DEVELOPMENT AND TESTING OF ALGORITHM FOR AUTOMATING TEAM FORMATION IN IT PROJECTS

Roman Zolotukha
Postgraduate Student, National University of Life
and Environmental Sciences of Ukraine, Ukraine
e-mail: remko740@gmail.com, orcid.org/0000-0003-3099-722X

Olena Glazunova
Doctor of Pedagogical Sciences, Professor,
Dean of the Faculty of Information Technologies,
National University of Life and Environmental Sciences of Ukraine, Ukraine
e-mail: o-glazunova@nubip.edu.ua, orcid.org/0000-0002-0136-4936

Summary

The research aims to investigate the automation of recruiting and the formation of adaptive teams for IT projects using a proposed mathematical model. Additionally, the proposed algorithm needs to be tested in practice. The algorithm is based on weighting coefficients formed on the basis of expert assessments of key characteristics of candidates which are obtained by testing candidates for soft and hard skills. The research compares the results of two approaches: arbitrary team formation and a method based on a proposed mathematical algorithm. The authors conducted a practical study of both approaches during a hackathon whose task was to develop a web page for a future HR platform. The research methodology involved dividing the participants into two groups. The participants are divided into two groups. The first group forms teams independently during the hackathon, while the second group undergoes soft and hard skills tests during registration to determine team formation using a mathematical algorithm. The study's authors present the effectiveness results of both groups and describe the advantages and possibilities for improving the algorithm. The conclusions drawn identify the achievements of the study and prospects for further development of the algorithm for automating team selection in the IT industry. This work contributes significantly to the understanding of effective strategies for forming IT teams, which is crucial in the face of constant technological changes.

Key words: team development, IT, mathematical models, recruiting, multifactor coefficients, management.

DOI https://doi.org/10.23856/6125

1. Introduction

In the contemporary realm of information technology, where competition in the IT market and the challenge of innovations are increasingly prominent, the judicious selection and formation of IT teams become paramount for the flourishing of IT enterprises. The intricacies of employment in the information technology sphere are characterized not solely by the technical qualifications of the candidate but also by the employee's ability to collaborate effectively in a team-based environment. This work is devoted to examining a novel approach to the constitution of IT teams.

It is also important to note the dynamics of the labor market in the IT market *(opendatabot.ua, 2022)*, which has changed significantly after the full-scale invasion.(Fig. 1)

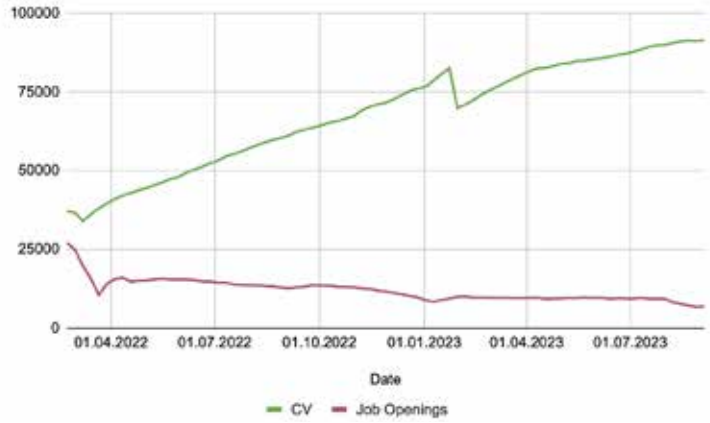

Fig. 1. Vacancies and resumes of IT specialists

Reducing the number of vacancies and increasing the number of candidates increases the competitive situation in the IT labor market, which directly affects the process of finding candidates in companies. Automating the process of recruiting candidates for IT teams can significantly improve its efficiency. It can reduce the time and costs of selection, as well as improve the quality of future personnel.

Solving the problem of selecting a balanced team is an urgent and important task for increasing the efficiency of IT projects. The task of creating effective working teams is necessary not only for project implementation, but also for introducing innovations and ensuring stable competitiveness of companies in a dynamic IT environment. However, common methods of candidate selection and team building often do not take into account the complexity of the requirements for candidates, and also significantly affect the time it takes to hire a candidate and close a position in the team.

The problem of forming teams for project implementation, as well as personnel management, is the subject of many domestic and foreign works. In particular, in research *(O. M. Krivoruchko and T. O. Vodolazhska, 2023)*, the authors consider the methods of developing personnel management strategies existing in the scientific literature, and also propose methodological provisions for developing a general personnel management strategy.

In the article *(I. Ivchenko, L. Lingur and T. Filatova, 2021)*, the authors investigated and proposed a prototype of a mathematical model of human resources management, taking into account the specifics of IT companies, with the objective function of the model aimed at optimizing the time spent by HR managers on recruiting personnel for IT teams. The authors of the article [4] studied the problem of planning, management and optimal use of available human resources in the production process using mathematical modeling. It is important to note that this article takes into account the factor of the COVID 19 pandemic and the problem of reducing the number of employees, which is now also relevant for the IT market of Ukraine in the context of the full-scale aggression of the Russian Federation and the stagnation of the IT market in Ukraine. The application of the model proposed by the authors has been studied in practice and the effectiveness of mathematical modeling and human resource management has been proven.

In the modern scientific literature *(A. Kostić, B. Maric, M. Kostura, and V. Timotić, 2021, D. Kabachenko and A. Lutsenko, 2015, L. Balabanov and O. Sardak, 2021, V. Nikiforenko, 2014)*, a combination of mathematical models and intuition is often used to find the optimal balance in team management.

Main goal of this research is to implement and test the results of applying a mathematical algorithm for forming IT teams and comparing it with the traditional arbitrary method of forming a working team.

2. Mathematical model apparatus for automating the formation of IT teams

The specifics of recruiting candidates for an IT project are the diversity of skills required for its successful implementation. Thus, the main condition for the successful acceptance of a candidate's application for a vacant position is the maximum compliance of his or her proposal with the project requirements. In our study, to develop a mathematical algorithm for selecting candidates for an IT project, we will take into account the following factors: the candidate's field of activity, the candidate's level of mastery of the main (hard) skills, the candidate's personality type, the candidate's remuneration, the candidate's level of English proficiency, and the candidate's personality type.

The set of candidate characteristics can be represented as follows:

$$SCC = \Sigma (X_1, ..., X_5) \tag{1}$$

where SCC – is the total set of candidate characteristics;
$X_1, ..., X_5$ – is candidate characteristics;
Also, let's represent the project requirements as Yn:

$$SPC = \Sigma (Y_1, ..., Y_5) \tag{2}$$

where SPC – is the total set of project requirements
$Y_1, ..., Y_5$ – is project requirements;

The requirements for the project that the HR manager operates with are the baseline that the candidate's set of characteristics must meet. Based on this, we will have three options for comparison results.

$$SCC/SPC < 1 \tag{3}$$

where the candidate's characteristics are lower than those specified in the project requirements.

$$SCC/SPC = 1 \tag{4}$$

where the candidate's characteristics match the defined project characteristics.

$$SCC/SPC > 1 \tag{5}$$

where the candidate's characteristics are higher than the specified project characteristics.

Under such conditions, the optimal scenarios for us are 2 and 3, since the candidate meets the requirements provided by the project manager. In scenario 1, the candidate is eliminated by one of the Xn factors, so he or she will not have the option to proceed to the second stage of team formation.

In order to select the right candidates, we need to categorize them by their specialty. Let the specialty be denoted as j.

Now we need to select candidates who fit the project conditions for each factor:

$$CCR_j = \frac{X_n}{Y_n} \tag{6}$$

where X_n – is the candidate's factor score in points;

Y_n – is the score of the minimum threshold provided by the project manager or hr.

It is also important to introduce weighting coefficients based on expert opinion to determine the most effective candidates, we denote these coefficients as k.

Taking into account the addition of the basic coefficients, we can calculate the requirements for candidates provided by the project manager (hr manager) using the following formula:

$$IPR = \Sigma\,(Y_1 \times k, \ldots, Y_5 \times k) \tag{7}$$

where Y_n – estimation of the basic factor from the project requirements;

k – factor weighting.

Now we can move on to the second iteration of the algorithm – team composition, and here we will use the principle of maximizing team efficiency using the points obtained after adding weighting factors for each candidate. The objective function here will be to maximize the total points received by the candidates in each area:

$$R_j = \frac{ICR}{IPR} \cdot \rightarrow max \tag{8}$$

where R_j – s the maximum score in the context of the professional area of activity;

ICR – he sum of the candidate's points;

IPR – the sum of the points of the minimum requirements of the project documentation.

3. Development of a prototype algorithm for further testing

After analyzing the requirements for this algorithm, we chose the PyQt5 library in the Python programming language. PyQt5 is a set of Python bindings for the popular Qt framework used to create GUI applications. *(M. Summerfield, 2015)* This solution is suitable for an HR manager to be able to interact with the application interface on a personal computer. Also, this solution will enable the ability to manually add and remove user characteristics and weighting factors.

The following functions were implemented to implement the calculation logic (Fig 2, 3):

- set_candidate_data – which sets the candidate's data for calculating the total score with the ability to dynamically add or remove characteristics. We also added a check that the score value should be in the range from 0 to 100.
- set_weights – configures weighting factors for various parameters of candidate evaluation. Added an additional check that the coefficients are set in the range from 0 to 1.
- calculate_score – a function that is responsible for the overall calculation of the candidate's rating. There is an additional check to ensure that the sum of the weighting coefficient is in the range from 0 to 1.

The application must be able to receive data from external resources. To do this, it must accept values in JSON *(JSON.org , 1999)* format and save the result of the calculation in this format.

- calculated_best_score – a function to create a team combination that will have the highest possible total score.

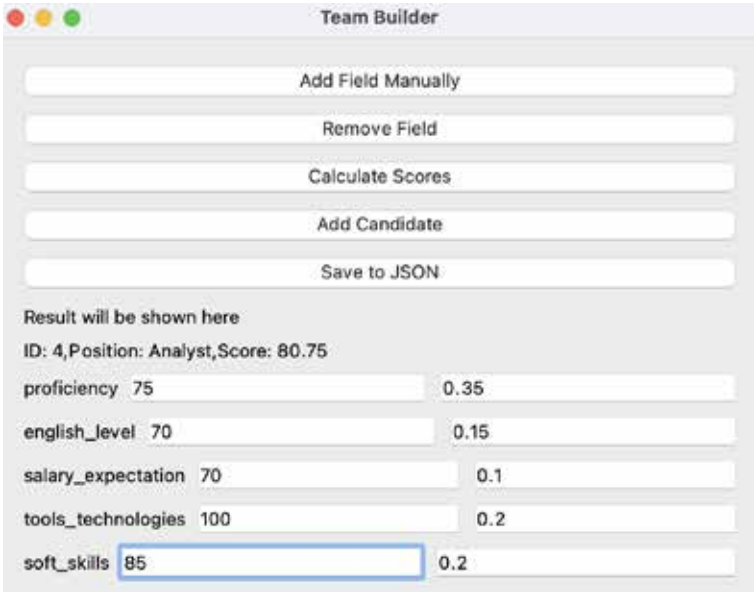

Fig. 2. The stage of adding candidate characteristics

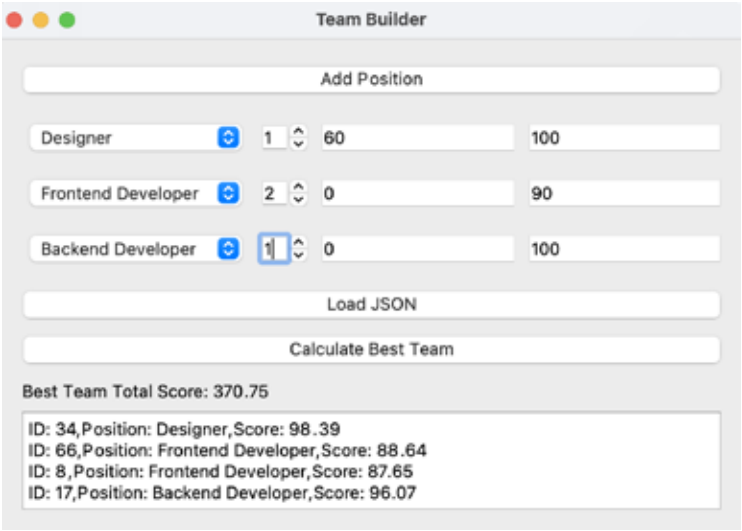

Fig. 3. The result of the automated command generation module

The implementation of the module includes two key iterations: selecting candidates from a common database according to project requirements and forming optimal teams. During the development of the module, functions were implemented to calculate the rating of candidates and set weighting factors, which allows for effective evaluation of candidates and takes into account various aspects of their profile.

4. Organization of the experiment and description of the research sample

An empirical research on the use of the developed modules to automate the process of forming IT teams was conducted on the basis of the annual IT hackathon "IT Marathon Online". The sample of subjects was composed of hackathon participants, as most of the registered participants are IT professionals who either already work in the industry or are starting their careers.

The goal of the hackathon was to create a web application for an HR platform in one day. When preparing for the hackathon to develop a web page for the HR platform, all participants were asked to fill out an additional questionnaire during registration in which candidates indicated their levels of proficiency in the following characteristics:

- *X1* – level of professional competencies;
- *X2* – English language proficiency;
- *X3* – salary expectations;
- *X4* – knowledge of Git, Jira, Figma tools and technologies.

They were also offered to take tests for factor *X5* – the level of soft skills formation.

All candidate characteristics were converted to a scale from 1 to 100 points depending on the level of candidate mastery of each factor. Also, a weighting factor (k) was added to each factor:

- *X: k* = 0.35
- *X2: k* = 0.15
- *X3: k* = 0.1
- *X4: k* = 0.2
- *X5: k* = 0.2

To implement the project, a team consisting of Front-End, Back-End developers, a designer, and a project manager/leader had to create a web page according to the specified requirements in 1 day.

The experiment involved 48 candidates, most of whom were aged 20–25 (Fig. 4).

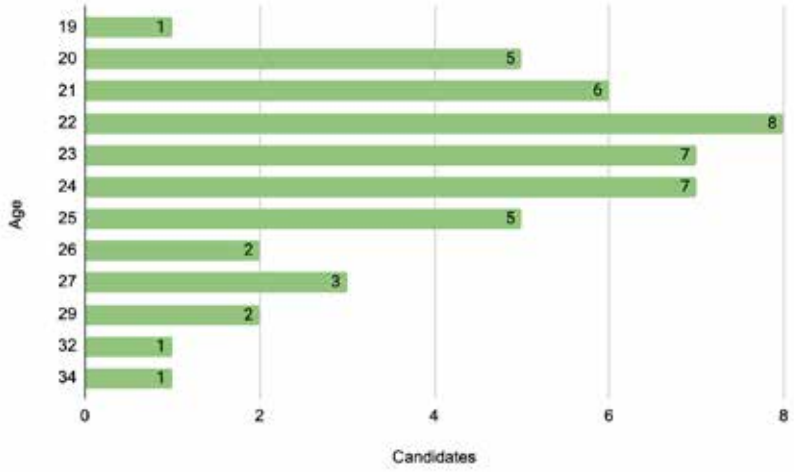

Fig. 4. Breakdown of participants by age

By gender, 85% are men and 15% are women. Or 41 men and 7 women (Fig. 5).

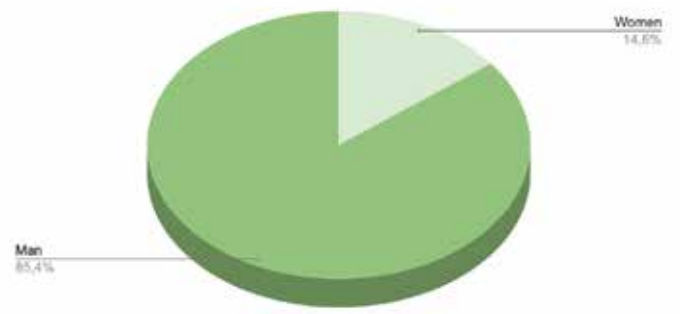

Fig. 5. Breakdown of participants by gender

After receiving the results of the survey using a randomizer, we selected 50% of the hackathon participants to be formed into teams to complete the task using the algorithm we implemented to automate the formation of teams based on our proposed mathematical model. The other 50% of the participants were allowed to form a team on their own, and the remaining participants were assigned to teams blindly using a randomizer, taking into account the condition that each team should have 1 Back-End and 1 Front-End developer, designer, and project manager.

At the end of the hackathon, the team's work was evaluated according to the following parameters:

- timely submission of the project – 100 points, each hour of delay reduces the maximum value by 10 points.
- technical compliance with the requirements – 100 points, the absence of one of the requirements reduces the number by 10 points.
- web page design – 100 points, inaccuracies in the design, lack of necessary elements reduces the value by 10 points.
- assessment of candidates in team interaction – the average score of each team member in terms of interaction with each other from 10 to 100 points.

Below are the total team scores. Of these, teams 2, 3, 4, 5, 7 were formed using the proposed algorithm. (Fig. 6)

Fig. 6. Overall score of teams based on their performance

5. Conclusions

This study demonstrated the importance of automating the process of team formation for IT projects and showed the effectiveness of the developed mathematical algorithm for the formation of IT teams. Teams 3, 4, 5,7 formed use of the mathematical algorithm, showed the highest results in all the parameters evaluated. The research showed, however, that some teams that were formed using the traditional method of getting to know each other at the beginning of the event also scored quite high. Therefore, further development of the potential of the algorithm for automating the selection of candidates is in order:

- Further improvement of the algorithm based on the results obtained to ensure more accurate selection and formation of teams.
- Expanding the algorithm's capabilities to take into account additional parameters and characteristics of candidates.
- Adding verified soft skills tests.
- Adding the ability to take tests to determine the level of all characteristics.
- Testing the algorithm in real conditions of hiring candidates for an IT company.

This research opens great prospects for further research in the field of selecting IT candidates and building effective teams. The results allow for a deeper understanding of the importance of objective and comprehensive criteria when building teams in the field of information technology.

References

1. Nevdalyi veresen ta skorochennya v IT: dynamika eksportu poslug pid chas viyny [Unsuccessful September and cuts in IT: dynamics of service exports during the war.] (2022). opendatabot.ua. https://opendatabot.ua/analytics/itexport-sept [in Ukrainian]

2. Kryvoruchko, O., & Vodolazhska, T. (2016). Rozrobka strategiy upravlinnya personalom metodom kontseptualnogo abstraguvannya [Development of personnel management strategies by the method of conceptual abstraction]. Ekonomika transport. kompleksu, (28), 69–83. http://nbuv.gov.ua/UJRN/ektk_2016_28_8 [in Ukrainian]

3. Ivchenko, I., Lingur, L., & Filatova, T. (2021). Modelyuvannya upravlinnya kadramy na IT-rynku pratsi [Modeling human resources management in the IT labor market]. Visn. Kharkiv. nats. un-tu im. V. N. Karazina seriya «Ekonom.», (101), 101–112 [in Ukrainian]

4. A., Marich, B., Kostura, M., & Timotich, V. (2021). Matematychna model dlya planuvannya lyudskykh resursiv u vyrobnychomu protsesi [Mathematical model for planning human resources in the production process]. U Mizhnar. symp. DAAAM (s. 4–9). [in Ukrainian]

5. Kabachenko, D., & Lutsenko, A. (2015). Vybor i realizatsiya strategii upravleniya personalom predpriyatiya. [Selection and implementation of the enterprise's HR management strategy] Nauk. visn. Kherson. derzh. un-tu Seriya «Ekon. nauki», (15), 58–62. http://ej.kherson.ua/journal/economic_15/1/16.pdf [in Ukrainian]

6. Balabanov, L., & Sardak, O. (2011). Upravlinnya personalom: pidruchnyk [Personnel management : A handbook]. Tsentr uchb. lit. [in Ukrainian]

7. Nikyforenko, V. (2014). Stratehichne upravlinnya lyudskymy resursamy: Navchalno-metodychnyy posibnyk dlya samostiynoho vyvchennya dystsypliny [Strategic management of human resources: Study guide for self-study of the discipline]. Atlant. [in Ukrainian]

8. Summerfield, M. (2015). Rapid GUI Programming with Python and Qt: The Definitive Guide to PyQt Programming. Prentice Hall.

9. Introducing JSON. (1999). www.json.org. https://www.json.org/json-en.html