.}
If order priority == 3 and order complexity == 2 and expertise == 2 available then:
Assign order.
If order priority == 3 and order complexity == 2 and expertise == 2 not available then:
Check if available employee with expertise == 3 then: Assign order. ***Reason:*** the priority of the order is high, and it must be finished as soon as possible.
If order priority == 3 and order complexity == 2 and expertise == 1 available then:
Assign order. ***Reason:*** since the priority is high and the complexity is medium, the best idea to try to complete the order with the available employee.
If *order priority == 3 and order complexity == 1 and expertise == 1 available then:*
Assign order.
If *order priority == 3 and order complexity == 1 and expertise == 2 available then:*
Assign the order.
If *order priority == 3 and order complexity == 1 and expertise == 3 available then:*
Assign the order.
If order priority == 2 and order complexity == 3 and expertise == 2 available then:
Assign order. ***Reason:*** since the priority is not high an employee with lower expertise than the order complexity will be assigned for learning process and check how the person can handle higher complexity orders.
If order priority == 2 and order complexity == 3 and expertise == 3 available then:
Assign order.
If order priority == 2 and order complexity == 3 and expertise == 1 available then:
The order is on hold. ***Reason:*** The difference between the employee expertise and the order complexity is outside of the imposed limits of the report and the employee has small chances to complete the order.
If order priority == 2 and order complexity == 2 and expertise == 1 available then:
Assign the order. ***Reason:*** Learning process. If now is not available an employee with expertise equal with one, but soon it will be, put the order on hold.
If order priority == 2 and order complexity == 2 and expertise == 2 available then:
If an employee with lower expertise will complete his/her assignments soon, put the order on hold until the employee is available, else assign order.
If order priority == 2 and order complexity == 2 and expertise == 3 available then:
If an employee with expertise equal with two will complete its assignments soon, put the order on hold until the employee is available, else assign order.
If order priority == 2 and order complexity == 1 and expertise == 1 available then:
Assign order.
If order priority == 2 and order complexity == 1 and expertise == 2 available then:
Check if an employee with expertise one will be available soon. If so, put the order on hold. If the employee will not be available soon, assign the order.

If order priority == 2 and order complexity == 1 and expertise == 3 available then:

Check first if an employee with expertise one will be available soon. If yes, put the order on hold. If no, check if an employee with expertise two will be available soon. If yes, put the order on hold.

If the above cases will not happen assign the order.

If order priority == 1 and order complexity == 1 and expertise == 1 available then:

Assign the order. If an employee with expertise one, will finish his/her task soon, put the order on hold and assign it to him when the current order is complete.

If order priority == 1 and order complexity == 1 and expertise == 2 available then:

If an employee with low expertise will be available soon, put the order on hold.

Else: Assign the order.

If order priority == 1 and order complexity == 1 and expertise == 3 available then:

Verify if an employee with expertise one or two will finish their current orders soon, if so, put the order on hold.

If an employee with a level of expertise one or two will not be available soon, interchange the orders, the employee with high expertise should take care of a task that has at least the complexity two, and an employee with expertise one or two should be assigned to this order.

Reason: efficient resources management.

If order priority == 1 and order complexity == 3 and expertise == 2 available then:

Assign order. ***Reason:*** learning process.

If soon an employee with expertise level of two will become available put the order on hold.

If order priority == 1 and order complexity == 3 and expertise == 3 available then:

First check if an employee with expertise level of two will become soon available, if yes put the order on hold. If not check if an interchange could be possible. If not assign order to the employee with the expertise high.

Reason: because the priority is low, now the learning process for the employees to increase their experience and expertise should take place.

If order priority == 1 and order complexity == 3 and expertise == 1 available then:

Order on hold. ***Reason:*** the employee with expertise equal with one will have difficulties to solve this kind of orders.

If order priority == 1 and order complexity == 2 and expertise == 1 available then:

Assign order. ***Reason:*** learning process.

If order priority == 1 and order complexity == 2 and expertise == 2 available then:

Check if an employee with expertise equal with one will finish his/her current order to have enough time to take this order. If it will happen like this, put the order on hold.

Else: Assign order.

If order priority == 1 and order complexity == 2 and expertise == 3 available then:

If an employee with lower expertise will complete its assignments soon, put the order on hold until the employee is available.

Else: Assign order.

If order priority == 1 and order complexity == 1 and expertise == 1 available then:

Assign order.

If order priority == 1 and order complexity == 1 and expertise == 2 available then:

If an employee with lower expertise will complete its assignments soon, put the order on hold until the employee is available.

Else: Assign order.

If order priority == 1 and order complexity == 1 and expertise == 3 available then:

If an employee with lower expertise will complete its assignments soon, put the order on hold until the employee is available.

If it is not the case, check if an employee with expertise two is working on and order with complexity two, he can change the orders with the available employee with expertise three.

Else: Assign order.

In the case that there is not an employee available with the level of expertise equal with the level of complexity of the order and the priority is low, then the level of expertise should be decreased with one level until it is assigned the order.

In the case that there is not an employee available with the level of expertise equal with the level of complexity of the order and the priority is high, then the level of expertise should be increased with one level until it is assigned the order.}

All cases and the necessary actions for each one of them are presented in Tabel 1.

Table 1

Possible scenarios and algorithm application for orders allocation

Order priority	**Order complexity**	**The highest available expertise**	**Assignation**	**Explanation**
High	High	High	High	
High	High	Medium	Medium	
High	High	Low	**1: Interchange 2: hold 3: assign**	Employee with expertise medium or high already assigned on order with low or medium complexity or low priority, change the order assignation
High	Medium	Medium	Medium	
High	Medium	High	High	
High	Medium	Low	Low	
High	Low	Low	Low	
High	Low	Medium	Medium	
High	Low	High	**1: Interchange 2: assign**	Change the orders with an employee that has an expertise two level, if not possible, assign the order.
Medium	High	Medium	Medium	
Medium	High	High	High	
Medium	High	Low	**1: Interchange 2: hold 3: assign**	If the priority is high and an employee with the expertise equal with the order complexity is not available, it must be found an employee with higher expertise.
Medium	Medium	Low	Low	
Medium	Medium	Medium	Medium	
Medium	Medium	High	**1: Interchange 2: hold 3: assign**	
Medium	Low	Low	Assign	
Medium	Low	Medium	On hold, interchange or assign	
Medium	Low	High	**1: Interchange 2: hold 3: assign**	
Low	High	Medium	Assign	
Low	High	Low	Hold, interchange or assign	
Low	High	High	**1: Interchange 2: hold 3: assign**	
Low	Medium	Low	Assign	
Low	Medium	Medium	On hold, assign	
Low	Medium	High	**1: Interchange 2: hold 3: assign**	
Low	Low	Low	Assign	
Low	Low	Medium	**1: Interchange 2: hold 3: assign**	
Low	Low	High	**1: Interchange 2: hold 3: assign**	

Table 2

Random order assignation

Employees and their expertise	Orders, their priority and complexity
E11 <--	O133
E21 <--	O232
E32 <--	O323
E43 <--	O422
E52 <--	O511

Table 2 shows how orders are assigned to different employees, following the basic rule: first available employee receives the first order received from headquarters. $E1_1$, with a low expertise receives the order number one with high priority and high complexity. As expectable, many corrections are necessary to fulfill quality requirements, and the deadline was extended with three days (beside the two days allocated for it), with high pressure on employee and not so many extra – knowledge added; the customer was not satisfied with the results and costs were higher than expected. In this case $I_{E/C}$:

$$I_{E1/C1} = \frac{1}{3} = 0{,}333, \qquad (2)$$

outside the recommended interval of [0,5; 1,5].

In the second case, employee has a lower-level expertise (ranked as 1) and receives an order with high priority and medium complexity. The employee does not have the necessary expertise to complete the order in time, the order had a deadline of five days, and was completed in eight working days.

$$I_{E2/C2} = 0{,}5, \qquad (3)$$

within the limits, but talking into account the high priority is high, the assignment should be made for higher expertise employee.

The third case, a medium level expertise employee (ranked as 2) receives an order with medium priority and high complexity. Here the deadline was four days, but the order was finished in five days. In this case:

$$I_{E3/C3} = 0{,}666 \qquad (4)$$

The result fits the proposed interval, but, if we consider the priority, the suggestion for the future is to assign in such cases higher expertise employee.

The fourth employee with high expertise receives an order with medium priority and medium complexity. The order deadline had three days and it was completed in one day and a half. In this case:

$$I_{E4/C4} = 1{,}5, \qquad (5)$$

we can talk about a waste of resources, and probably a medium expertise employee should take over this order.

In the last case presented, the result is:

$$I_{E5/C5} = 2 \qquad (6)$$

The expertise of the fifth employee is higher than the complexity of the order and the priority of the order is low. The deadline for this order was one day and it was completed in half a day. In this case an employee with low expertise should have the order number five, since there is sufficient time for completing the order.

The results of these study cases are presented in Figure 2: Expected deadlines and actual (real) deadline for each of the 5 order. We can have an image of the waste here: 7 extra days, not paid by the customer and, if we report this at the number of allocated days (11), we have a 63% extra time needed for the orders completion.

In Table 3 and Figure 3 we present the results after the algorithm was applied

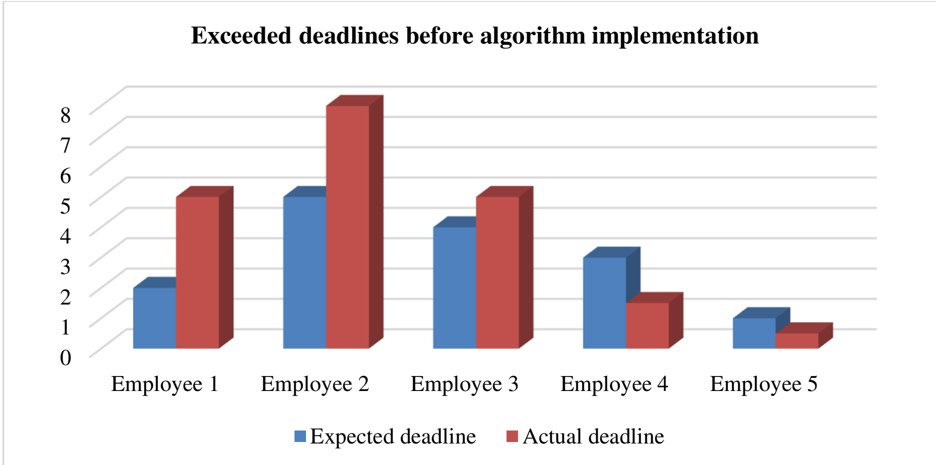

Fig. 2. Outcomes of the random assignation of the orders

Table 3

Orders assigned according with the algorithm.

Employees and their expertise	**Orders, their priority and complexity**
E11 <-----	O511
E21 <-----	O422
E32 <-----	O232
E43<-----	O133
E52<-----	O323

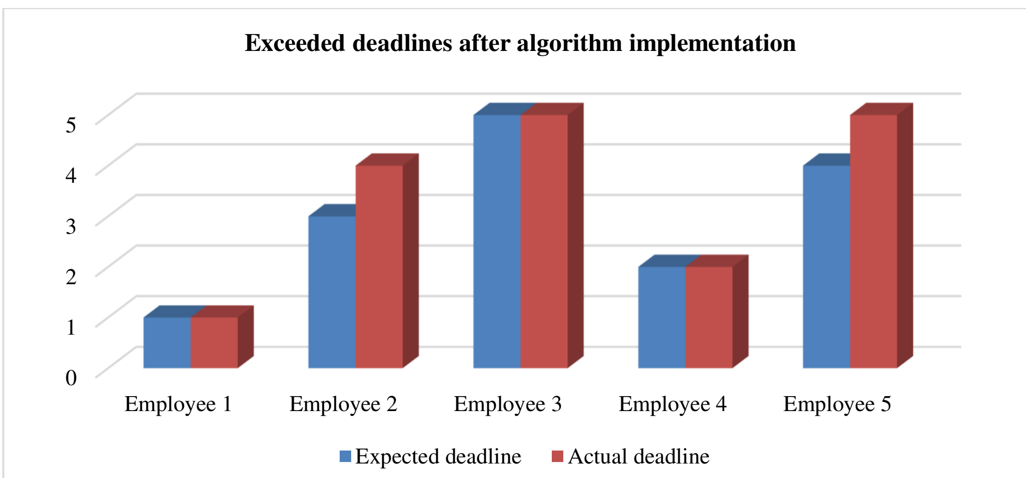

Fig. 3. Outcomes of the algorithm application

As it can be seen from the table, the number of employees who needed extra time for finishing their orders is lower, equal with two.

For the order 3 and five were given 7 days for finishing the orders, and two extra days were needed for completing the orders, representing extra 28,57%, (an improvement with 34% from the previous case). We estimate that in the future the improvement will be higher, based on the fact that the team is evolving and gather experience.

4. CONCLUSIONS

In offshore branches from automotive companies, employees with different levels of expertise and knowledge can be found in the newly created teams. The tasks coming from the headquarters are usually assigned to the team members without such criteria like complexity of the task, or the expertise of the employee required for that task, the orders being assigned based on their priority, to the employees available at that moment. The results can be, in many cases, exceeded deadlines, low quality of the software products, a high rework rate and a high pressure on employees. The algorithm for tasks allocation based on the order's priority, complexity and employee's availability and expertise was tested in a real life scenario and showed improvement especially in term of extra days needed for different orders. Using the algorithm, the companies could better value and improve the knowledge capital, challenging the employees to solve high qualified tasks and of course giving them reasonable deadlines and support.

5. REFERENCES

[1] Ruxandra Maria Bejinariu, *Conceptual framework of the organizational business processes*, Springer Fachmedien Wiesbaden, Print ISBN: 978-3-658-29388-8, Electronic ISBN: 978-3-658-29389-5, https://doi.org/10.1007/978-3-658-29389-5_2, 2020.

[2] Castelli, C., & Castellani, D. *The internationalisation of R&D: Sectoral and geographic patterns of cross-border investments*. Economia e Politica Industriale - Journal of industrial and business economics, 1, 127–143, 2013. doi:10.3280/POLI2013-001006, 2013

[3] Albertoni, F., & Elia, S., *The global sourcing of business services: Evidence from the offshoring research network survey*. Economia e Politica Industriale, 41(2), 131–146, 2014.

[4] Davide Castellani, Maria Luisa Mancusi, Grazia D. Santangelo, Antonello Zanfei, *Exploring the links between offshoring and innovation.* Economia e Politica Industriale, 42(1)1, 1–7, 2015. DOI: 10.1007/s40812-014-0008-8, 2015

[5] Fel Fabienne, Griette Eric. *An Analysis of the Off shoring Decision Process: The Influence of the Company's Size*, Procedia - Social and Behavioral Sciences, Volume 58, 596-605, https://ac.els-cdn.com/S1877042812044990/1-s2.0-S1877042812044990-main.pdf?_tid=a34cb648-e952-11e7-bc62-00000aab0f6b&acdnat=1514192859_4fdb7b5969afcc8f79396a316a801882, 2012

[6] Christian Kreiner, Richard Messnarz, Andreas Riel, Damjan Ekert, Michael Langgner, Dick Theisens, Michael Reiner, Automotive Knowledge Alliance AQUA – *Integrating Automotive SPICE, Six Sigma, and Functional Safety*. Publisher: Springer Berlin Heidelberg. DOI https://doi.org/10.1007/978-3-642-39179-8_30 ,2013

[7] Fabio Falcini, Giuseppe Lami, *Embracing Software Process Improvement in Automotive Through PISA Model*. Springer International Publishing, Print ISBN: 978-3-030-35332-2, Electronic ISBN: 978-3-030-35333-9; DOI https://doi.org/10.1007/978-3-030-35333-9_5 , 2019.

[8] Fajar Ramadhani Mahendrawathi ER, *A Conceptual Model for the Use of Social Software in Business Process Management and Knowledge Management*. Procedia Computer Science Volume 161, 2019, Pages 1131-1138. https://www.sciencedirect.com/science/article/pii/S1877050919319362 https://doi.org/10.1016/j.procs.2019.11.225 , 2020.

[9] Lewin, A. Y., Massini, S., & Peeters, C. *Why are companies offshoring innovation and quest; the emerging global race for talent*. Journal of International Business Studies, 40(6), 901–925. DOI:10.1057/jibs.2008.92 , 2009.

[10] Ana Belén Pelegrina, Kawtar Benghazi, María Visitación Hurtado, Manuel Noguera. *A Framework for the Semantic Representation of Business Processes within Business Organizational Models*, Print ISBN: 978-3-642-15722-6, Electronic ISBN: 978-3-642-15723-3, https://doi.org/10.1007/978-3-642-15723-3_6 , 2010

[11] Marius, G. A. L., & Kifor, C. V., *Offshoring for companies specialised in software for automotive*. Acta Technica Napocensis-Series: Applied Mathematics, Mechanics, and Engineering, 61(4), 2018.

Algoritm şi model pentru capitalizarea cunoştinţelor în filialele cu specific de Cercetare-Dezvoltare deschise de companiile din domeniul automotive

Rezumat: Când se deschide o filială nouă într-o ţară diferită de sediul central al companiei, nivelul cunoştinţelor şi al expertizei angajaţilor precum şi nivelul de înţelegere al proceselor interne ale companiei nu este egal între membri echipei, deci nu oricare membru al echipei poate să se ocupe de comenzi cu un nivel de complexitate ridicat sau mediu. Astfel, obiectivul acestei lucrări este de a ajuta echipele noi să evite alocarea aleatorie a comenzilor către membrii echipei. De asemenea, o formulă matematică este propusă cu scopul de a facilita alocarea comenzilor către membrii echipei pe baza unor criterii clare: prioritatea şi complexitatea comenzii precum şi expertiza şi disponibiltatea angajatului. În final avem un exemplu de două situaţii care arată rezultatul neaplicării şi al aplicării algoritmului propus.

Marius GAL, PhD student, Lucian Blaga University of Sibiu, Department of Engineering and Management, marius.gal@ulbsibiu.ro

Claudiu Vasile KIFOR, Prof.Univ.Dr.Eng., Lucian Blaga University of Sibiu, Department of Engineering and Management, claudiu.kifor@ulbsibiu.ro