

# An Approach To Project Team Formation Using Task Management System

V. Y. Sokol, M. O. Bilova\*, V. O. Krykun

National Technical University "Kharkiv Polytechnic Institute"

\*Corresponding author. E-mail: missalchem@gmail.com

Paper received 29.08.20; Accepted for publication 18.09.20.

<https://doi.org/10.31174/SEND-NT2020-238VIII29-06>

**Abstract.** The research focuses on the analysis of project team formation specifics in IT companies. It overviews main characteristics of task management systems (TMS) and identifies the place of project team formation process during project planning. The article addresses issues of creating project schedule taking into account project team members availability. It proposes an algorithm of members selection, including adaptation to the problem of determining possible performers of tasks.

**Keywords:** *task management system, software development team, team formation, IT company.*

**Introduction.** One of the main approaches in a modern enterprise is to improve development of production technologies and that may be achieved through digitalization of various business processes [1]. Especially in IT field we witness significant increase in the complexity of projects and their coordination. Considering the internationalization trend of project work in IT companies, managing the effective process of remote employees is becoming increasingly difficult. Thus, a need for management systems that can improve the process of project planning and performance becomes more significant. Such systems are implemented in various project management systems and, in particular, task management systems.

Along with the increasing of projects internationalization, the complexity of the work of managers who need to control large teams increases. Systems require a large number of tasks to be performed. Determining the team members is a complex task, which defines the effectiveness of a project. That is why staff selection becomes an important process in project planning that requires automation to increase work effectiveness.

**The purpose of the work** is to analyze the process of forming a project team in task management systems to determine the mathematical formulation of the problem and the algorithm for its solution, which will provide a basis for developing a software component to automate the process of selecting employees to work on the project.

**A brief overview of publications on the topic.** Project management can be defined as the usage of knowledge, skills, and tools to work on a project in order to achieve the project requirements [2].

A project management system is a software that has the ability to help strategize, organize, and manage resource streams and develop resource approximations. Depending on the complexity of the software, resource breakdown structures, resource availability, resource rates and various resource calendars can be defined to assist in optimizing resource utilization. [3]. At the same time, it is possible to distinguish some project management systems, that are primarily related to the concept of the task in the project. A task is an element that displays information about a problem that needs to be solved over a period of time. The task can be described by priority, status, due time, task owner, assignee [4].

For effective task management, it is common to use task management systems, that are usually an integral part of project management systems. The goal of such systems is to increase the efficiency of creating and monitoring the

process of performing tasks and to control work on everything necessary for project success. Given the fact that quite often the workers responsible for different parts of the project are far away from each other, such systems are a necessity of modern management, in particular in the field of information technology. An example of this is a situation where a business analyst has the necessary information on issues in the subject area, without which the task becomes impossible to be done. In this case, the task management system is an effective tool to obtain all the answers needed by the worker.

At the same time, human resource allocation plays a critical role in delivering software projects since people are directly involved in all software development activities. Therefore, forming an appropriate project team is an essential activity for the project implementation that remains challenging for most of the IT-companies [5]. Studies have presented relevant attributes to form a software development team, from which the most common are technical abilities, availability, project importance, individual cost, productivity, behavior, and personality [6]. The amount of attributes makes the process of team formation hard and time-consuming for managers, which requires new approaches. Automation of this process can help not only solve the discussed problems but also reduce the amount of subjectivity. Thus, it is important to give the mathematical description of the problem of team formation to identify ways of its further automation in task management systems.

**Materials and methods.** A large and very important place in the process of working on the project is the transition from the global goal of the project to the selected elementary tasks. This process determines how well the work on the project will be planned and, as a consequence, the result of this work. If the elementary tasks are well identified, the project management process is improved. In this case, it is easier to distinguish a clear organization of responsibility for one or another part of the work.

The result of such a decomposition process can be described and formalized by a work breakdown structure (WBS). It is a set of work elements or components, hierarchically organized, which together describe a project and which must be performed by a team to achieve the goal of the whole project, as well as to implement the delivery of these elements to the customer [2, 7]. Components can be organized at some levels, they are usually depicted in the form of a graph of the hierarchy. They must be independent

to be able to work on them in parallel and to appoint a responsible person who will perform a certain amount of work with clearly defined criteria. This allows to qualitatively plan, track progress in working on a particular element, assess its complexity, determine the time and material costs required for it. As a result, the whole project management process becomes more flexible and controlled [7].

The WBS approach is very useful in the context of implementing the transition from a comprehensive document with a full set of requirements to individual elements of work, which can be detailed into elementary sets of tasks

to be performed in a single time period by a competent person. Such small tasks will be the basis for further entries in the task management system.

After the WBS was created, a schedule was assigned to all tasks, then the entire set of tasks is defined. When each task in this set has been described, all tasks can be systematized and presented in the form of a comparative table, which contains all the necessary information to solve the problem, namely to form a team of developers to implement the project. Information about the executors of the tasks is not provided, because it is created to identify them.

**Table 1** An example of a comparative table of tasks to problem formulation

| ID  | Complexity | Date of start | Deadline   | Specialty | Level  | Skills                 |
|-----|------------|---------------|------------|-----------|--------|------------------------|
| P-1 | 10h        | 20.07.2020    | 21.07.2020 | Developer | Junior | Java SE, Spring        |
| P-2 | 10h        | 20.07.2020    | 21.07.2020 | BA        | Middle | Domain                 |
| P-3 | 12h        | 21.07.2020    | 22.07.2020 | Developer | Junior | Java SE                |
| P-4 | 10h        | 21.07.2020    | 22.07.2020 | QA        | Junior | Selenium, Manual       |
| P-5 | 8h         | 22.07.2020    | 22.07.2020 | Developer | Junior | Java SE, Spring        |
| P-6 | 14h        | 23.07.2020    | 24.07.2020 | QA        | Middle | Selenium, Auto-testing |

Once the project schedule has been determined, it is necessary to solve the problem of implementing a set of tasks for which their place in the schedule has already been determined. In this case, the schedule can be determined without specifying additional conditions for available resources. The next task is to select the team that will work on the project. Even if the available resources are not sufficient to meet the needs of the project, it is important to allocate the available resources to identify team members for as many tasks as possible. Therefore, the statement of the problem, which was adapted for the task of finding the maximum number of workers to perform the tasks, can be described as follows:

- set of tasks  $N = \{1, \dots, n\}$  is given;
- set of employees  $K = \{1, \dots, k\}$  is given;
- at any given time (for example, on working days) there is an available number  $Q_k$  of  $k$  worker's resource units (for example, in working hours);
- the values  $p_i \geq 0$  that describe the required duration of the task  $i$  ( $i = 1, \dots, n$ ) are given;
- during the implementation the amount of  $q_i \leq Q_k$  resource units of  $k$  ( $q_i = p_i$ ) worker is needed, if this worker was identified as the executor of this task  $i$  ( $\varphi_{ik}(t) = 1$ ) (the worker after the completion of the task is completely free to perform other tasks);
- the values  $z_{ik}$  are determined, that define whether the worker  $k$  has the necessary skills to perform the task  $i$ ;  $z_{ik} = 1$  if the sets of skills, specialties and level required to perform the task are subsets of the worker's  $k$  skills, specialties and level;
- the values  $\varphi_{ik}(t)$  are given;  $\varphi_{ik}(t) = 1$ , if worker  $k$  was identified as the executor of the task  $i$ , performed at a time  $t$ , and  $\varphi_{ik}(t) = 0$  in other case.

The goal is to maximize the  $L$  value (number of tasks assigned to performers):

$$L = \sum_{i=1}^n \varphi_{ik}(t).$$

This requires that several restrictions should be met, namely:

1) the task must be provided with the necessary amount of resources for its implementation: at any time  $t \in [0, C_{\max})$  ( $C_{\max}$  – the day of completion of the last task of the project, the deadline of the project) restrictions must be observed:

$$\sum_{i=1}^n \varphi_{ik}(t) \cdot q_i \leq Q_k(t).$$

2) for each task, the worker must have all the necessary skills to perform it, i.e. the relation of implication must be fulfilled:

$$\varphi_{ik}(t) = 1 \rightarrow z_{ik} = 1.$$

To determine the maximum number of workers for tasks, we can propose the next criterion of the priority. The most foreground task to select is the one with the least number of workers able to perform this task (i.e., for which all limitations given in the problem statement are fulfill).

For instance, the set of tasks  $N$  consists of two elements, and the set of workers  $K$  consists of three elements. If the first task under these restrictions can be performed by only one worker, and the other task can be performed by all three workers, then the value  $L$  will be maximum if we start with the first task for which there are the least number of capable workers  $L = 2$ . However, the maximum value is not guaranteed if the criterion mentioned above is not met (if a person who can perform the first task is appointed to perform the second, after which it becomes possible to violate the restrictions for the first task, i.e.  $L = 1$ ).

The idea of the algorithm for solving the problem of a team formation is to appoint a person who can perform task and at the same time, this person can solve the least number of other project tasks for which no executors have yet been identified.

Several auxiliary associative arrays are proposed for the implementation of the algorithm. The first of them represents a set of tasks  $x_i \in X, i = \{1, \dots, n\}$ , where for every  $x_i$  exist the set of appropriate workers. That is, those who

have time to complete the task  $x_i$  and who can perform this task on the basis of available skills, specialization and level. The second array represents the set  $y_j \in Y, j = \{1, \dots, k\}$ , where the corresponding value for a person  $y_j$  is the set of tasks that can be performed by this person. Also at the stage of determining the executor for the task, an auxiliary associative array  $O$  can be used, which contains  $o_i \in x_i, i = \{1, \dots, n\}$  the correspondence of the worker and the set of tasks that this worker can perform, taking into account the time required to complete the task  $x_i$ .

An algorithm is suggested to solve the problem of a project team selection in the form of a diagram of activities (Fig. 1).

**Conclusions.** The article outlines some specific issues of software development projects in IT companies. It identifies how project management system and task management systems may both benefit from process of successful project implementation. Main approaches of project team formation are analyzed, we presented the mathematical model and algorithm how that issue may be solved is suggested.

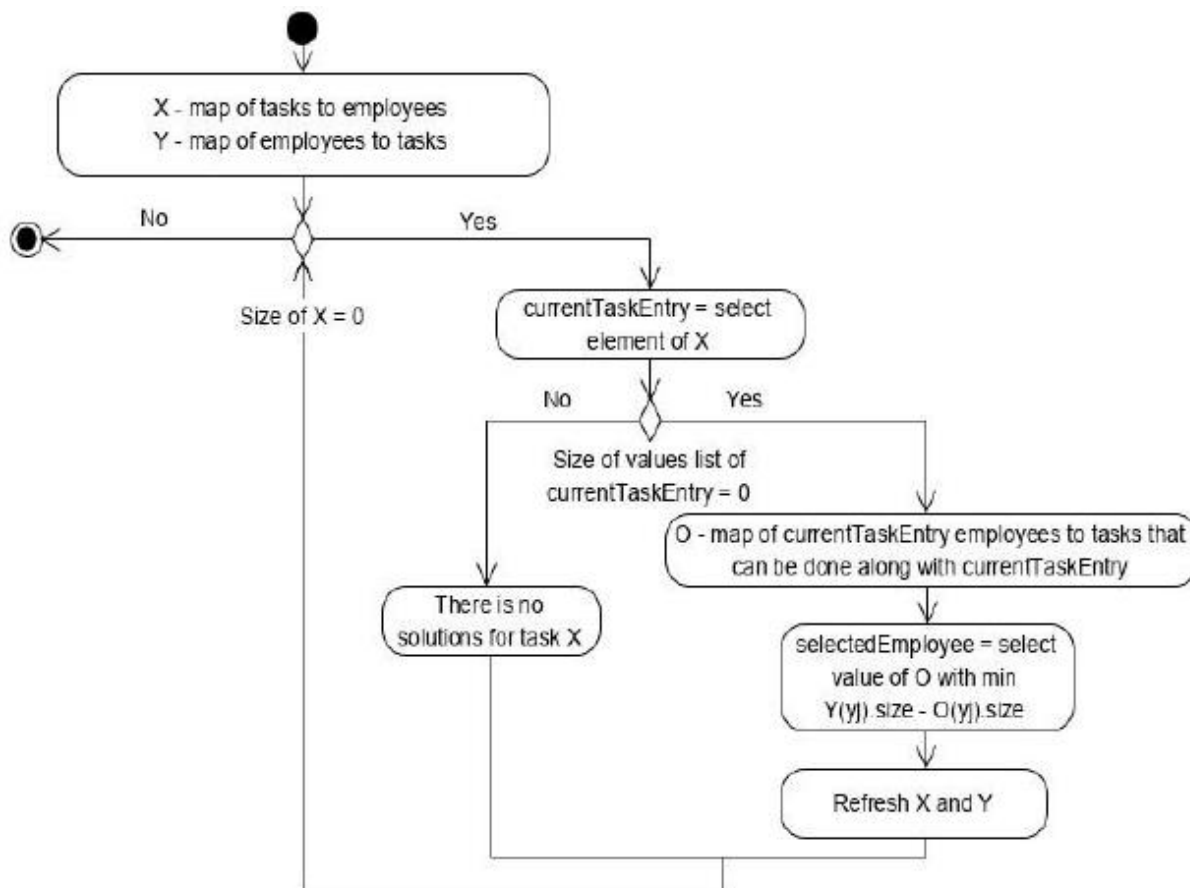


Figure 1. Employee selection for the project team formation

REFERENCES

1. Sokol V. Y., Bronin S. V., Karnaukh V. E., Bilova M. O. Developing Adaptive Learning Management Application for Project Team in IT-Industry. Bulletin of the National Technical University «KhPI». Series: System analysis, control and information technology, 2020, Vol. №1 (3), p. 97–105.
2. A Guide to the Project Management Body of Knowledge (PMBOK Guide): Fifth Edition/ Project Management Institute. Project Management Institute; Fifth Edition, 2013, 589 p.
3. Izang A. A., Ihesiulo C. C., Ofuru M., Okafor C. A Web-Based Project Management System. International Journal of Advanced Research in Computer Science and Software Engineering, 2016, Vol. 6(4), p. 39–45.
4. Duan Z., Li W., Cai Z., Distributed auctions for task assignment and scheduling in mobile crowdsensing systems, The 37th IEEE International Conference on Distributed Computing Systems (ICDCS), 2017.
5. Costa A., Ramos F., Perkusich M., Dantas E., Dilorenzo E., Chagas F., Meireles A., Albuquerque D., Silva L., Almeida H., Perkusich A. Team Formation in Software Engineering: A Systematic Mapping Study. IEEE Access, 2020, Vol. 8, P. 145687-145712.
6. da Silva F. Q. B., Franca A. C. C., Gouveia T. B., Monteiro C. V. F., Cardozo E. S. F., Suassuna M. An empirical study on the use of team building criteria in software projects. : Proceeding of the International Symposium on Empirical Software Engineering and Measurement (ESEM), 2011, pp. 58–67.
7. T. Rajani Devi, V. Shobha Reddy Work Breakdown structure of the project. International Journal of Engineering Research and Applications, 2012, Vol. 2, p. 683–686.