What do software team managers want from a skills identification?

Matheus A. Flauzino
Federal University of Lavras
Lavras-MG, Brazil
matheus.flauzino2@estudante.ufla.br

Vinicius H. S. Durelli
Federal University of São João del Rei
São João del Rei-MG, Brazil
durelli@ufsj.edu.br

Mauricio R. D. A. Souza
Federal University of Lavras
Lavras-MG, Brazil
mauricio.ronny@ufla.br

Rafael S. Durelli
Federal University of Lavras
Lavras-MG, Brazil
rafael.durelli@ufla.br

ABSTRACT

Searching for developers with the necessary skills to work effectively on a project is not an easy task for companies. For software team managers, it is important to know their team's experience and what technologies they are using in order to be able to allocate resources optimally. Therefore, in this paper, we decided to study the needs of the managers of a software development company in order to identify their team's skills. We carried out a focus group. We have interviewed 11 managers from a development company (with approximately 75 developers). As result we have mapped 5 issues, 4 possible applications and another 7 needs to create a new methodology for identifying skills. Finally, we conclude that managers are interested in a methodology to identify skills in business flow.

KEYWORDS

skill identification, development team, team management

1 INTRODUCTION

Due to the high demands and complexity, the software developer has gone from being a simple professional who transcribes code to someone who has to deal with a complex work environment and with diverse skills [31]. Finding developers with these skills is not simple for recruiters [17].

Professional skills can be divided into two categories: hard skills and soft skills [25]. Hard skills are technical skills and specific knowledge of a professional. This skill is learned and developed through courses, book reading, study, observation and training. Soft skills are communication skills, leadership, self-control, confidence, persistence, and others [2, 5].

There are several techniques for tracking the experience of software developers [9, 23, 27]. For a Software Team Manager, it is important to know the experience of each member of your team and what technologies they are using. Therefore, it is possible to allocate development resources optimally and with as much economy as possible [10, 26].

One of the ways to evaluate a developer's experience is to analyze their contributions in Version Control System (VCS) repositories [1, 7, 18]. In addition to storing code, VCS provides a rich source of information about the developers and the project they are working on. Therefore, we can track the activities of those developers, and consecutively measure experience and skills, simply by looking at the contributions that developer makes to the project [14, 18, 30]. Usually, during the development process the manager needs to perform adjustment to accommodate the team's requirement. Thus, we believe that it is important to investigate a way to identify skills using VSC, especially when working with heterogeneous/multidisciplinary teams and usually with different cultures.

Therefore, we have performed a Focus Group session with team managers in a software development organization to investigate the needs related to the identification of developer skills in the context of multidisciplinary teams. Focus Group is a qualitative research method that consists of carefully planned discussions to collect perceptions of a group of people about a defined area of interest [15]. The method allows the exploration of the actors' perspective in greater depth and enables a concise understanding of their behavior and the real-life dilemmas of individuals and the group as a whole [8]. We have interviewed 11 managers from a development company (with approximately 75 developers). The results were promising and everyone agrees that an approach to assist in skill identification would greatly help the evolution of a development team. The contributions of this work are threefold:

(1) Investigation of a data-driven approach to identify the needs of managers regarding skills identification in their teams;

(2) We pave the way to other researchers to create approaches for identifying skills based on the needs of software team managers;

(3) Investigation of a data-driven approach to assist HR managers in order to understand the relation between humans and digitalisation.

Paper Structure. Section 2 presents the description and settings of the experiment. The section 3 presents the survey results. Section 4 presents the related works. Section 5 describes the concluding remarks and future work.

2 SURVEY SETUP

In this section, we describe our experiment. We set out to investigate what is the need for software team managers related to the identification of developer skills in the context of teams in software developing companies. We hope that this will provide a model with

information so that other researchers can develop more efficient methods and tools for identifying skills in software teams.

We pose the following research questions:

- *RQ*1: What issues can be addressed through a developer skills identification approach?
- *RQ*2: How can evaluating developers' skills help in managing projects, teams or employees?
- *RQ*3: What other business needs can be met by an approach to identifying developer skills?

2.1 Scope

Defining the scope of an experiment comes down to setting its goals. We are organizing this proposed by the Goal/Question/Metric (GQM) [29] template to do so. Therefore, the scope of this study can be summed up as follows: **analyze** the need for managers to identify skills in the context of teams **for the purpose of** evaluation **with respect to** ability to create effective methods of identifying developer skills **from the point of view of** the managers **in the context of** software engineering.

2.2 Variables Selection

As mentioned, the purpose of this experiment is to assess the needs of managers related to identifying developer skills on their team. Thus, we are particularly interested in the following dependent variable—understand the needs of development company managers—and the three following independent variables: (*i*) issues, (*ii*) applications and (*iii*) other needs that managers are interested in a skill detection approach.

2.3 Sample Selection

To perform this experiment, we invited only team managers, with the positions of Product Manager and Tech Leads. We tried to include a wide range of managers from the participating developer company to be able to generate relevant discussions.

We selected 17 managers — 11 Tech Leads and 6 Product Manager. The invitation was formalized by email, along with contextualization material and a survey so that participants could respond and reflect on the topic— use of a method to identify skills in their teams. We followed the guidelines proposed by Barbour [4] and Flick [8] during the construction of our sample, stating that the composition of the sample should not be a simple selection of people, but that the group is formed by participants who have a common alignment, to have a more adequate discussion about the content, presenting varied experience and perspectives so that there is a debate or divergence of ideas.

2.4 Operation

First, we selected managers from a large Fintech—located in the State of Minas Gerais, Brazil. Altogether, 17 managers were invited, leading 11 teams comprising approximately 75 developers. Only leadership positions were required for this focus group phase, since they were more involved with the company's problems, effectively knowing the structure, as a need and the work context of the teams. This inclusion criterion adopted above is based on the premise that research subjects have not been chosen randomly but intentionally by certain criteria and characteristics in common,

established in the plan and hypotheses that have been formulated by the researcher [28].

The invitation was formally made by email with background material on identifying developer skills, along with a questionnaire—no required–, so that participants could prepare for the focus group discussions. The intention of this material is to encourage participants to come up with ideas and proposals already formulated for the focus group meeting.

2.5 Execution

In order to accomplish the goals for this study and answer the research questions a focus group session was planned. According to Kontio et al. [15], the main steps to carry out a focus group, consist of: (*i*) define the research questions; (*ii*) plan the focus group event; (*iii*) select the participants; (*iv*) conduct the focus group session; e (*v*) analyze data and report results;

The definition of research questions (*i*) is described at the beginning of Section 2. The planning of the focus group event (*ii*) consisted of the idea of contextualizing the participants about the purpose of the research, with explanatory material and a questionnaire so that they could delve deeper into the topic and, on the day of the meeting, they would have better arguments for discussions. The selection of participants (*iii*), in accordance with Section 2.4, was carried out through an invitation, sent to 17 team managers—distributed between product manager and tech leads—, and had the support of 11 responses in the questionnaire and 6 participations on the day of the meeting, as shown in Table 1. The focus group session (*iv*) was conducted by video conference, mediated by the author. The meeting was recorded for later transcription. We used the questionnaire responses to promote group discussions, generating new insights into the topic. In analysis (*v*), we grouped all the content generated in the questionnaire and in the meeting into categories for better use of the information generated.

Table 1: Focus Group Participants

	Invits	Survey	Meeting
Product Manager	6	2	2
Tech Leads	11	9	4
Total	**17**	**11**	**6**

3 SURVEY RESULTS

This section describes the research results of the qualitative study used in the focus group. We argue, three categories of discussions generated in the focus group session: issues, applications in the company, and other needs. We have tried to collect the perceptions of managers of a software development organization, with the purpose of understand the problems and needs related to skill identification. 11 managers participated in the questionnaire, resulting in a response rate of nearly 64.7% of the guests. For the discussion session, we had a participation of 6 managers, resulting in a rate of nearly 35.29% of the guests. At the end, we mapped 5 problems, 4 possible applications and 7 other needs.

3.1 Overview

This section presents an overview of both evaluated categories, (I) issues, (AC) applications in the company, (ON) and other needs. The subjects generated in the category were quantified by number of interactions, that is, as participants interacted with a given subject, we counted this interaction, in this way we were able to quantify the frequency of participation in each of the subjects.

Although the (ON) category was the one that generated the greatest number of topics, the participants interacted more with discussions in the (AC) and (I) category, as shown in Figure 1. We can also see a greater number of interactions from tech leads, this is probably related to the proximity with the teams, since these managers are responsible for technically guiding the developers, and the product managers manage the project in general. Another factor that can also influence the number of product manager interactions is that they represent only 22.2% of the participants.
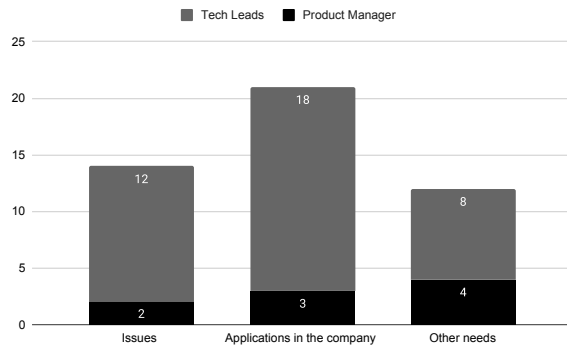

Figure 1: Number of interactions

3.2 Issues

In this section we answer the first research question (RQ1).
RQ1 - What issues can be addressed through a developer skills identification approach?

Below, it is possible to see the issues raised according to the responses of the participants, and as a result generated and categorized in the focus group:

- I_1: In legacy projects, it is very difficult to identify developers who understand the business flow implemented;
- I_2 Difficulty in knowing each developer's proficiency in large projects analyzing only the tasks performed;
- I_3 Difficulty in directing specific and urgent tasks;
- I_4 At first, it is not easy to identify the lack of theoretical and practical knowledge to perform tasks;
- I_5 It is not easy to identify who is implementing, reviewing or just fixing some part of the code.

The research question RQ1 is related to the possible problems involving the lack of a method for identifying skills in the daily routines of the company's managers. The problems discussed in this category range from the need to find professionals in the company who know the business flow, identification of those responsible for certain features, or simply difficulty in knowing the proficiency

of each developer based on the overview of each task. To aid the analyses, we grouped the managers' participations according to their interaction with each of the problems, as this can be seen in Figure 2.
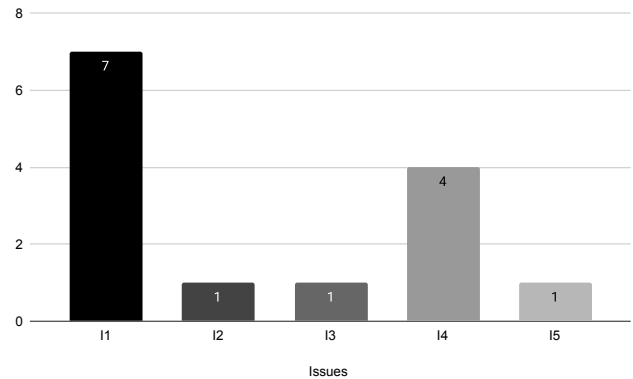

Figure 2: Interactions in the (I) category

It is possible to observe that I1 was the most discussed, with the participation of 7 managers, either answering the questionnaire or participating in group discussions. I1 is about finding more experienced developers who have skills in the business flow of legacy systems, either to perform such a task, or simply passing on the knowledge to other less experienced developers. Since such legacy systems do not have formal documentation.

The second most voted issue is I4, and is related to the difficulty of managers in knowing the theoretical and practical knowledge of developers. The other issues had only 1 interaction, and are related to the difficulty in directing specific tasks and what is the role of each developer in the team.

3.3 Applications in the company

In this section we answer the second research question (RQ2).
RQ2 - How can evaluating developers' skills help in managing projects, teams or employees?

Below, it is possible to see the issues raised according to the responses of the participants in the AC category and the discussions generated and categorized in the focus group:

- AC_1: Assist in identifying key developers who broaden expertise in the project flow;
- AC_2: Direct tasks more efficiently according to team registration skills;
- AC_3: Indicate the team's growth points to offer training and help balance it;
- AC_4: Group developers by roles according to the implementation context and their knowledge of business rules.

As seen in Figure 3, this category is the one with the highest number of participants' collaboration, where it is observed that AC1 and AC3 had the participation of 6 managers. The first application is directly related to the identification of developers with business skills and who dominate the project flow, whereas the second is related to the team's development points, that is, knowing which

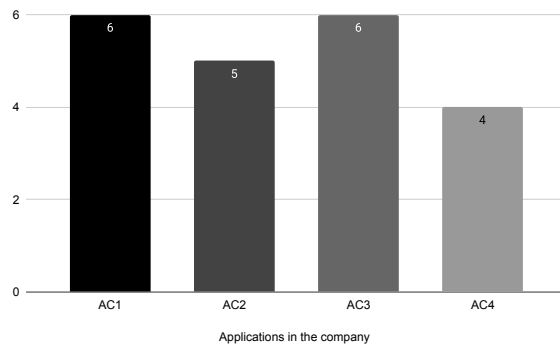skills the team needs to develop, and thus perform training to hone that skill.

Figure 3: Interactions in the (AC) category

We can also observe that AC4 is related to the need to classify developers in the context of implementation and application business rules, reinforcing the need for managers to solve I1, which is the difficulty they have in finding developers who know the flow of application business.

Finally, AC2 had a participation of 5 managers in its discussions, and in this need the leaders make it clear the importance of directing developers more efficiently according to their abilities.

3.4 Other needs

In this section we answer the third research question (RQ3).
RQ3 - What other business needs can be met by an approach to identifying developer skills?

Finally, it is possible to see below the issues answered and discussed in the ON category of the focus group:

- **ON$_1$**: Measure commit granularities to identify who else is implementing or refactoring;
- **ON$_2$**: Assist in identifying a person who knows business rules to assist the risk control department;
- **ON$_3$**: Identify training needs;
- **ON$_4$**: Identify targets for possible promotions;
- **ON$_5$**: Monitor project evolution based on contribution history;
- **ON$_6$**: Identify the evolution of the project regarding the number of refactorings or bugs;
- **ON$_7$**: Evaluate the practical projects in the test for recruiting new employees.

It is possible to observe that this category generated the largest number of topics discussed, as shown in Figure 4. However, there was a lower participation of managers. It is possible to observe that there is a greater interest in the discussion of the subject at ON4, which had the participation of 3 managers. This ON is related to job promotions, that is, developers want to assess their developers' knowledge to apply possible job changes (e.g. changing the developer from junior level to medium level).

ON5 and ON6 are related to the evolution of the project, whether in terms of technologies and their distribution, or simply who is
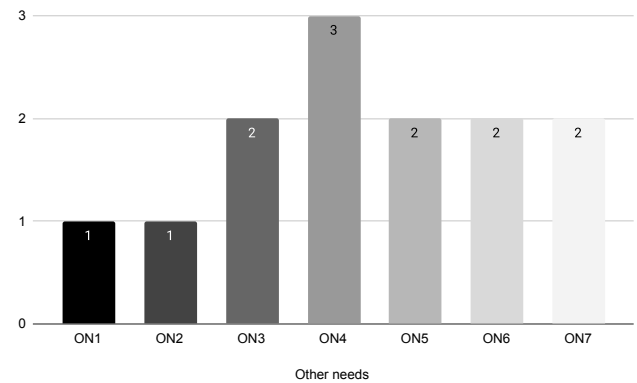

Figure 4: Interactions in the (ON) category

carrying out more implementations, refactorings or bugs. ON3 is related to the need for team training, that is, identifying which team developers need training according to their level of knowledge.

The ON2 stands out again in this category of discussions, which refers to identifying people who have knowledge of business rules, but to assist in documenting contingency plans and risks, which are measures that the company adopts, including activation manual processes, to make your vital processes work fully, or in a minimally acceptable state, as quickly as possible, thus avoiding a prolonged downtime that could cause greater damage to the corporation.

Finally, ON1 and ON7 speak respectively of measuring implementations and evaluating the company's selection process.

3.5 Threats to Validity

To minimize some threats to validity, the following actions were taken:

Regarding **Internal validity**, we carefully selected people in leadership positions and tried to conduct the focus group as impartially as possible, without interfering in the discussions generated. However, it is possible that some participants did not express their opinions fully due to some kind of shyness. Regarding **External validity**, the results of this study can not be generalized to other organizations, since the reality of the participating company may not be that of others. Finally, concerning **Conclusion validity**, the fact that we used a focal group, this can favor the deepening of discussions that are decisive for our approach, since the composition of the members was formed by participants who have a common alignment.

4 RELATED WORK

The proposed approach of this work to identify the needs of software team managers—in the context of a skills identification—it is not the first. However, as far as we know, previous approaches do not use focus groups to extract information from managers. Identifying developer skills in team members is a decisive factor in software engineering projects [19]. Currently, there are several approaches to extracting information from developers' skills, whether

it is accompanied by tools [16, 24], with the aid of machine learning [20, 21], extracting information from developer contribution logs [6, 11], or simply analyzing the developers' curriculum [3, 13]. To support skills identification methodologies, it is important to know which skills managers consider important to build a qualified team.

Matturro et al. [19] analyzes which skills are more important to leaders and other team members. In a similar work to the previous one, Mtsweni et al. [22] assesses which skills are important to the project's success. Finally, taking a validation approach through a focus group, [12] presented a survey with the objective of identifying and classifying the essential skills for a Scrum team.

5   CONCLUSION AND FUTURE WORK

It is possible to notice in the discussions generated in the focus group that managers present a certain difficulty in identifying developers with skills in the company's business flow. The need to assess weather knowledge and apply training, as well as promotions, also stood out at the event. We believe that this further reinforces the construction of a methodology to identify skills aimed at the application's business flow, which can help managers to be more assertive about the assignment of more complex tasks or even team alignment, which is possible to identify the main developers of that project.

The focus group managed to show 5 possible problems that the lack of skills identified can cause in the management of teams. It was also possible to map 4 applications that would likely help directly in the management, and 7 other needs. We also saw that managers believe that a method that helps to identify skills could be useful for the company, helping to better plan to allocate resources to the project.

For future works, we intend to create a methodology aimed at identifying skills in business flows, using the support of a tool for identifying skills based on logs of contributions from VCS developers. We believe that a tool that can deliver statistics on the skills of the developers on the team can contribute to better project plan according to the resources allocated, in addition to offering ways to identify possible training and salary promotions for the company's employees.

REFERENCES

[1] Ali Sajedi Badashian, Afsaneh Esteki, Ameneh Gholipour, Abram Hindle, and Eleni Stroulia. 2014. Involvement, contribution and influence in GitHub and stack overflow.. In *14th Proceedings of 24th Annual International Conference on Computer Science and Software Engineering (CASCON)*. 19–33.

[2] Jiří Balcar. 2014. Soft Skills and Their Wage Returns: Overview of Empirical Literature. *Review of Economic Perspectives* 14, 14 (2014), 3–15.

[3] Shivam Bansal, Aman Srivastava, and Anuja Arora. 2017. Topic Modeling Driven Content Based Jobs Recommendation Engine for Recruitment Industry. *Procedia Computer Science* 122, 9 (2017), 865 – 872.

[4] Rosaline Barbour. 2009. *Grupos focais: coleção pesquisa qualitativa*. Bookman Editora.

[5] Anthony P. Carnevale and Nicole Smith. 2013. Workplace basics: the skills employees need and employers want. *Human Resource Development International* 16, 5 (2013), 491–501.

[6] Eleni Constantinou and Georgia M. Kapitsaki. 2016. Identifying Developers' Expertise in Social Coding Platforms. In *42th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*. 63–67.

[7] Laura Dabbish, Colleen Stuart, Jason Tsay, and Jim Herbsleb. 2012. Social Coding in GitHub: Transparency and Collaboration in an Open Software Repository. In *12th Conference on Computer Supported Cooperative Work (ACM 2012)*. 1277 – 1286.

[8] Uwe Flick. 2009. Desenho da pesquisa qualitativa. In *Desenho da pesquisa qualitativa*. 164–164.

[9] Raluca Florea and Viktoria Stray. 2019. The skills that employers look for in software testers. *Software Quality Journal* 27, 4 (2019), 1449–1479.

[10] Thomas Fritz, Gail C. Murphy, and Emily Hill. 2007. Does a programmer's activity indicate knowledge of code?. In *6th European Software Engineering Conference and the ACM SIGSOFT International Symposium on Foundations of Software Engineering (ESEC-FSE)*. 341–350.

[11] Claudia Hauff and Georgios Gousios. 2015. Matching GitHub developer profiles to job advertisements. In *12th Working Conference on Mining Software Repositories (IEEE/ACM)*. 362–366.

[12] Anita Hidayati, Eko K. Budiardjo, and Betty Purwandari. 2020. Hard and Soft Skills for Scrum Global Software Development Teams. In *3rd International Conference on Software Engineering and Information Management (ICSIM)*. MISSING.

[13] Kameni Florentin Flambeau Jiechieu and Norbert Tsopze. 2020. Skills prediction based on multi-label resume classification using CNN with model predictions explanation. *Neural Computing and Applications* 32 (2020), 1433–3058.

[14] E. Kalliamvakou, D. Damian, K. Blincoe, L. Singer, and D. M. German. 2015. Open Source-Style Collaborative Development Practices in Commercial Projects Using GitHub. In *37th IEEE International Conference on Software Engineering (IEEE/ACM)*. 574–585.

[15] J. Kontio, L. Lehtola, and J. Bragge. 2004. Using the focus group method in software engineering: obtaining practitioner and user experiences. In *4th International Symposium on Empirical Software Engineering (ISESE)*. 271–280.

[16] S. Kourtzanidis, A. Chatzigeorgiou, and A. Ampatzoglou. 2020. RepoSkillMiner: Identifying software expertise from GitHub repositories using Natural Language Processing. In *35th IEEE/ACM International Conference on Automated Software Engineering (ASE)*. 1353–1357.

[17] Jennifer Marlow and Laura Dabbish. 2013. Activity traces and signals in software developer recruitment and hiring. In *13th conference on Computer supported cooperative work (CSCW)*. 145–156.

[18] Jennifer Marlow, Laura Dabbish, and Jim Herbsleb. 2013. Impression Formation in Online Peer Production: Activity Traces and Personal Profiles in GitHub. In *13th Conference on Computer Supported Cooperative Work (CSCW)*. 117–128.

[19] Gerardo Matturro, Florencia Raschetti, and Carina Fontan. 2015. Soft Skills in Software Development Teams: A Survey of the Points of View of Team Leaders and Team Members. In *8th International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE)*. MISSING.

[20] Joao Eduardo Montandon, Luciana Lourdes Silva, and Marco Tulio Valente. 2019. Identifying experts in software libraries and frameworks among GitHub users. In *2019 IEEE/ACM 16th International Conference on Mining Software Repositories (MSR)*. IEEE, 276–287.

[21] João Eduardo Montandon, Marco Tulio Valente, and Luciana L. Silva. 2021. Mining the Technical Roles of GitHub Users. *Information and Software Technology* 131 (2021), 106485.

[22] Emmanuel S. Mtsweni, , Tertia Hörne, and John Andrew van der Poll. 2016. Soft Skills for Software Project Team Members. *International Journal of Computer Theory and Engineering* 8, 2 (2016), 150–155.

[23] Chris Northwood. 2018. *The Full Stack Developer: Your Essential Guide to the Everyday Skills Expected of a Modern Full Stack Web Developer*. Springer.

[24] Johnatan Oliveira, Markos Viggiato, and Eduardo Figueiredo. 2019. How Well Do You Know This Library? Mining Experts from Source Code Analysis. In *18th Brazilian Symposium on Software Quality (SBQS)*. 49–58.

[25] Michael Polanyi. 1966. The Tacit dimension. *New York: Doubleday & Co.* MISSING, MISSING (1966), MISSING.

[26] David Schuler and Thomas Zimmermann. 2008. Mining usage expertise from version archives. In *08th Mining software repositories (MSR)*. 121–124.

[27] Ardian Sopa, Masduki Asbari, Agus Purwanto, Priyono Budi Santoso, Dhaniel Hutagalung Mustofa, Siti Maesaroh, and R Primahendra. 2020. Hard skills versus soft skills: Which are more important for indonesian employees innovation capability. *International Journal of Control and Automation* 13, 2 (2020), 156–175.

[28] Claes Wohlin. 2013. Empirical software engineering research with industry: Top 10 challenges. In *1st international workshop on conducting empirical studies in industry (CESI)*. 43–46.

[29] C. Wohlin, P. Runeson, M. Höst, M. C. Ohlsson, B. Regnell, and A. Wesslén. 2012. *Experimentation in Software Engineering*. Springer. 236 pages.

[30] Alexey Zagalsky, Joseph Feliciano, Margaret-Anne Storey, Yiyun Zhao, and Weiliang Wang. 2015. The Emergence of GitHub as a Collaborative Platform for Education. In *18th ACM Conference on Computer Supported Cooperative Work Social Computing (CSCW)*. 1906–1917.

[31] Cheng Zhou, Sandeep Kaur Kuttal, and Iftekhar Ahmed. 2018. What Makes a Good Developer? An Empirical Study of Developers' Technical and Social Competencies. In *18th Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*. 319–321.