Difficulties of Newcomers Joining Software Projects Already in Execution

Conference Paper · May 2018

DOI: 10.1109/CSCI.2017.171

3 authors, including:

Gerardo Matturro
Universidad ORT Uruguay
34 PUBLICATIONS **209** CITATIONS

SEE PROFILE

2017 International Conference on Computational Science and Computational Intelligence

Difficulties of newcomers joining software projects already in execution

Gerardo Matturro
Departamento
de Ingeniería de software
Universidad ORT Uruguay
Montevideo, Uruguay
matturro@uni.ort.edu.uy

Karina Barrella
Departamento
de Ingeniería de software
Universidad ORT Uruguay
Montevideo, Uruguay
kbarrella@gmail.com

Patricia Benitez
Departamento
de Ingeniería de software
Universidad ORT Uruguay
Montevideo, Uruguay
patobenitez@gmail.com

***Abstract*— It is not uncommon that, in software projects, it is necessary to incorporate new developers at an advanced stage of project execution. These "newcomers" face various difficulties and challenges to find their place in the project that prevents them from starting to contribute quickly to the progress of the project. This article reports the results of an exploratory-descriptive study aimed at identifying the difficulties faced by new team members when they join a project later, as well as identifying actions that are often adopted to mitigate these difficulties. The study reveals that scarse or null documentation and the need to know the product under construction are the main difficulties, while the assignment of a referent and the provision of training are mentioned as the main actions organizations usually take to mitigate those problems.**

Keywords—newcomers, software projects, onboarding

I. INTRODUCTION

According to Fairley, a particular factor that distinguishes software projects from other types of projects is that software development projects are team-oriented and knowledge-intensive efforts [1]. To this author, teams are necessary because it would take too long for a single person to develop a modern software system and because it is unlikely that a single person has the necessary range of required skills to develop it.

When a developer joins a running software development project, he faces a situation where he must become familiar with various technical, technological, organizational, and social aspects of the project itself and the team that is already participating in the execution of that project [2].

These "newcomers" face various difficulties and challenges in finding their place in the project and being able to contribute quickly, with their previous knowledge and experience, to the progress of the project.

In this regard, Rastogi and colleagues argue that while companies expect early contributions to the project, new members often require several weeks to achieve the same level of productivity as existing employees. [3].

Adding people to a software development team does not, as a rule, increase overall productivity in a linear manner because the increased overhead of communicating with and coordinating work activities among the added people decreases the productivity of the existing team [1].

It is well known the Brooks law on this aspect, which states, in a somewhat simplified way, that adding manpower to a late software project makes it later. [4].

Despite the importance of this issue, little is known about the barriers faced by newcomers when they participate in a project, a process that, according to Wolff-Marting and colleagues, still presents many open issues [5].

The purpose of this article is to report the process followed and the results obtained from an exploratory-descriptive study oriented a) to identify the problems and difficulties that new members of a project team usually face when they join a project that is already in execution, and b) to know the actions and strategies at individual, team, and organizational levels that are usually adopted to mitigate those problems and difficulties.

The focus of this study was the software industry in Uruguay. With a population of 3.2 million people, Uruguay has positioned itself in recent years as a leading exporter of software in Latin America. In 2010, exports of software and related services reached 250 million dollars, and CEOs of leading companies expect to reach 1 billion dollars by 2020. The main foreign markets are the United States, Argentina, Brazil, Spain, and Canada. At present, there are about 450 companies that produce software, and the unemployment rate in this industrial sector is almost zero.

The rest of this article is organized as follows. In section II we present a review of the main concepts and definitions related to work teams, norms and rules of behavior, and some related works to the researched topic. In section III we explain the methodological design of the study, which discusses research questions and data collection and preparation procedures. Section IV presents the main results obtained, while section V presents the analysis of these results. In section VI we compare our findings with the ones reported in similar studies on the subject. Section VII addresses the threats to the validity of this study, and section VIII presents the conclusions and outlines future research work.

II. BACKGROUND AND RELATED WORKS

A team consists of at least three people who share a common goal, perform specific functions, and depend on each other to achieve that common goal.

In fact, Uhl-Bien and colleagues define a **team** as a group of people brought together to use their complementary skills to achieve a common purpose for which they are collectively accountable [6].

The success of the team depends on the cooperation of all members. Humphrey and colleagues argue that teams are often needed for tasks involving more work, the availability of a

978-1-5386-2652-8/17 $31.00 © 2017 IEEE
DOI 10.1109/CSCI.2017.171

variety of skills, or some other capacity that a single person could not provide alone [7].

Usually, teams are conformed at the initial stage of the projects they must execute. However, and due to various circumstances, the need to incorporate new members into a team already formed, and at a later stage of project implementation, may occur.

We define the concept of "newcomer" as that of the person who joins a project team at a time other than the formation of the initial team.

According to George and Jones, a group's ability to control the behaviors of its members depends on the extent to which newcomers learn the roles, rules and norms of the group. Newcomers do not know at the outset what is expected from them and what they can and cannot do [8].

Several authors addressed the study of problems and difficulties faced by newcomers to a software project already in execution, but mainly focusing on *open source* projects.

This is the case of Steinmacher and colleagues who carried out a systematic literature review on the subject [9]. In their study, which involved the review of 21 research papers, they identified a series of "onboarding" barriers that they classified into five categories, named: Finding a way to start, Social interactions, Code issues, Documentation problems and Newcomers' previous knowledge.

Other studies on the same topic, also focusing on open source projects and reported by this author with the same or with different colleagues, are [10], [11] y [12]. The latter, also addresses crowdsourcing software projects.

Panichella and colleagues also addressed the study of newcomers to software projects, and also focused on open source projects, in [13] y [14].

In a similar vein, Fagerholm and colleagues report on what they call the "onboarding process" of new members, also related to open-source projects: [15] y [16].

On the other hand, there are studies on the process of integration of newcomers to software projects, but with an exclusive focus on industrial environments of software production.

These are the cases of Begel and Simon [17] and of Dagenais and colleagues [2]. The latter reports the identification of five categories or "landscapes" of difficulties faced by newcomers: Product, Process and practices, Team, Documentation, and Organizational context.

The study reported in this paper also focuses on an industrial setting of software development.

III. METHODOLOGICAL DESIGN

The methodological design of this study consisted in the definition of the research questions, the selection of key informants, the elaboration of data collection instruments, their application in personal interviews, the critical reading and coding of the answers obtained, and the analysis and interpretation of the obtained data.

A. Research questions

For this study the following research questions were posed:

- RQ1: What are the problems and difficulties encountered by newcomers when joining an ongoing software development project?
- RQ2: What actions and strategies are usually implemented at the organizational and team levels to mitigate those problems and difficulties?

B. Key informants selection

To contact key informants who could give insight into the problems of newcomers and the strategies used to mitigate them, 16 software development companies based in Montevideo, Uruguay were contacted. Nine of them finally agreed to participate in the study. Of these nine companies, five mentioned that they manage their projects by applying some agile practices (without specifying details), while the other four manage their software projects following a more traditional methodology.

To collect data for this study, twelve team leaders or project managers and eight team members were interviewed, all of them working on software project teams belonging to these nine participating companies.

Regarding the team leaders, the selection criterion for interviewing them was that during their project management activities, a new member had been added to the team after the start of the project. These team leaders are professionals with between 4 and 17 years of general experience in software development, and with a history of having led project teams at least in three opportunities.

Regarding team members, the selection criterion was that, at least in one occasion, they had lived the personal experience of having joined a project team in an instance after the start of project execution. All team members interviewed indicated that they played the role of programmer, and that they had little or no professional software development experience at the time they joined the project team as newcomer.

C. Data collection

For the interviews, two questionnaires were designed: one for team leaders or project managers, and one for team members. Both questionnaires have closed and open-ended questions, and their purpose was to guide the authors in conducting the interviews and to ensure the collection of data necessary to answer the research questions.

D. Reading and coding of answers

For coding the answers to the open questions the procedure proposed in [18] was followed; that is, open-ended questions are coded once all participants' responses are known, or at least the main response trends in a sample of the questionnaires applied. With the coding of open questions certain categories are obtained that represent the final results.

The procedure, then, consisted in finding and naming the general response patterns (similar or common responses), list-

ing these patterns, and then assigning a name or textual value to each pattern. In this way, each pattern constitutes a specific category of responses.

IV. DATA ANALYSIS

In this section we present the main results obtained from the interviews, and after performing the coding and categorization of responses.

Of the various questions included in the questionnaires, we present here the ones directly related to the research questions formulated in Section III.

A. Answers obtained from team members

For team members who once were "newcomers" to a running software project, the specific question related to RQ1 was: What were the difficulties you faced when joining the project?

Table I lists the categories of difficulties identified in the responses of the team members, and the corresponding times that were mentioned.

TABLE I. DIFFICULTIES MENTIONED BY TEAM MEMBERS AND FREQUENCY OF MENTIONS

Difficulties	Frequency
Documentation	3
Undestanding of solution / Knowing the product	3
Comunication	1
Work methodology	1
Tools and technology in use	1

The meanings of the categories of difficulties mentioned are as follows:

- **Documentation**: It refers to the little or no documentation of the source code already developed, either in the form of comments in the code or in the form of "written notes". It also refers to the lack of documentation on the initial software requirements and the changes that have occurred since the beginning of the project.
- **Understanding the solution / Knowing the product:** It refers to have an overall view of the system in development, the business problems that are intended to solve with the new system, the generalities of the proposed solution and the organization and current state of the system which is being built.
- **Communication**: It refers to the little and incomplete communication from the rest of the team to explain the changes, issues, and project decisions that were made during the period prior to joining the team.
- **Work methodology**: It mainly refers to knowing and familiarizing with the software process in use by the team, with the project management methodology, with the software development life cycle being applied, as well as the work and meeting routines, and with the forms and criteria for assigning and delegating tasks.
- **Tools and technology in use**: It refers to knowing and learning how to use the software tools that are being used in the development process, as well as to the diverse technologies of solution implementation, particularly those related to software architecture.

B. Answers obtained from team leaders

Here we present the questionnaire questions used for team leaders or project managers, specifically related to the research questions, and the summary of the answers obtained.

Question: What difficulties do new team members face when joining a project already under way?

Table II lists the categories of difficulties faces by newcomers to a project from the point of view of team leaders or project managers, and the corresponding times they were mentioned.

TABLE II. DIFFICULTIES MENTIONED BY TEAM LEADERS AND FREQUENCY OF MENTIONS

Difficulties	Frequency
Undestanding of solution / Knowing the product	9
Work methodology	6
Knowing the company	5
Integration to the team	5
Understanding the project	3
Documentation	3
Tools and technology in use	3

The meanings of the categories of difficulties mentioned are as follows:

- **Understanding the solution / Knowing the product**: It refers to have an overall view of the system in development, the business problems that are intended to solve with the new system, the generalities of the proposed solution and the organization and current state of the system which is being built.
- **Work methodology**: It mainly refers to knowing and familiarizing with the software process in use by the team, with the project management methodology, with the software development life cycle being applied, as well as the work and meeting routines, and how tasks and responsibilities are assigned and delegated.
- **Knowing the company**: It refers to the fact that when the "newcomer" is not only new to the team but also to the company (recently hired), he feels the lack of knowledge about the company itself, its organizational structure and culture, its mission, the types of clients or market to which it directs its operations or services of software development.
- **Team integration**: It refers to the initial steps in the team, to know "who is who", to whom to ask for help or advice, to establish relationships with the other team members, to know their roles and their personal and technical characteristics.

- **Understanding the project**: It refers to knowing, roughly, what the project is about, what its objectives are, planned implementation time, current progress, main functionalities and "urgencies" of development.
- **Documentation**: It refers to the little or no documentation of the source code already developed, either in the form of comments in the code or in the form of "written notes". It also refers to the lack of good documentation on the initial software requirements and the changes that have occurred to those requirements since the beginning of the project.
- **Tools and technology in use**: It refers to knowing and learning how to use the software tools that are being used in the development process, as well as to the diverse technologies of solution implementation, particularly those related to software architecture.

Question: What actions or mechanisms do you usually implement to mitigate these difficulties?

Table III lists the categories of actions or mechanisms implemented by team leaders or project managers to deal with the difficulties of the newcomers, and the times (interviewees) that were mentioned.

TABLE III. ACTIONS IMPLEMENTED BY TEAM LEADERS AND FREQUENCY OF MENTIONS

Actions implemented	Frequency
Asign a referrent	10
Monitoring and follow-up	6
Training	5
Documentation	5
Support for the person	1
Assign simple functionalities	1
Give him “freedom”	1
Define an integration plan	1
Pair programming	1

The meaning of these categories of actions mentioned are as follows:

- **Assign a referent:** It refers to assigning an experienced team member to coaching the newcomer, to guide him in the technical aspects of the project and to facilitate his integration to the team.
- **Supervision and follow-up**: It refers to continuous monitoring of tasks assigned to the newcomer, with spaces to consult doubts.
- **Training**: It refers to provide the newcomer with some training regarding the role to be played in the team, as well as about the general aspects of the project.
- **Documentation**: It refers to giving the newcomer access to the documentation that may exist on the project, and giving him time to read, internalize its technical details and to ask questions.
- **Support to the person**: It refers to providing emotional support to motivate and guide the person, and to facilitate their personal and technical integration to the team.
- **Assign simple functionalities**: It refers to assigning the newcomer simple project tasks that allow him to gain confidence in his role, and to begin to know the product in development.
- **Give him “freedom”:** It refers to allow and encourage the newcomer to give his opinions, and to propose changes and personal points of view that make him feel comfortable.
- **Define an integration plan**: It refers to establishing a personalized plan of gradual assignment of tasks and responsibilities.
- **Pair programming**: It refers to apply this programming approach.

V. ANALYSIS OF RESULTS

This section presents the analysis of the collected data, and the answers to the research questions raised at the beginning.

A. Difficulties faced by newcomers

The research question RQ1 established at the beginning refers to what are the problems and difficulties that usually find the people who join a software project that is already under way.

Table IV summarizes the categories of difficulties mentioned by team leaders and by the "newcomers" themselves.

The difficulties perceived in common are: a) little or no documentation, mainly of the source code developed so far, b) lack of knowledge of the main characteristics and functionalities of the product under development, c) need to know and learn the methodology used by the team to which the newcomer is incorporated, and d) lack of knowledge or experience in the implementation technology or in the software tools in use.

TABLE IV. COMPARISON OF DIFFICULTIES REPORTED BY TEAM LEADERS AND BY TEAM MEMBERS

Difficulties	Team leaders	Team members
Communication	---	X
Knowing the company	X	---
Documentation	X	X
Understanding the solution / Knowing the product	X	X
Team integration	X	---
Work methodology	X	X
Tools and technology in use	X	X

B. Actions implemented to mitigate the difficulties

The research question RQ2 refers to knowing, from the point of view of team leaders or project managers, what actions and mechanisms they usually implement to reduce or eliminate the difficulties and problems faced by "newcomers" to a project in progress.

Of the various actions shown in Table III, the assignment of a team member as a coach to guide the new team member is the most usual option, followed by the monitoring and supervision of the tasks assigned to the new member, and the provision of training and documentation (if any) about the project and the product under construction.

VI. COMPARISON TO PREVIOUS WORKS

With the summary of difficulties categories presented in Table IV, we are now in position to compare those ones with the "landscapes" identified by Dagenais et al. in [2], as shown in Table V.

TABLE V. MAPPING BETWEEN CATEGORIES IN THIS STUDY AND DAGENAIS "LANDSCAPES"

Dagenais et al. "landscapes"	Categories in this study
Context	Knowing the company
Documentation	Documentation
Processes and practices	Tools and technology in use Work methodology
Product	Undestanding the solution/Knowing the product
Team	Communication Team integration

This comparison shows that evidence exists about a common set of difficulties and problems newcomers found when joining a software project already in execution.

We based this mapping on the sub-categories of the landscapes explained in [2] and the description of our categories described in section IV.

We left out of the comparison the other studies reviewed in section II because the latter refer to open source projects, not to projects developed in an industrial environment (which was the focus of this study, as stated at the beginning).

VII. THREATS TO VALIDITY

The most relevant threats are mainly related to selected key informants. Given their small number, they are not a representative sample of the total number of professionals who, at some point in their lives, were "newcomers" to a project. This population is even difficult to quantify. On the other hand, the team members interviewed had little or no working experience at the time of being "newcomer", which left out professionals who lived that situation in an advanced instance of their professional lives, that is, those that we can call "newcomers with prior professional experience".

VIII. CONCLUSION AND FURTHER WORK

The purpose of this paper was to report the first results of a study aimed at identifying the initial difficulties and problems faced by "newcomers" to a software project that is already underway, and to know the actions that are usually taken to mitigate those difficulties.

From the previous analysis of the answers obtained from the team members interviewed, the most outstanding problems refer to the lack of documentation of the product in development, and the lack of knowledge of the methodology of work and the technologies and development tools used by the team.

To mitigate these problems, team leaders mention as the main strategy the assignment of a referent to accompany the newcomer in its first steps in the project. They also mention that newcomers are provided with training and with the documentation that may exist.

As future work we will deepen this study, particularly on the following questions: a) what type of documentation and on what aspects of the project or product do the newcomers consider they should have? b) how effective are the strategies adopted by the project managers to reduce or eliminate the difficulties of new members of a running software project team? c) what personal and technical characteristics should the person assigned reference to accompany the newcomer to a project?

REFERENCES

[1] R. Fairley, *Managing and leading software projects*. Hoboken, NJ: John Wiley & Sons, 2009.

[2] B. Dagenais, H. Ossherz, R. K. E. Bellamyz, M. P. Robillard, and J. P. de Vries, "Moving into a New Software Project Landscape," presented at the 32nd International Conference on Software Engineering (ICSE 2010), Cape Town, 2010.

[3] A. Rastogi, S. Thummalapenta, T. Zimmermann, N. Nagappan, and J. Czerwonka, "Ramp-up Journey of New Hires: Do strategic practices of software companies influence productivity?" in *ISEC 2017: Innovations in Software Engineering Conference*, Jaipur, India, 2017.

[4] F. Brooks, *The Mythical Man-Month. Essays on Software Engineering. Anniversary edition*. Boston, MA: Addison Wesley, 1995.

[5] V. Wolff-Marting, C. Hannebauer, and V. Gruhn, "Patterns for tearing down contribution barriers to FLOSS projects," presented at the 12th Intl. Conf. on Intelligent Software Methodologies, Tools and Techniques (SoMeT 2013), Budapest, 2013, pp. 9–14.

[6] M. Uhl-Bien, J. R. Schermerhorn, and R. Osborn, *Organizational behavior*, 13th ed. Hoboken, NJ: John Wiley & Sons, 2014.

[7] W. S. Humphrey, T. Chick, W. R. Nichols, and M. Pomeroy-Huff, *Team Software Process (TSP) Body of Knowledge (BOK), SEI Technical report CMU/SEI-2010-TR-020*. Hoboken, NJ: John Wiley & Sons, 2010.

[8] J. M. George and G. Jones, *Understanding and managing organizational behavior*, 6th ed. Upper Saddle River, NJ: Pearson, 2010.

[9] I. Steinmacher, M. A. Graciotto Silva, and M. A. Gerosa, "Barriers Faced by Newcomers to Open Source Projects: A Systematic Review," presented at the 10th IFIP WG 2.13 International Conference on Open Source Systems, OSS 2014, San José, Costa Rica, 2014.

[10] I. Steinmacher, T. Uchoa Conte, Ch. Treude, and M. A. Gerosa, "Overcoming Open Source Project Entry Barriers with a Portal for Newcomers," presented at the 38th IEEE International Conference on Software Engineering (ICSE 2016), Austin, TX, 2016.

[11] I. Steinmacher, I. Scaliante Wiese, and M. A. Gerosa, "Recommending Mentors to Software Project Newcomers," presented at the 3rd. International Workshop on Recommendation Systems for Software Engineering (RSSE 2012), Zurich, 2012.

[12] A. Lazaretti, I. Steinmacher, L. Santos Machado, C. R.B. de Souza, and R. Prikladnicki, "Barriers Faced by Newcomers to Software-Crowdsourcing Projects," *IEEE Softw.*, vol. 34, no. 2, pp. 37–43, 2017.

[13] G. Canfora, M. Di Penta, R. Oliveto, and S. Panichella, "Who is going to mentor newcomers in open source projects?" presented

at the 20th International Symposium on the Foundations of Software Engineering, Cary, North Carolina, 2012.
[14] S. Panichella, "Supporting Newcomers in Software Development Projects," presented at the 31st International Conference on Software Maintenance and Evolution (ICSME2015), Bremen, 2015.
[15] F. Fagerholm, P. Johnson, A. Sánchez Guinea, J. Borenstein, and J. Münch, "Onboarding in Open Source Software Projects: A Preliminary Analysis," presented at the 8th International Conference on Global Software Engineering Workshops (ICGSEW 2013), Bari, 2013.
[16] F. Fagerholm, A. Sánchez Guinea, J. Borenstein, and J. Münch, "Onboarding in Open Source Projects," *IEEE Softw.*, vol. 31, no. 6, pp. 54–61, 2014.
[17] A. Begel and B. Simon, "Novice Software Developers, All over Again," presented at the ICER '08: 4th Intl. Workshop on Computing Education Research, Sydney, 2008.
[18] P. H. Sampieri, R. Fernández, and C. Baptista, *Metodología de la investigación*, 5th ed. México: McGraw Hill, 2010.