Exploring Onboarding Success, Organizational Fit, and Turnover Intention of Software Professionals

Gaurav G. Sharma[a,c], Klaas-Jan Stol[b,c]

[a] *IBM, Cork, Ireland*
[b] *Lero—the Irish Software Research Centre*
[c] *School of Computer Science and Information Technology*
University College Cork, Ireland.

Abstract

The IT sector struggles with talent acquisition and low retention rates. While several field studies have explored onboarding of software developers, the software engineering literature lacks studies that develop and evaluate theoretical models. This study seeks to explore the link between onboarding of new hires and turnover intention of these professionals. In particular, we develop a theoretical model that identifies a number of onboarding activities, and link these to onboarding success. We then look at what we have termed "organizational fit," which we define as two aspects of software professionals, namely job satisfaction and the quality of their relationships on the workfloor, and investigate how these mediate the relation between short-term onboarding success and a longer-term intention to leave (or stay with) an organization. We test our model with a sample of 102 software professionals using PLS-SEM. The findings suggest that providing support to new hires plays a major role in onboarding success, but that training is less important. Further, we found that job satisfaction mediates the relationship between onboarding success and turnover intention, but workplace relationship quality does not. Based on the findings, we discuss a number of implications for practice and suggestions for future research.

Keywords: Onboarding, job satisfaction, turnover intention, survey, PLS

1. Introduction

Software professionals are extremely mobile in today's highly interconnected world (Forrest, 2018). Developers could potentially work from any place with an Internet connection, and can therefore change jobs very easily. Besides this type of flexibility, it is quite common for developers to move to a new company every few years or so (Miller, 2018). Each time an organization hires a new software developer, this person must be introduced to the organization, its processes, and its culture. Typically this is done through an *onboarding* process. Recruiting and onboarding people with the right skills and personality is crucial to the success of software development organizations and projects (Hall et al., 2008). Recruitment, however, is an expensive activity; it is not unusual that newly recruited developers are initially a liability since there is a learning curve that any employee goes through before becoming productive (Brooks, 1975; DeMarco and Lister, 1987). Some experts suggest it may take up to 6 to 12 months before new recruits become productive (Sim and Holt, 1998).

In addition to the "productivity delay," there is also a chasm between the skills and knowledge that new graduates' formal education offers, and what industry requires (Brechner, 2003; Legier et al., 2013; Radermacher and Walia, 2013; Tang et al., 2001; Trent, 1988). Technology is changing constantly (Trent, 1988), and while the technical challenges that newcomers face can be significant, non-technical skills have also consistently been found to be very important and highly rated skills by employers (Aasheim et al., 2009; Begel and Simon, 2008b; Legier et al., 2013; McMurtrey et al., 2008; Simon and Jackson, 2013; Tesch et al., 2008).

Onboarding, also known as *"organizational socialization"* in the management literature (Bauer, 2010; Van Maanen and Schein, 1979), is a formal or informal process of integrating newly hired employees and transforming them from being "outsiders" to productive members of the organization. This involves the transfer of knowledge, skills, rules, and familiarity with the organizational culture to be able to work within a team (Britto et al., 2018; Sim and Holt, 1998). Research suggests that the first 90 days are crucial and decide the success of a newcomer in his or her job (Watkins, 2013), and it is typical that onboarding activities take place during this initial period. This period is also important from a financial perspective, because hiring and onboarding new employees is a costly process. The costs associated with onboarding new employees are due to an initial low level of productivity, and HR administration and bureaucracy involved in hiring new staff (Snell, 2006). In the context of software development, developing an understanding of an unfamiliar codebase written by others can be a complicated task for most software developers (LaToza et al., 2006). Therefore, the importance of the onboarding process looms even larger for multinational software companies working across the globe and which often depend on legacy systems. A lack of an effective plan could cause problems in maintaining the software (Littman

et al., 1987), which could lead to a longer lead time to full productivity (Sim and Holt, 1998).

It is worth clarifying what we mean by the term "onboarding." In this article we adopt the term to mean the same as in the management literature cited above, namely the process of organizational socialization. This is consistent with Microsoft's use of the term defined by Begel and Simon (2008b), who define "onboarding" as *"the Microsoft term for the orientation process by which new hires adjust to and become effective software developers within the corporation."* Other metaphors to characterize newcomers and their journey have been used, such as *"explorers who must orient themselves within an unfamiliar landscape"* (Dagenais et al., 2010), *"ramp-up journey"* (Rastogi et al., 2015), and *"software immigrants"* who must adapt through a process of *"naturalization"* (Sim and Holt, 1998). Following Sim and Holt (1998), we note that the term "novice" may be inappropriate because it implies that a new recruit is a new graduate, but clearly new recruits may have extensive industry experience when they come from other organizations. Hence, we adopt the term "newcomer" in this article.

The interpretation of the term "onboarding" described above is a holistic process of settling in, rather than the more narrow meaning of becoming familiar with a specific software code base which has also been used by some software engineering researchers (cf. Yates et al. (2019)). There is a considerable body of literature that focuses on understanding legacy systems and program comprehension, pioneered primarily by the software maintenance community (represented by the International Conference on Software Maintenance and Evolution (ICSME) and related journals such as *Journal of Software: Evolution and Process*) (Von Mayrhauser et al., 1997). Specific techniques include information seeking (O'Brien, 2007), feature location (Dit et al., 2013), source code analysis (Sillito et al., 2008), reading software documentation (Lethbridge et al., 2003), and querying knowledgeable peers with a longer tenure at the organization (Hertzum and Pejtersen, 2000). While these techniques are related to onboarding software developers in a more specific sense, in this article we focus on the more holistic process of onboarding suggested above.

There are many studies in the management literature that discuss strategies and best practices in *general*—including Van Maanen and Schein (1979)'s seminal Theory of Organizational Socialization—but few focus specifically on software professionals. The software engineering domain faces particular challenges given regular reports of shortages of software engineers, briefly mentioned above, including newcomers' productivity delay, a fast rate of technology change, and a high level of turnover of developers. Some media have reported that the turnover in the software sector is the highest of all (Forrest, 2018). A recent study by Gupta et al. (2018) investigating the relationship between new hires' onboarding experience and their intention to leave (what we refer to as "turnover intention") found that the latter was highest for the IT sector. Hence, this suggests the need for further studies that explore this relationship in the IT sector. Another reason to focus specifically on the software sector is that new hires, or what Sim and Holt (1998) have labeled "software immigrants," must acquire a wide variety of knowledge in order to become productive. Besides general knowledge such as programming languages and tools, company-specific knowledge must be acquired such as project jargon, team dynamics, coding standards, and organizational structures (Sim and Holt, 1998; Hilton and Begel, 2018).

Within the software engineering literature, most studies of onboarding focus on attracting and sustaining of new contributors in open source software communities (e.g. (Steinmacher et al., 2014; Gharehyazie et al., 2015)), and relatively limited attention to the onboarding process of software professionals. There are a few field studies on onboarding practices and techniques adopted by globally distributed companies such as Google (Johnson and Senges, 2010) and Microsoft (Begel and Simon, 2008a,b) (see Table 1 for an overview). While field studies provide rich contextual insights, the findings of such studies are inherently limited in generalizability (Stol and Fitzgerald, 2018)—the software engineering literature lacks tested theories that help explain what makes the onboarding process successful, and its potential influence on professionals' sense of what we term "organizational fit," and ultimately their intention to stay with or leave their organization. In this article, we draw on the wider literature from several research fields, including organizational, management and psychology literature, to synthesize a theoretical model to investigate this. The goal of this study is to develop insights regarding which factors might help to achieve a successful onboarding experience, how the onboarding experience relates to developers "settling in" in the organization, and how a lack of "fit" might increase people's intention to leave the organization.

The remainder of this article is organized as follows. In Sec. 2 we develop a theoretical model that informs our empirical study to test our theory—Sec. 3 presents details on our research strategy, followed by the study results in Sec. 4. Sec. 5 discusses our findings, limitations of the study, and concludes this article.

2. Theory Development

In this section we review prior work on onboarding in commercial software development organizations, and draw on management and psychology literature to develop a theoretical model. Table 1 provides an overview of previous studies of onboarding in a commercial software engineering context. As we focus specifically on onboarding in companies, this overview does not include the growing body of literature on onboarding in open source projects (cf. Steinmacher et al. (2014); Fagerholm et al. (2014b); Casalnuovo et al. (2015)). Neither does the table include studies that focus exclusively on specific activities such as knowledge transfer between senior and novice developers (cf. Viana et al. (2014))—while related, the focus of our study is specifically on onboarding.

2.1. Onboarding Activities

Onboarding of new employees involves a variety of activities. Van Maanen and Schein (1979), who refer to onboarding as *organizational socialization* described this as:

Table 1: Selection of prior empirical work on onboarding in commercial software development organizations

Study	Method	Findings
Berlin (1993)	Field study at HP Labs comprising observations of three expert-apprentice pairs and interviews to explore the complex ways in which experts help apprentices.	Learning new development environments and tools can be a major barrier to productivity. Experts tend to ask others for help more readily than apprentices, as the latter may not want to overburden their mentor.
Sim and Holt (1998)	Longitudinal field study that seeks to understand the naturalization process of "software immigrants," based on periodic interviews with 4 developers working on a compiler component system, at a very large computer company.	Set of seven observed patterns of the "naturalization process," organized in four categories, relating to (1) mentoring, (2) difficulties unrelated to the software system being learned, (3) first assignments, and (4) job fit. An example pattern observed is: *"mentors are an effective, though inefficient way to teach immigrants about the software system."*
Begel and Simon (2008a,b)	Field study comprising 85 hours of observation of eight recent college graduates during their first six months at Microsoft Corp., studying newcomers' activities, interactions, and challenges.	Subjects' strongest skills include the ability to write code, design specifications, and persistence when facing difficult (technical) problems. Subjects struggled with knowing when and how to ask questions, team interaction skills, technical difficulties related to tools and testing, organizing and managing a wide range of information, and 'orientation' issues in the project (tools, code, people).
Johnson and Senges (2010)	Field study employing case study methodology at Google, involving interviews with 24 stakeholders, observations, documents, addressing the question: how is practice-based learning incorporated in the onboarding process of new software engineers?	"Nooglers" (new Google employees) receive two week face-to-face training and orientation on core technologies and practices; senior engineers share engineering values and language; online training including checklists, tutorials and 'codewalks'; on-the-job training includes a starter project. Performance feedback is given on activities, objectives, and also through code reviews.
Dagenais et al. (2010)	Qualitative survey analyzed with a Grounded Theory approach involving 18 newcomers from 18 projects at IBM, investigating: what are the key, prominent features in a project landscape, what orientation obstacles do new team members face, and what orientation aids can be provided?	Newcomers must learn the project landscape, with interactions and challenges in five categories: product (incl. design, technologies); processes and practices (incl. tools), team (incl. roles and expertise), documentation, and context (incl. inter-team organization). Newcomers settle into a project landscape through early experimentation, internalization of structures and cultures, and progress validation (feedback).
Rastogi et al. (2015)	Field study of 8 large projects at Microsoft, using quantitative data from a version control system and qualitative data from 4 interviews with developers to investigate the ramp-up journey of newly hired developers.	Lack of documentation, getting set up (i.e. access and permission, system set-up), and lack of technical skills inhibit productivity. Mentorship was highlighted to be important by managers.
Pham et al. (2017)	Sample study using three online questionnaires (with 54, 170, and 698 respondents); 22 interviews with developers, to investigate practitioners' perceptions of novice developers.	Software practitioners perceive a skill gap between university graduates and industry expectations in relation to testing skills. Training and mentoring efforts are expended to address this gap.
Britto et al. (2018)	Field study of three globally distributed legacy projects involving teams based in India, Norway, Poland, Sweden, and the USA, investigating onboarding and associated challenges.	Onboarding strategies vary across companies and even among different sites within the same company. Onboarding newcomers onto projects with legacy code is challenging when original developers are not onsite. Orientation was neglected in all three case projects. Coaching and mentoring are most prevalent practices, but this reduces mentors' productivity.
Yates et al. (2019)	Field study using a Grounded Theory methodology of onboarding sessions across eight different organizations.	Experts describe the code from four different viewpoints: a structural, an algorithmic, a rationale, and a temporal view. Onboarding sessions facilitate the transfer of knowledge that cannot be found in the code or documentation.

> the process by which one is taught and learns 'the ropes' of a particular organizational role. In its most general sense, organizational socialization is then the process by which an individual acquires the social knowledge and skills necessary to assume an organizational role.

In terms of the "social knowledge and skills" that Van Maanen and Schein (1979) refer to, we focus specifically on three types of activities which are recurrent themes in prior studies (see Table 1): orientation, involving introducing a newcomer to the organization (Begel and Simon, 2008a,b; Britto et al., 2018); training, which focuses on providing sufficient information to a newcomer to do their job (Begel and Simon, 2008a,b; Berlin, 1993; Johnson and Senges, 2010; Pham et al., 2017; Rastogi et al., 2015); and support, which involves a set of mechanisms to help, guide, and provide feedback to a newcomer (Dagenais et al., 2010; Rastogi et al., 2015; Sim and Holt, 1998).

Van Maanen (1978) characterized organizational socialization along six dimensions, two of which are relevant here. The first is collective vs. individual socialization. In a collective process, a group of newcomers are subjected to a *"common set of experiences together"* (Van Maanen and Schein, 1979), whereas in an individual process, each newcomer has a unique experience. The second dimensions is formal vs. informal. A

formal socialization process is one in which newcomers are separated from other employees as they are subjected to a program specifically tailored to them. An informal process does not differentiate newcomers, and in such cases newcomers are expected to "learn on the job." The remaining four dimensions describe other aspects, such as whether the process is sequential vs. random and fixed vs. variable. As we did not draw on these dimensions in this study, we do not discuss these here.

2.1.1. Orientation

Joining a new workplace comes with its share of stress and anxiety for a newcomer (Bourne, 1967; Van Maanen and Schein, 1979); this initial stress and anxiety may inhibit software professionals from becoming productive. Orientation programs should include emotion-focused methods, along with problem-focused methods, to reduce stress. Klein and Weaver (2000) defined orientation programs for newcomers to introduce them to their job, co-workers, and the larger organization. Most orientation programs cover the following three areas: (1) terms and conditions of employment, (2) health, safety and legal issues, and (3) the organization's history, culture, and values (Wanous and Reichers, 2000). Orientation, also termed *"early socialization,"* typically takes place within the first month of an employee joining an organization (Wanous and Reichers, 2000; Anderson et al., 1996).

There are three accepted frameworks which guide the research and design of orientation programs: (1) stress theory / coping methods, (2) attitude theory / change, and (3) Realistic Job Preview (RJP) theory. Different industries use one or a more of these to develop orientation programs for their newcomers (Wanous and Reichers, 2000).

Early studies of orientation programs showed few significant correlations with long-term success factors, such as job satisfaction and organizational commitment, self-efficacy, and coping ability (Anderson et al., 1996; Bolles, 2000; Louis et al., 1983; Nelson and Quick, 1991; Saks, 1995). However, these programs remain one of the most popular methods for early socialization across organizations (Feldman, 1989) because they help convey compliance requirements and promote a positive image of the organization (Fan et al., 2012). Several studies have demonstrated the benefits of attending formal orientation programs (Gundry, 1993; Klein and Weaver, 2000; Wesson and Gogus, 2005).

Building on this previous research, we argue there is a positive link between the organization of orientation activities and what Bauer (2010) has defined as "onboarding success," namely knowledge of the organizational culture, role clarity, self-efficacy, and social integration. Further, orientation programs have also been linked to a reduced level of stress in newcomers. We hypothesize that:

HYPOTHESIS 1 (H1). *Orientation programs for newly recruited software professionals have a positive association with onboarding success.*

2.1.2. Training

Whereas *orientation* is concerned with *"context performance"* (focusing on the organization and its culture), *training* focuses on *"task performance"* (focused on the tasks that the new recruit is expected to perform) (Wanous, 1992; Wanous and Reichers, 2000). An additional difference is that orientation tends to occur when a person joins an organization, whereas training could happen at any stage during one's organizational tenure (Wanous and Reichers, 2000). Job training for software professionals is a well-researched area; previous studies have focused on different types of roles in software such as developers (Johnson and Senges, 2010), testers (Pham et al., 2015), volunteers in open source communities (Panichella, 2015; Canfora et al., 2012; Sarma et al., 2016), and software security (Papanikolaou et al., 2011). These studies highlight the importance of training newcomers and suggest tools and best practices (Panichella, 2015; Cherry et al., 2004; Sarma et al., 2016).

One of Van Maanen (1978)'s six dimensions of organizational socialization is the level of formality (formal vs. informal). A formal socialization process is one in which newcomers are separated from other employees as they are subjected to a program specifically tailored to them. An informal process does not differentiate newcomers, and in such cases newcomers are expected to "learn on the job."

Many studies highlight a prevalent skill gap of newly hired employees who have recently graduated (Byrne and Moore, 1997; McGuire and Randall, 1998; Lethbridge, 1998, 2000; Tang et al., 2001; Brechner, 2003; Surakka, 2007; Tesch et al., 2008; Lee and Fang, 2008; Simmons and Simmons, 2010; Radermacher and Walia, 2013; Pham et al., 2017). Technical support, software installation, information management, and maintenance of computer devices or components are the tasks performed by most IT graduates (Legier et al., 2013). Three broad categories of training can be identified: (1) practice-based learning (PBL) (Johnson and Senges, 2010), (2) class-based learning, and (3) mentoring (Casado-Lumbreras et al., 2011), which are briefly summarized below. Other training techniques include online training, tool based training (Panichella, 2015; Cherry et al., 2004), and task curation (Sarma et al., 2016). Organizations may follow one or a combination of these practices to train their newly recruited software professionals.

PBL is a work-centered learning methodology (Johnson and Senges, 2010), which is rooted in Lave and Wenger (1991)'s general theory of legitimate peripheral participation, which attempts to create an environment conducive to growth and innovation within the organization. Brown and Duguid (2000) described practice-based learning as creating organizational conditions where newcomers learn techniques of software development practices by watching their fellow colleagues. It is carefully integrated as part of the normal job of an employee and thus rendered invisible. In terms of Van Maanen (1978)'s dimensions of socialization, this corresponds to *individual* and *informal* organizational socialization.

Mentoring, involving connecting a newcomer (the mentee) and a more experienced senior colleague (the mentor), is one of the most common ways of transferring knowledge (Casado-Lumbreras et al., 2011). The relationship between the mentor and the mentee has been identified as one of the most important relationship in a person's professional career. Apart from transferring skills and knowledge, mentors also provide moral

support to their protégés. Mentors have a two-fold responsibility towards their mentees: career development and psycho-social support. Career development involves accustoming the newcomer towards the necessary knowledge and skills to succeed in the job. This is also referred to as *"cross pollination"* (Bauer, 2010). This includes, for example, transferring knowledge about a programming language, framework, or methodology. This can be done through one-on-one interactive sessions, coaching, providing exposure, or giving challenging assignments. Psycho-social modeling is the informal aspect of mentorship. It involves being a role model from whom the mentee can seek personal guidance, acceptance, counseling, and friendship.

Fagerholm et al. (2014a) found that active mentoring of new developers correlated with higher levels of activity, suggesting a higher level of productivity. Based on an interview study with software engineers, Enes (2005) found that this technique was more successful than classroom-based teaching, with respondents indicating that formal and organized teaching courses did not provide adequate application-domain knowledge.

Whatever the training mechanisms, the main aim of training is to ensure that a newcomer can perform the tasks of the job he or she is recruited to do. We suggest that training is positively linked to onboarding success, as defined above. Hence, we hypothesize that:

Hypothesis 2 (H2). *Training programs for newly recruited software professionals are positively associated with onboarding success.*

2.1.3. Support

The transition that new employees undergo to become a fully functioning employee is not a set of *discrete* steps. Rather, it is a *continuous process* that starts with orientation and training, and is achieved through a continuous system of providing support and feedback to the newcomer. Although university curricula teach students the basic principles and concepts of software engineering, SE is a field where newcomers must continuously learn new skills and technologies, such as new programming paradigms, languages, and techniques (Begel and Simon, 2008a). Other skills include the capability to create and debug specifications, documenting code, understanding and following a software development process, managing projects, and working within a team. These are skill gaps that are usually overlooked in university curriculum (Byrne and Moore, 1997; Tang et al., 2001; Surakka, 2007) and even a company-trained newcomer may have difficulty in understanding these concepts. Therefore, a support framework should be provided in the workplace, so that newcomers can discuss their challenges and doubts with seniors or colleagues without feeling embarrassed or weak. For example, if a newcomer is having difficulty in understanding a piece of code written by others who have left the organization, he or she should not feel embarrassed to ask a colleague or supervisor for help. We argue that the presence of support mechanisms, such as the availability of help, appreciation for a lack of newcomers' knowledge, feedback, and the ability to discuss personal issues that might affect performance, will positively link to newcomers' onboarding experience—or, what we call *onboarding success*. This in turn may lead to a better adjustment to the new setting and an increase in self-efficacy. Hence, we propose the following hypothesis:

Hypothesis 3 (H3). *Offering support mechanisms to newly recruited software professionals is positively associated with onboarding success.*

2.2. Onboarding Success and Organizational Fit

Thus far, we have focused on activities that organizations can organize and provide to newcomers in order to help improve those newcomers' onboarding experience. If that experience is positive, newcomers will feel comfortable, accepted, and confident to do their job. These activities—onboarding, training, support—are usually provided in the first stage of employment at an organization. In the longer term, organizations will be interested in whether employees will stay with the organization. Studies have demonstrated that an individual's *"socialization trajectory"* and becoming part of the core group in a team takes time and effort. Studies of open source communities suggest that only a small part of the peripheral group of newcomers, who have their performance recognized, are valued and eventually successful in their roles (Ducheneaut, 2005; Fang and Neufeld, 2009). Qureshi and Fang (2011) suggested that the lead time for different newcomers to attain a core status within the work group may vary. This is why onboarding is also referred to as *"organizational socialization"* (Bauer, 2010). Following previous studies that have linked effective onboarding to job satisfaction (Klein et al., 2006; Cable et al., 2013; Lavigna, 2009; Snell, 2006), we propose the following hypothesis:

Hypothesis 4 (H4). *Software professionals' degree of onboarding success is positively associated with job satisfaction.*

Further, new employees who are adequately socialized and have effective relations with their peers will feel more adapted to their new job demands, have an improved level of self-efficacy, and have a stronger attachment to the organization leading to greater organizational commitment (Bauer et al., 2007). Fisher (1985) found that sixty percent of employees consider strained relationships with their peers as the reason for failed onboarding. We therefore argue that, besides job satisfaction, which refers to contentment with the position, successful onboarding experience is also linked to good relationships in the workplace. Hence, we hypothesize:

Hypothesis 5 (H5). *Software professionals' degree of onboarding success is positively associated with the quality of their workplace relationships.*

We refer to these two characteristics of (1) being content with the job (job satisfaction) and (2) having good relationships within the workplace as an employee's "organizational fit." Together, these characteristics reflect a person's "fit" with the job and his or her fit within the social environment of the workplace.

2.3. Organizational Fit and Turnover Intention

Turnover intention is defined as a conscious and deliberate willingness to leave an organization (Tett and Meyer, 1993).

Ajzen (1991)'s Theory of Planned Behavior suggests that people act according to their intentions and perceptions of control over their behavior (Lenberg et al., 2017). Despite the very strong relationship between intended behavior and actual behavior, it is worth noting that this is not a perfect relationship. Lee and Mitchell (1994) suggest that employees may not follow through with their intention to leave until a precipitating "shock" event occurs, such as a reorganization or being assigned a new manager. It is also likely that acting upon intentions may rely on the availability of other concrete and more exciting opportunities.

As briefly pointed out, numerous studies have demonstrated a positive link between effective onboarding and job satisfaction and a negative link to turnover intention (Klein et al., 2006; Cable et al., 2013; Lavigna, 2009; Snell, 2006). Employee turnover has been the topic of extensive research—Hom et al. (2017) present a concise overview of one hundred years of research.

Much of the research on employee turnover consists of long-term studies; that is, these studies provide insights as to the turnover decisions of long-term employees of organizations (Holtom et al., 2008). Further, this relationship is debatable when it comes to IT professionals. For example, Gupta et al. (2018)'s large-scale survey of newcomers in five industrial sectors found that turnover intention was the highest in the IT sector. Gupta et al. (2018) also found that newcomers with a high level of self-efficacy (an indicator of successful onboarding (Bauer, 2010)) showed a higher level of turnover intention. The study suggested that employees with higher self-efficacy are more confident in their ability to switch over to a different job.

Given that the IT sector regularly expresses concerns about a shortage of talent and the high cost of recruiting new staff (incurred partly due to the productivity delay mentioned in Sec. 1),[1] it is worthwhile investigating this relationship for software developers. Hence, we posit the following two hypotheses, linking a newcomer's organizational fit to a reduced intention to leave his or her organization:

Hypothesis 6 (H6). *Job satisfaction is negatively associated with turnover intention.*

Hypothesis 7 (H7). *Workplace relationship quality is negatively associated with turnover intention.*

Professionals who have recently started with an organization are unlikely to have any intention to leave that organization within a very short time. We argue that the perceived onboarding experience will not immediately correlate to turnover intention, but, instead, that there are long-term mechanisms at work, specifically job satisfaction and workplace relationship quality, two characteristics of what we have labeled "organizational fit." Job satisfaction is not a state of being that appears immediately after joining an organization—it is a sense of comfort and happiness that emerges over time. Likewise, workplace relationships (i.e. relationships with peers, managers, and friendships with colleagues) do not form immediately, but rather develop over time. Thus, job satisfaction and workplace relationship quality are longer-term phenomena, which we argue in this study, subsequently negatively correlate with an intention to leave the organization. In other words, these two factors mediate the relationship between onboarding success and turnover intention. Hence, we propose the following hypothesis:

Hypothesis 8 (H8). *Job satisfaction and workplace relationship quality mediate the relationship between onboarding success and turnover intention.*

Fig. 1 presents the full theoretical model (we note that there is no standard notation for mediated relationships, hence the dotted box and line for H8, which refers to the mediating role of job satisfaction and workplace relationship quality on turnover intention).

3. Research Design

In order to evaluate our theoretical model, we conducted a cross-sectional survey, targeting software professionals with any length of experience. We conducted a cross-sectional survey rather than a survey within one specific organization, as this would more likely provide the requisite variety in responses that is necessary to evaluate a theoretical model such as ours. We selected Partial-Least Squares Structural Equation Modeling (PLS-SEM) to analyze the data. PLS-SEM is suitable to conduct exploratory theory-development studies (Hair et al., 2011, 2016; Russo and Stol, 2019). A well known alternative SEM approach is covariance-based SEM (CB-SEM), which is more suitable for confirmatory research and tends to require larger sample sizes (Hair et al., 2016). The remainder of this section proceeds as follows. Of particular importance is that the constructs used in the hypotheses are well defined and operationalized. Hence, we first discuss the measurement model. We then discuss data collection and analysis procedures.

3.1. Measurement Model

The theoretical model comprising the eight hypotheses are based on a number of *constructs*, or so-called *latent variables*. A latent variable cannot be directly measured or observed, but instead is measured through a set of *indicators* or *manifest variables*. In our model, all constructs are "reflective" (as opposed to "formative"). Any change in a reflective construct is said to be "reflected" in its indicators. That is, if the construct changes (which cannot be directly measured or observed), it will "cause" changes in its indicators, which *can* be measured.

Defining the constructs of studies such as ours is particularly important given their latent (unobservable) nature, and links directly to the issue of construct validity, which is concerned with the question: does the researcher measure what she intends to measure? A potential issue is that different studies may operationalize a certain construct differently by defining different indicators. Further, particular care must be given to the issue of construct clarity, so as to be able to clearly define and distinguish related, but different constructs.

[1] Reports appear regularly in the news, for example: `https://www.forbes.com/sites/forbesbusinessdevelopmentcouncil/2018/06/29/the-real-problem-with-tech-professionals-high-turnover/\#4738d0aa4201`

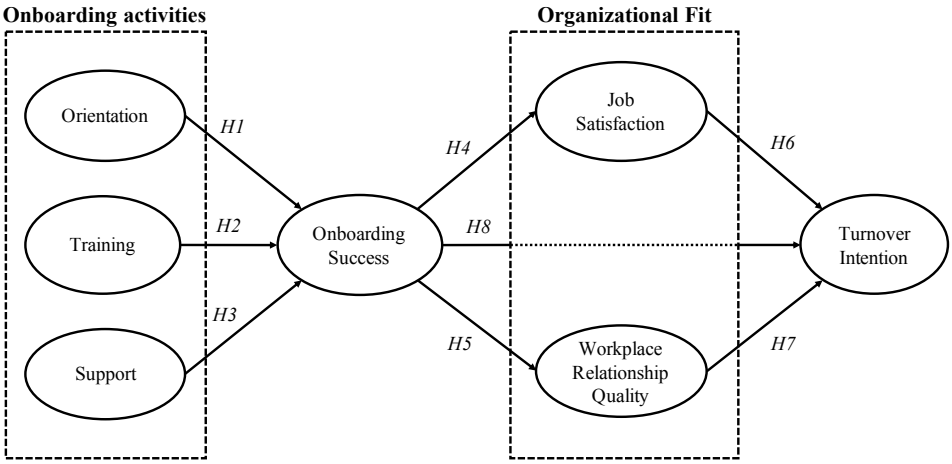

Figure 1: Research Model

We define the constructs of our model below, indicating what we mean by each construct, and through which indicators we measured them. We adopted and tailored as needed existing measurement instruments that have previously been used and validated. Each construct had between two and six indicators, resulting in a survey instrument of 30 questions. All indicators (questions) were measured using a five-point Likert scale (ranging from 1=Strongly Disagree to 5=Strongly Agree). The complete survey instrument is available in Appendix A; below we summarize the origins of the instruments to measure each construct.

- **Orientation**. Orientation comprises activities organized by an organization that is specifically targeted at newcomers and usually within a relatively short time frame after newcomers' entry to the organization. Orientation was measured using six items adopted from Louis et al. (1983) and Wanous and Reichers (2000). Items included attendance of orientation programs, awareness of organizational rules and policies, and assignment of a "buddy" or mentor, and items related to the organization organizing icebreaker events to facilitate meeting new colleagues, as well as team activities.

- **Training**. Training is concerned with professionals' task performance, and thus this refers to specific activities to ensure that newcomers can perform their tasks. We developed a new instrument to measure software professionals' training experience with four items based on prior literature. We captured formal training for their job role (either a classroom based training (Casado-Lumbreras et al., 2011) or one-on-one training from a senior/mentor (Panichella, 2015)). Training on internal systems and operating practices was adopted from Gupta et al. (2018) without tailoring as it was directly applicable to software professionals. The remaining two items were developed targeting specifically software professionals, considering training in specific tools and methods, and the availability of a point of contact or portal during training.

- **Support**. Support refers to the extent to which an organization helps newcomers in the onboarding process. We derived an instrument with four items based on work by Bauer (2010) and Gupta et al. (2018). The items measure the availability of a senior or mentor to ask for help when the newcomer is stuck with a given task, the extent to which newcomers feel weak or embarrassed to ask for help, the extent to which supervisors provide constructive feedback, and the extent to which newcomers feel that they can discuss personal issues when these affect their performance.

- **Onboarding Success**. We define onboarding success as the extent to which a newcomer feels they are comfortable in their new position. Bauer (2010) defines short-term outcomes that reflect a successful onboarding experience: knowledge of organizational culture, role clarity, self-efficacy, and social integration; each of these is an item in our instrument. Further, Bauer et al. (2007) suggest that role ambiguity (as opposed to role clarity) is a source of dissatisfaction associated with stress, and in a more general sense, joining a new workplace comes with a certain level of stress (Bourne, 1967; Van Maanen and Schein, 1979). Hence, we captured the absence of stress as a fifth item to reflect the construct onboarding success as defined above.

- **Job Satisfaction**. Job satisfaction has been studied extensively in the management literature. Spector (1985) developed an instrument of 36 questions to measure job satisfaction. We found that some of the questions were overlapping, so we selected six indicators that are appropriate indicators of the construct job satisfaction.

- **Workplace Relationship Quality**. We define workplace relationship quality as a person's perceived quality of his or her relationships within the workplace. We developed an instrument with three items to measure this construct. Sias (2005) studied two types of workplace relationships: supervisor-subordinate and peer co-worker relationships.

We defined one indicator for each of these types of relationships for the workplace relationship quality construct. Korte and Lin (2013) suggested that newcomers not only seek acceptance into the group, but also friendship (affect). Several others have studied friendships in the workplace (Sias and Cahill, 2013; Rawlins, 1992). Hence, we defined a third item to capture workplace friendships.

- **Turnover Intention**. Turnover intention is defined as a respondent's inclination to leave their organization. We adopted a two-item instrument from Metcalf et al. (2015), which queries (1) whether respondents think of quitting their job frequently, and (2) whether respondents plan to search for a new job within the next 12 months.

3.2. Data Collection and Analysis

We implemented our survey instrument with SurveyMonkey, and advertised it through a number of channels. We distributed the survey through our professional network; contacts were also invited to share the link with their colleagues. We also shared the link on Twitter; this microblogging platform provides analytical data about the tweet, indicating the tweet was retweeted 20 times, and the link was clicked 30 times. The total number of "engagements" (a metric defined by Twitter) was 72—this is the number of Twitter users that the tweet was presented to. While the survey was in principle anonymous, we did offer respondents an option to enter their email address voluntarily if they were interested in receiving the results of the study; a number of respondents were subsequently sent a preliminary report with some of the findings. We did not capture any geographical data or information about respondents' employers. We received 102 complete responses that could be used for analysis. While sample size is commonly an issue for covariance-based SEM (CB-SEM) due to the requirements for its computation, PLS can also be used with more modest sample sizes. A common minimum threshold for a sample is the so-called "10 x" rule, which states that the sample should be a minimum of ten times the largest number of structural paths directed at a latent construct (Hair et al., 2011)—in our study, the maximum number of structural paths to a latent construct is three, suggesting a sample size of only 30. This rule of thumb has been criticized, in particular when such calculations leads to small sample size recommendations; our sample size is more than three times this number. Within the software engineering domain, studies using PLS-SEM commonly use samples between 50 and 100 (Parolia et al., 2013, 2015; Vijayasarathy and Turk, 2012).

We also conducted a power analysis, using the freely available tool G*Power (Faul et al., 2009).[2] Given the exploratory nature of this study, we used the threshold value for a medium effect size (0.15 (Cohen, 1988)), a significance level of 0.05, and a default value for the power $(1 - \beta)$ of 0.8 (Marcoulides and Saunders, 2006). The maximum number of predictors is 3 in our model. This calculation indicated a minimum sample size of 77, well below our sample of 102. Using a higher value for power of 0.9 yielded a minimum sample of 99.

Table 2 presents aggregate information about the respondents' roles, and Table 3 presents the total number of years of work experience, and the number of years in their current role. Most respondents identified as a software developer/engineer, analyst, or software tester. A small number identified as a project manager or director. Not all respondents provided this information. In terms of experience, 59 respondents (58%) indicated they had 0-3 years total work experience, but 78 reported to be in their current role only for 0-3 years. Almost all respondents were in their current role for up to 7 years; only 5 reported to be in the current role for 7+ years.

A variety of PLS-SEM software packages is available; we used the software package SmartPLS version 3.2.8 for the analyses, the results of which are presented in Sec. 4. The analysis procedure for PLS-SEM consists of two main steps, with several tests and procedures in each step. The first step is to evaluate the *measurement model*, which empirically assess the relationships between the indicators and the constructs. The results of this step are presented in Sec. 4.1. The second step is to evaluate the *theoretical* or *structural model* which represents the set of hypotheses—thus, at is in this step that the hypotheses are evaluated. We present the results of the second step in Sec. 4.2.

4. Results

4.1. Evaluation of the Measurement Model

Before the structural model can be evaluated, we evaluate the measurement model. We discuss the internal consistency reliability, convergent validity, and discriminant validity.

Table 2: Respondents' roles

Role	Count
Software developer/engineer	41
Analyst	16
Software tester	16
Technical support	6
Technical consultant	6
Researcher	6
Project manager	4
Web designer	4
Other	19

Table 3: Respondents' total work experience, and experience in current role

	0-3 yr.	3-7 yr.	7+ yr.
Total experience	59	20	23
Experience in current role	78	19	5

[2] G*Power has a wide range of tests; we used the following settings: Test family: F-tests; statistical test: linear multiple regression with fixed model, R^2 deviation from zero; type of power analysis: a priori analysis.

Table 4: Internal Consistency Reliability

Construct	Cronbach α	Composite Reliability	Average Variance Extracted (AVE)
Orientation	0.709	0.882	0.602
Training	0.761	0.847	0.583
Support	0.716	0.823	0.539
Onboarding success	0.774	0.846	0.525
Workplace relationship quality	0.738	0.848	0.651
Job Satisfaction	0.834	0.882	0.602
Turnover intention	0.714	0.871	0.772

4.1.1. Internal Consistency Reliability

Internal consistency reliability assesses how well the different indicators are able to measure the constructs *reliably* and *consistently*. There are several tests to measure this. We performed the Cronbach's alpha and Composite Reliability tests. Cronbach's alpha tests generally show lower values of reliability and are more conservative compared to composite reliability, which sometimes overestimates the values (Hair et al., 2016). The true measure of internal consistency reliability lies between the lower bound of Cronbach's alpha and upper bound of composite reliability. For exploratory research, values of 0.6-0.7 are acceptable, while for research in a more advanced stage values between 0.7 and 0.9 are recommended (Hair et al., 2016). Values below 0.6 suggest a lack of internal consistency reliability, whereas values over 0.95 suggest that indicators are too similar and therefore not desirable. Table 4 shows that the Cronbach alpha and CR for all fall within the range 0.7-0.9.

4.1.2. Convergent Validity

Convergent validity measures how well different indicators of a construct correlate positively with one another. All constructs in our model are reflective (not formative), which means that indicators are considered to be different ways to measure the same construct—they should share a considerable proportion of variance, which means that they *converge*. To assess convergent validity, two metrics are important: the Average Variance Extracted (AVE), and the outer loadings of a construct's indicators. The AVE values should be at least 0.5, and Table 4 shows that all AVE values are all above that threshold.

Table 5 reports the outer loadings of all items. A standard rule of thumb is that outer loadings should be higher than 0.70 (Hair et al., 2016), but this does not mean that values below this threshold should always be removed. If outer loadings are between 0.40 and 0.70, the effect of removing them on the AVE should be considered—if the AVE improves significantly, the items should be removed, but otherwise they can be retained. In this case, we found that AVE values of all the constructs, except orientation, were above the desired threshold of 0.50. Hence, two indicators of the orientation construct were removed, leading to an improvement of the AVE for orientation from 0.416 to 0.527. After their removal, all outer loadings were above 0.65 except one (js_5, importance of work), which had a negative effect on discriminant validity (discussed below); hence, we also removed this indicator, leaving all outer loadings with values higher than 0.659.

4.1.3. Discriminant Validity

Discriminant validity refers to the extent to which the different constructs capture different phenomena or concepts. In other words, it is a measure of how *distinct* each construct is in relation to other constructs. It implies that each construct is unique and represents characteristics that are not measured by other constructs. There are three common ways to assess discriminant validity. First, we investigated the cross-loadings: the outer loadings of a construct's indicators should be higher for that construct than on any of the other constructs. If an indicator would load higher onto a different construct than the one that it purportedly measures, then that suggests the indicator is a better measure for that other construct. Table 5 shows that (inspecting row by row) all constructs' indicators load highest onto their respective constructs.

The second approach to assess discriminant validity is evaluating the Fornell-Larcker criterion. This criterion states that the square root of a construct's AVE should be higher than that construct's correlation with other constructs. In plain terms, a construct should share more variance with its own indicators than with other constructs. We observed that the AVE value for onboarding success was slightly lower (approx. 0.01) than the correlation with job satisfaction; we resolved this by removing item js_5 as mentioned above. Table 6 lists the construct correlations, with the square roots of the AVE values on the diagonal. All square roots of the AVE values comply with the Fornell-Larcker criterion.

Third, we also considered Henseler's Heterotrait-Monotrait (HTMT) ratio (Henseler et al., 2015) (see Table 7). The cut-off value is 0.9 beyond which discriminant validity could be considered problematic (Henseler et al., 2015), though some researchers recommend a more conservative cut-off of 0.85 (Hair et al., 2016). Table 7 shows that most HTMT ratios are below 0.85, with only two ratios between 0.85 and 0.9 (onboarding success/support, and job satisfaction/support). Besides these cut-off values, HTMT ratios should also be significantly different from 1.0; this can be calculated using a bootstrap procedure, which calculates bias-corrected confidence intervals for all ratios. (We

Table 5: Cross loadings of the retained indicators on the constructs (A complete list of the items is available in Appendix A)

Item	Item Description	Orientation	Training	Support	Onboarding Success	Job Satisfaction	Social Acceptance	Turnover Intention
or_3	Buddy/mentor assigned to help	**0.730**	0.345	0.398	0.334	0.462	0.157	−0.205
or_4	Made aware of challenges	**0.660**	0.492	0.205	0.264	0.302	0.203	−0.136
or_5	Icebreakers to meet colleagues	**0.724**	0.494	0.144	0.315	0.272	0.219	−0.150
or_6	Org. has team days / activities	**0.784**	0.282	0.307	0.470	0.484	0.505	−0.258
tr_1	Formal training program	0.269	**0.764**	0.112	0.211	0.194	0.114	−0.094
tr_2	Internal system training	0.423	**0.808**	0.200	0.276	0.370	0.096	−0.188
tr_3	Training in technology, methods	0.434	**0.809**	0.139	0.280	0.307	−0.008	−0.195
tr_4	Point of contact / online portal	0.431	**0.664**	0.144	0.313	0.294	0.104	0.124
su_1	Can seek help if stuck	0.264	0.241	**0.662**	0.444	0.383	0.270	−0.124
su_2	Not embarrassed asking for help	0.217	0.089	**0.741**	0.441	0.356	0.292	−0.270
su_3	Supervisor ongoing feedback	0.185	0.155	**0.742**	0.482	0.379	0.256	−0.133
su_4	Speak to supervisor if personal issues affect performance	0.392	0.113	**0.787**	0.609	0.654	0.549	−0.302
os_1	Joining new workplace less stressful	0.346	0.464	0.430	**0.662**	0.455	0.246	−0.032
os_2	Familiarity with organization's culture	0.388	0.305	0.336	**0.659**	0.526	0.176	−0.137
os_3	Understand role's responsibilities	0.336	0.332	0.534	**0.730**	0.564	0.447	−0.270
os_4	Confidence in capability to do job	0.332	0.110	0.497	**0.749**	0.463	0.380	−0.255
os_5	Socially integrated in workplace	0.397	0.165	0.622	**0.811**	0.546	0.587	−0.315
js_1	Fair chance on promotion	0.487	0.320	0.498	0.626	**0.845**	0.358	−0.395
js_2	Org. offers growth opportunities	0.398	0.380	0.490	0.626	**0.845**	0.358	−0.395
js_3	Fair compensation	0.376	0.348	0.318	0.441	**0.692**	0.278	−0.102
js_4	Achievements recognized	0.337	0.203	0.461	0.478	**0.763**	0.315	−0.328
js_6	I am satisfied with my job	0.474	0.297	0.603	0.603	**0.858**	0.409	−0.438
rq_1	Professional relations with peers	0.348	0.188	0.416	0.532	0.435	**0.871**	−0.191
rq_2	Friends in the workplace	0.245	0.029	0.199	0.305	0.209	**0.759**	−0.081
rq_3	Professional relations with seniors	0.368	−0.014	0.498	0.408	0.411	**0.785**	−0.254
ti_1	I frequently think of quitting	−0.300	−0.106	−0.323	−0.345	−0.407	−0.193	**0.923**
ti_2	Will look for jobs within next year	−0.148	−0.078	−0.160	−0.144	−0.271	−0.219	**0.832**

Table 6: Fornell-Larcker criterion: correlations among the constructs and square roots of the AVE values (on diagonal)

Construct	1	2	3	4	5	6	7
1. Orientation	**0.726**						
2. Training	0.526	**0.763**					
3. Support	0.371	0.199	**0.734**				
4. Onboarding success	0.494	0.363	0.683	**0.724**			
5. Workplace relationship quality	0.406	0.099	0.485	0.535	**0.807**		
6. Job Satisfaction	0.539	0.392	0.623	0.705	0.460	**0.776**	
7. Turnover intention	−0.269	−0.107	−0.289	−0.297	−0.230	−0.396	**0.879**

discuss the bootstrap procedure in more detail below.) None of these included the value 1.0, indicating that all HTMT ratios were acceptable. Based on these three tests to assess discriminant validity, we conclude that the discriminant validity of our study is satisfactory.

4.2. Evaluation of the Structural Model

We now turn our attention to the evaluation of the structural model, which includes the evaluation of the hypotheses.

Table 7: Heterotrait Monotrait (HTMT) ratios

Construct	1	2	3	4	5	6	7
1. Orientation							
2. Training	0.731						
3. Support	0.503	0.271					
4. Onboarding success	0.644	0.483	0.884				
5. Workplace relationship quality	0.500	0.199	0.603	0.641			
6. Job Satisfaction	0.671	0.487	0.765	0.869	0.552		
7. Turnover intention	0.342	0.273	0.372	0.366	0.304	0.466	

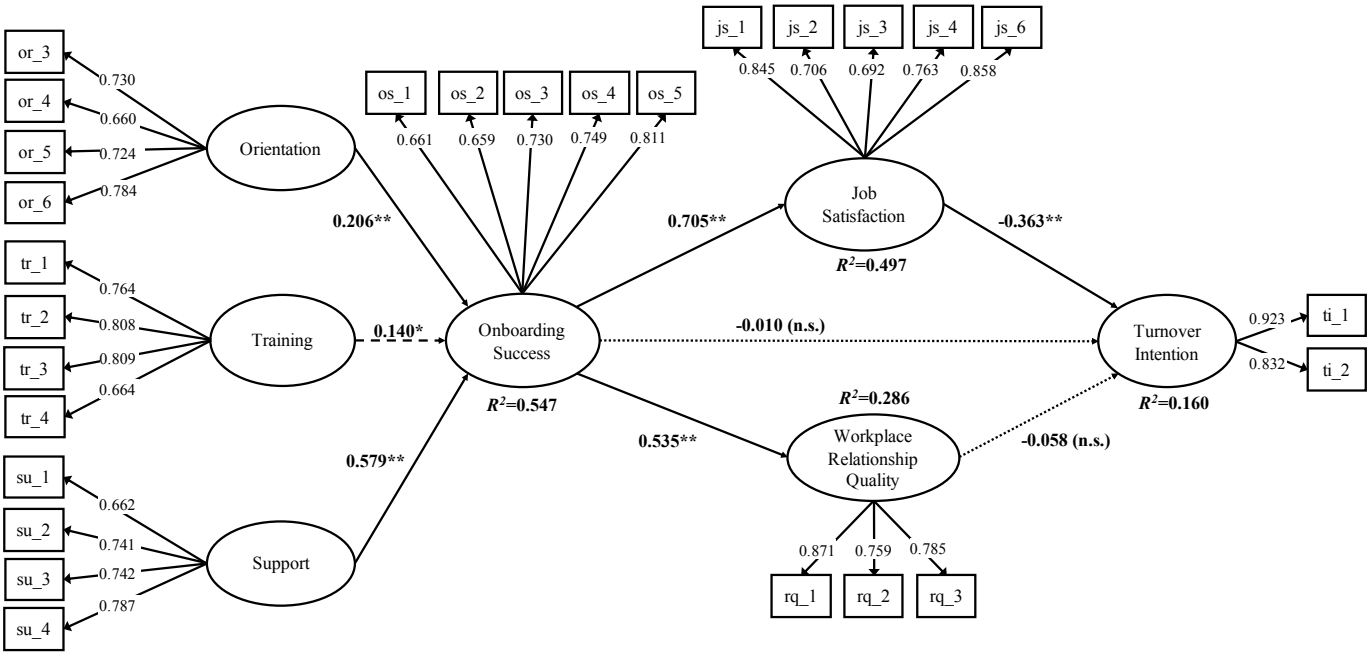

Figure 2: Outer loadings (measurement model) and path coefficients (structural model) (* $p < 0.10$ indicated by a dashed line, ** $p < 0.05$). Non-significant links are indicated with a dotted line.

Table 8: Coefficients of determination

Construct	R^2	Q^2
Onboarding success	0.547	0.244
Job satisfaction	0.497	0.266
Workplace relationship quality	0.286	0.165
Turnover intention	0.160	0.089

Table 9: Effect sizes for all constructs (with specified endogenous variable to which the effect size applies)

Construct	f^2
Orientation	0.061
Training	0.031
Support	0.638
Onboarding success	
Job satisfaction	0.989
Workplace relationship quality	0.401
Turnover intention	0.000
Workplace relationship quality	0.003
Job satisfaction	0.077

4.2.1. Assessing Collinearity

Our theoretical model consists of seven constructs, of which three are exogenous (orientation, training, and support). To ensure that the exogenous constructs are independent, we evaluate their collinearity by means of the Variance Inflation Factor (VIF). A widely accepted cut-off value for the VIF is 5 (Hair et al., 2011), though collinearity issues may also occur with VIF values between 3 and 5, which is why Hair et al. (2019) recommend a cut-off value of 3. In our model, all VIF values

are below this more conservative cut-off value between 1.1 and 2.3, indicating that there are no collinearity issues in our model.

4.2.2. Path Coefficients and Significance

PLS does not make any assumptions about the distribution (such as a Normal distribution) of the data; therefore, it cannot use any parametric tests of significance. In order to determine whether path coefficients are statistically significant, PLS packages depend on a bootstrapping procedure. This involves drawing a large number (typically five thousand) of random subsamples with replacement; replacement is needed to ensure that all subsamples contain the same number of observations as the original data set. For each subsample, the PLS path model is estimated; these sets of coefficients form a bootstrap distribution, which can be considered as an approximation of the sampling distribution (following the Central Limit Theorem). From this, a standard error and standard deviation are determined (Hair et al., 2016), which can subsequently be used to make statistical inferences. Table 10 shows the results for our eight hypotheses, including the mean of the bootstrap distribution, the standard deviation, the 95% confidence interval, and the p values. Based on these results, we found support for Hypotheses 1, and 3-6. We only found weak support for Hypothesis 2 (p=0.064). Hypothesis 7 was not supported (p=0.663).

Hypothesis 8 proposed that job satisfaction and workplace relationship quality mediate the link between onboarding success and turnover intention. To evaluate mediating relationships, we must compare the *indirect* paths suggested by the mediators, to the *direct* paths (Zhao et al., 2010; Nitzl et al., 2016). Variables may have *no* mediating effect (indirect effect is insignificant), a *partial* mediating effect (if the direct effect is significant), or a *full* mediating effect (if the direct effect is insignificant).

Table 2 shows that the *direct* association between onboarding success and turnover intention is not significant (p=0.954). On the other hand, the indirect association between onboarding success and turnover intention with job satisfaction as a mediator was statistically significant ($p < 0.02$). The association between onboarding success and turnover intention mediated by workplace relationship quality was not statistically significant (p=0.673). These findings lend support to our hypothesis that job satisfaction fully mediates the relationship between onboarding success and turnover intention, but workplace relationship quality does not. Sec. 5 presents a discussion of these results.

4.2.3. Coefficient of Determination and Effect Sizes

This stage of analysis helps assess the relationship between constructs and the predictive capabilities of the model. The R^2 values of the four endogenous variables in our model are listed in Table 8. R^2 values range between 0 to 1, with values of 0.75, 0.50, or 0.25 considered substantial, moderate, or weak respectively (Hair et al., 2016; Henseler et al., 2009). We found moderate values for onboarding success and job satisfaction, but only a weak result for workplace relationship quality and turnover intention.

The value of R^2 will change when an exogenous construct is removed from the model—a measure for the extent to which the exogenous construct contributes to an endogenous construct's R^2 value is the f^2 effect size. Table 9 lists the f^2 values for each of the constructs. An effect size below 0.02 means that there is no effect of the exogenous construct on the endogenous construct. The threshold values 0.02, 0.15, and 0.35 refer to, respectively, small, medium, and large effects (Cohen, 1988). The table shows that of the three constructs associated with onboarding success, support makes the largest contribution to the R^2 with an f^2 effect size of approximately 0.64. Orientation and training have rather small effects with effect sizes of 0.03 and 0.06, respectively.

Finally, we also inspected Stone-Geisser's Q^2 value which is an indicator of the model's predictive relevance (Hair et al., 2016) (these can be obtained through a blindfolding procedure; Hair et al. discuss this in detail). Q^2 values are calculated only for reflective endogenous constructs—a value larger than 0 indicates the construct has predictive relevance. The same thresholds as for the R^2 apply to Q^2 (0.02. 0.15, 0.35). Table 8 shows that all Q^2 values are greater than zero, suggesting our model has predictive relevance as well.

5. Discussion and Conclusion

5.1. Discussion of Results

In this study, we have drawn on the wider literature beyond software engineering, including management and psychology literature, to derive a theoretical model for studying onboarding success of software professionals, and its relation to their inclination to leave their job. Evaluation of the theoretical model helps us understand the role of a number of activities that have been shown to be important in onboarding *in general* but not yet within the software development domain. Our analysis highlights a number of key findings and implications—Table 11 provides a summary.

We found support for *H1* which proposed a positive association between orientation and onboarding success. Orientation typically happens during the first days or weeks of an employee joining an organization, and involves giving out essential information about the rules and policies of the company and helping newcomer interaction. Of the six indicators, we retained four after inspecting the outer loadings and the AVE. The items that were removed reflected the extent to which respondents attended an orientation program, and whether they were made aware of organizational rules and policies. The retained indicators all refer to "social" aspects, including a buddy to help people settle in, dealing with challenges, and icebreaker activities as well as social gatherings; the removed indicators both referred to more "objective" (rather than opinionated) aspects, such as awareness of rules and regulations.

We found only weak support for *H2* ($p < 0.10$), suggesting that training of newcomers only has a marginal link to onboarding success. Task-oriented training can have a direct effect on role clarity and self-efficacy, which are indicators of onboarding success—this in turn may help to address the skills gap mentioned in Sections 1 and 2. This might also explain training in internal systems and operating practices (environment), and tools/technology used for the job have the highest loadings on the training construct (both over 0.8). Our study has focused

Table 10: Path coefficients, bootstrap estimates, standard deviations, confidence intervals, and p values

Hypothesis	Coefficient	Bootstrap mean	Std.dev.	95% CI	p
H1: Orientation → Onboarding success	0.205	0.211	0.080	(0.024, 0.347)	0.010
H2: Training → Onboarding success	0.140	0.156	0.076	(–0.029, 0.265)	0.064
H3: Support → Onboarding success	0.579	0.576	0.068	(0.439, 0.702)	0.000
H4: Onboarding success → Job satisfaction	0.705	0.706	0.062	(0.548, 0.801)	0.000
H5: Onboarding success → Workplace relationship quality	0.535	0.546	0.079	(0.354, 0.671)	0.000
H6: Job satisfaction → Turnover intention	−0.363	−0.366	0.140	(–0.591, –0.020)	0.010
H7: Workplace relationship quality → Turnover intention	−0.058	−0.055	0.133	(–0.306, 0.223)	0.663
Mediators					
H8:					
Direct effect:					
Onboarding success → Turnover intention	−0.010	−0.014	0.170	(–0.325, 0.344)	0.954
Indirect effects:					
Onboarding success → Job satisfaction → Turnover Intention	−0.256	−0.257	0.062	(–0.437, –0.026)	0.017
Onboarding success → Workplace relationship quality → Turnover Intention	−0.031	−0.028	0.073	(–0.170, 0.125)	0.673

on a rather narrow meaning of training to facilitate a clear and precise definition and measurement. Recently, Baltes and Diehl (2018) have presented a more holistic treatment of the notion of developer expertise.

H3 proposed a positive association between support and onboarding success—this hypothesis is also strongly supported by our study. Support can be considered a more continuous process, and ongoing support will make newcomers at ease when seeking help from seniors and peers regarding professional and personal issues, without invoking feelings of being judged or embarrassed. We found that support provided to newcomers is the largest contributor to onboarding success (standardized path coefficient of ca. 0.58). The effect size (f^2=0.638) analysis also shows that the support construct makes a large contribution to the coefficient of determination (R^2) of onboarding success in our model. Over 80 percent of respondents felt they could ask for help from a senior in matters related to the job tasks, and over 70 percent indicated that they would not feel weak or embarrassed in doing so. This suggests that an organizational culture that encourages transparency and helping others can help newcomers to onboard successfully. Newcomers initially may have many questions about their new job, and having a personal point of contact to address these questions will help to ease into the new work environment.

We found support for *H4* and *H5*, which suggested that onboarding success (which tends to be a short-term outcome, as it takes place within the first few months) is positively associated with job satisfaction and workplace relationship quality. Both job satisfaction and relationship quality are "longer-term," effects, in that both are *emergent* perceptions; people need time before they can reflect on their relationship with their job (job satisfaction) and relationships with colleagues (with peers, managers, and creating friendships) also take time to shape. Although most respondents showed high levels of job satisfaction, the level of satisfaction with remuneration was relatively low compared to other indicators. Eighty-six percent and 89 percent of respondents said they had good personal and professional relationships with their colleagues, respectively. This might be a result of either the efforts of the organization by organizing social events or through their university social network—the latter is a possibility because almost 58 percent of the respondents are recent graduates with less than three years of professional work experience (see Table 2).

Job satisfaction was found to have a negative link to turnover intention (*H6*), implying that people who are happy with their job are less likely to leave their employer. The quality of their relationships within the workplace (*H7*), on the other hand, did not seem to have a significant link to turnover intention: people may have very good relationships with their peers, managers, and even have friends within the workplace, this does not seem to stop people from considering to leave their position.

Finally, we argued that job satisfaction and workplace relationship quality mediate the relationship between onboarding success and turnover intention (*H8*). We only found partial support for this. We found support for the mediating role of job satisfaction, suggesting that a successful onboarding experience (a "short-term" experience, soon after recruitment) may help to achieve job satisfaction (a longer-term state of contentment),

Table 11: Summary of results and implications

Hypothesis	Findings	Implications
H1: Orientation → onboarding success	Supported. While the standardized path coefficient was moderate (0.2), the orientation program does contribute to onboarding success, though the effect size is low (0.06).	Orientation activities have a moderate correlation with onboarding success. Organizations should leverage the opportunity to give newcomers a good introduction to help people settle in.
H2: Training → onboarding success	Weak support. Training has a low standardized path coefficient (0.14) with a p value of 0.06, meaning it is not statistically significant following the standard alpha level of 0.05. Small effect size (0.03).	Training does not seem to be helpful towards successfully onboarding new recruits. It is likely that the knowledge required for a given job is highly specific, and too much to cover during a short-term formalized training program. Organizations might do better through catering for on-the-job training of new recruits.
H3: Support → onboarding success	Supported. Support was found to be the largest and most significant factor associated with onboarding success, with a standardized path coefficient of 0.58, and an effect size of 0.64.	Providing ongoing support to new recruits so that they feel supported in their job is likely to be most important in getting new staff to settle in. Organizations should create an environment where people feel supported and safe to ask for help.
H4: Onboarding Success → Job Satisfaction	Supported. Onboarding success has a considerable (standardized path coefficident of 0.7) and statistically significant positive association with job satisfaction.	Ensuring the onboarding process is successful may be key to achieving a high level of job satisfaction. Easing newcomers into the new job so as to integrate them is key when designing onboarding programs.
H5: Onboarding Success → Workplace Relationship Quality	Supported. Onboarding success has a considerable (standardized path coefficient of over 0.5) and statistically significant positive association with workplace relationship quality.	Professionals who perceive their onboarding experience to be successful also have good relationships within the workplace. They "fit in" socially, which may reduce conflicts in the workplace, which should be of interest to organizations; however, studies are needed to explore this further.
H6: Job Satisfaction → Turnover intention	Supported. Job satisfaction has a statistically significant and considerable (standardized path coefficient of -0.36) inverse association with turnover intention. Hence, software professionals who are content in their job are less likely to leave the organization.	This study finds support for this hypothesis in the software domain. Given the high cost of turnover in this domain, it may be of interest to conduct studies that explore what dissatisfies professionals so that organizations can take countermeasures in order to ensure that job satisfaction levels remain high.
H7: Workplace relationship quality → Turnover intention	Not supported. Having good relationships with peers and managers does not associate with a lower intention to leave the organization (standardized path coefficient < 0.06, $p > 0.6$).	While having good relationships within the workplace, perhaps even having friends, may be good for productivity but it bears no effect on a professional's intention to stay with or leave the organization. Organizations may still want to take measures to improve workplace relationships for other reasons (e.g. productivity), but it does not help to retain staff.
H8: Job satisfaction and workplace relationship quality mediate onboarding success → turnover intention	Partially supported. Job satisfaction fully mediates the relationship between onboarding success and turnover intention (standardized path coefficient -0.25, $p = 0.017$), but workplace relationship quality does not (-0.031, $p = 0.67$). No direct effect from onboarding success to turnover intention (-0.01, $p = 0.95$).	Job satisfaction plays a key role in achieving a "short-term" organizational socialization (onboarding) and "longer-term" retaining of staff (i.e. low turnover). Having good workplace relationships does not. Organizations that wish to focus on retaining staff should take measures to *"naturalize"* new recruits as well as take measures that staff are content with their job in the longer run. Social activities may help reduce friction in day-to-day conflicts, this will not help in retaining staff.

which in turn may help to retain staff. Workplace relationship quality, on the other hand, is not a mediator; while we found that a successful onboarding experience was positively associated with the quality of workplace relationships (*H5*), this did not translate to a lower turnover intention—having already established that no support was found for *H7*, this is not surprising.

5.2. Limitations of this Study

5.2.1. Construct Validity

We adopted and tailored existing measurement instruments, and developed derived measurement instruments for some constructs based on prior literature. Our analysis of the measurement model confirmed that our constructs were internally consistent, and scored satisfactory on convergent and discriminant validity tests. We defined a new construct called workplace relationship quality; we did not identify a construct in prior literature that represents the quality of relationships within a workplace. Though newly defined, it scored well on the tests mentioned above.

5.2.2. Internal Validity

This study is a sample study rather than an experimental study (i.e., we made no interventions), and drawing causal relationships is typically not possible (Stol and Fitzgerald, 2018). Our hypotheses propose associations between different constructs rather than causal relationships. While it is clear that activities such as orientation and training tend to occur at an early stage of an employee's tenure and that job satisfaction tends to emerge over time (i.e., after orientation and training), we cannot exclude the possibility that other factors are at play. Two of four coefficients of determination (R^2) can be considered moderate at values over 0.5; thus, while other factors are likely to play a role, these results represent a useful starting point for future studies.

5.2.3. External Validity

This survey was conducted online and anonymously, and thus we cannot report any details on the extent to which our sample was representative. The nature of our sample is a convenience sample, which we contacted through our professional networks and through social media. However, in our study we sought to get responses from software professionals in general—we deem it unlikely that people not active in the software industry would have completed the survey. Table 2 shows that our population sample represents a variety of software professionals; hence we argue that the sample served our study goal, namely to seek evidence for our theoretical model focused on software professionals. The responses were sufficiently consistent to find full empirical support ($p < 0.05$) for five of eight hypotheses, weak support for one ($p < 0.10$), and partial support for another (*H8*). However, we suggest further studies to replicate our findings.

5.3. Conclusion and Future Work

Designing a successful onboarding program is a key part of any organization's talent management and retention strategy. As previously discussed, IT professionals show one of the highest levels of turnover intention compared to employees in other industries. While several field studies exist, the software engineering literature lacks theoretical models that help organizations understand which factors play a role in achieving success in the onboarding process, and how this might translate to longer-term "organizational fit" as manifested by job satisfaction and workplace relationships, and ultimately, a reduced level of turnover.

This study sought to develop such a theoretical model specifically targeting the software engineering domain, which may inform the development of onboarding processes for software professionals. Drawing from the literature on onboarding from other disciplines as well as studies of onboarding in the software engineering literature, we derived a theoretical model comprising eight hypotheses. We evaluated these hypotheses through a sample study, using an online survey instrument. Based on 102 responses, we found (partial) support for six hypotheses.

The strongest statistical significance was found for the association between support and onboarding success. Support is a *continuous* rather than a *discrete* time-bound activity, and can be easily overlooked; rather than "pushing" information and training onto newcomers, new recruits "pull" support from senior colleagues and are offered a safe environment to ask questions without evoking a sense of embarrassment. Our study shows that support is key to a newcomer's "organizational socialization" (Bauer, 2010) or "naturalization" (Sim and Holt, 1998). Software organizations are increasingly growing and operating in ever more complex and diverse settings. Technologies are evolving constantly, and new methods and practices emerge continuously—with methods in use becoming irrelevant. Few other industries see such continuous change and evolution as the IT industry. This could be why an environment that provides constant support and feedback to its employees is indispensable to assist its employees in keeping up-to-date. Orientation and training, on the other hand, had only modest associations with onboarding success. Hence, one of the two most important recommendations from this study is not only emphasizing the importance of a supporting environment, but also the relatively insubstantial contribution of potentially expensive orientation and training programs to the success of onboarding programs.

Besides onboarding activities, our study also considers what we have termed "organizational fit" of a new recruit, which refers to how content that person is with the job, and how well the person fits in socially. We found that successful onboarding experiences correlate positively with both these aspects, and so a good onboarding experience helps to establish a person's organizational fit. Of the two aspects, only job satisfaction had a negative association with turnover intention; the other aspect, a person's relationships within the workplace, did not. Further, we found that job satisfaction mediates the relationship between onboarding success and turnover intention; whereas a successful onboarding experience does not associate with a lower turnover intention, an indirect association does exist through the job satisfaction construct.

Based on our findings, we suggest a number of avenues for future research.

While job satisfaction seems to play a major role in a software professional's decision to stay or leave an organization,

the reality in the IT sector faces significant challenges in staff retention. Future work might explore which other factors play a role in a decision to leave an organization.

The results of our study emphasize the importance of support—future work might explore effective ways that organizations can offer this support to software professionals, which likely must be tailored to different roles within the software industry.

The results of this study suggest a positive link between onboarding success and job satisfaction, but it is highly likely that this relationship may change over time; that is, the sense of onboarding success might 'fade,' and other factors might start to impact software professionals' job satisfaction in more significant ways.

Whereas our study did not find any benefit of having good workplace relationships (in terms of turnover intention), it is likely that staff who have good relationships within the workplace are happier, which some studies suggest will benefit productivity (Graziotin et al., 2018). It is worth exploring how good workplace relationships can be established (aside from having a successful onboarding experience), and which effects this might have on software professionals. In this context, it is also worth noting that our study did not distinguish between *internal* and *external* turnover intention; that is, poor workplace relationships may not encourage people to leave the organization (i.e., external turnover), but it might encourage them to join a different team or unit within the same organization (i.e., internal turnover).

Furthermore, future research could also investigate software professionals' character attributes, personality, attitudes, and beliefs in relation to the relationships they forge within the workplace. It is likely that these factors affect how software professionals' perceive the value of workplace relationships, and therefore it is of interest to evaluate whether or not these different attitudes and personal values relate to an intention to leave the organization.

The software industry's landscape is fast-moving, with many start-ups, acquisitions, and mergers. When a company is acquired by larger ones, its staff are effectively hired *wholesale*. Our model has not considered this option.

To conclude, this study contributes to the relatively limited literature within the software engineering domain on onboarding of software professionals. While considerable attention has been paid to onboarding in open source communities in recent years, most software that is developed remains to be closed source. It is our hope that this study offers a good starting point for future work.

Acknowledgment

We are grateful to the anonymous reviewers who offered generous and constructive feedback which has led to dramatic improvements of this manuscript. This work was supported, in part, by Science Foundation Ireland grant 15/SIRG/3293 and 13/RC/2094 and co-funded under the European Regional Development Fund through the Southern & Eastern Regional Operational Programme to Lero—the Irish Software Research Centre (www.lero.ie).

Appendix A. Survey Instrument

This sample study was conducted via an online survey implemented with SurveyMonkey. The full survey instrument is listed below. Items prefixed with a (*) were dropped due to poor loading onto their constructs (see Sec. 3 for details).

Orientation

- or_1 (*) I attended an orientation program with other new hires
- or_2 (*) I was made aware of the organizational rules and policies
- or_3 I was assigned a buddy/mentor to help me settle in my job
- or_4 I was made aware of the challenges/difficulties I may face in my job and how I should cope with them
- or_5 There were activities (like ice breakers) organized where I could interact with my new colleagues and seniors
- or_6 My company organizes team days / social gatherings to help socialize with my colleagues

Training

- tr_1 I attended a formal training program tailored for my job role (A formal training program may involve class room based training as a group or one-on-one training from a senior)
- tr_2 I received training to understand the internal systems and operating practices to perform my job. (Operating practices could be methodologies like Agile Programming and Extreme Programming)
- tr_3 I was specifically trained in the technology/tools that I used for my job
- tr_4 I had a point of contact/online portal that I could use if I had any faced any difficulties regarding training

Support

- su_1 If I am stuck at some task and cannot find a way through I can ask my senior/supervisor/mentor for help
- su_2 I will not feel weak or embarrassed to ask for help in the above scenario
- su_3 My supervisor provides me with ongoing constructive feedback about my performance
- su_4 I can speak to my supervisor if any personal issues are affecting my performance at work

Onboarding Success

- os_1 The initial orientation program helped me feel less stressful about joining a new workplace
- os_2 I got a good idea about the organizational culture during my onboarding
- os_3 I clearly understand the expectations and responsibilities of my job
- os_4 I am confident that I am capable of excelling in my job
- os_5 I can say I am socially integrated in my workplace

Workplace Relationship Quality

- rq_1 I have good professional relations with my peers
- rq_2 I am good friends with some of my colleagues

rq_3 I have good relations with my seniors

Job Satisfaction

js_1 I feel that I am given a fair chance of being promoted
js_2 My work is helping my professional growth by developing my skills and learning new technologies/tools/practices
js_3 I feel I am being given a fair compensation of the work that I am asked to do
js_4 My performance and achievements are recognized and appreciated by my senior
js_5 (*) I think that the work I am asked to do at my job is important and meaningful
js_6 I would say that I am satisfied with my job

Turnover Intention

ti_1 I frequently think of quitting
ti_2 I will be actively looking for a new job within the next one year

References

Aasheim, C.L., Williams, S., Butler, E.S., 2009. Knowledge and skill requirements for IT graduates. Journal of Computer Information Systems 49, 4853.

Ajzen, I., 1991. The theory of planned behavior. Organ Behav Hum Decis Process 50, 179–211.

Anderson, N.R., Cunningham-Snell, N.A., Haigh, J., 1996. Induction training as socialization: Current practice and attitudes to evaluation in british organizations. International Journal of Selection and Assessment 4, 169–183.

Baltes, S., Diehl, S., 2018. Towards a theory of software development expertise, in: Proceedings of the 26th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering, ACM. pp. 187–200.

Bauer, T., Erdogan, B., Bodner, T., Truxillo, D., Tucker, J., 2007. Newcomer adjustment during organizational socialization: a meta-analytic review of antecedents, outcomes, and methods. Journal of Applied Psychology 92.

Bauer, T.N., 2010. Onboarding new employees: Maximizing success. SHRM Foundations Effective Practice Guideline Series .

Begel, A., Simon, B., 2008a. Novice software developers, all over again, in: 4th International Workshop on Computing Education Research, pp. 3–14.

Begel, A., Simon, B., 2008b. Struggles of new college graduates in their first software development job. ACM SIGCSE Bulletin 40, 226–230.

Berlin, L., 1993. Beyond program understanding: A look at programming expertise in industry, in: Proceedings of the Fifth Workshop on Empirical Studies of Programmers, pp. 6–25.

Bolles, B.E., 2000. International student training and orientation: Current trends in methods of programming. Ph.D. thesis. Colorado State University.

Bourne, P.G., 1967. Some observations on the psychosocial phenomena seen in basic training. Psychiatry 30, 187–196.

Brechner, E., 2003. Things they would not teach me of in college: what microsoft developers learn later, in: Companion of the 18th annual ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications (OOPSLA '03), ACM. pp. 134–136.

Britto, R., Cruzes, D.S., Šmite, D., Sablis, A., 2018. Onboarding software developers and teams in three globally distributed legacy projects: A multi-case study. J. Softw. Evol. Proc. 30, 1–17.

Brooks, F.P., 1975. The mythical man-month. Addison-Wesley.

Brown, J.S., Duguid, P., 2000. Organizational learning and communities of practice: Toward a unified view of working, learning, and innovation, in: Knowledge and communities, pp. 99–121.

Byrne, D.J., Moore, J.L., 1997. A comparison between the recommendations of computing curriculum 1991 and the views of software development managers in ireland. Comput. Educ. 28, 145–154.

Cable, D.M., Gino, F., Staats, B.R., 2013. Reinventing employee onboarding. MIT Sloan Manage. Rev. 54, 23.

Canfora, G., Di Penta, M., Oliveto, R., Panichella, S., 2012. Who is going to mentor newcomers in open source projects?, in: Proceedings of the 20th International Symposium on the Foundations of Software Engineering, ACM. pp. 44:1–44:11.

Casado-Lumbreras, C., Colomo-Palacios, R., Soto-Acosta, P., Misra, S., 2011. Culture dimensions in software development industry: The effects of mentoring. Scientific Research and Essays 6, 2403–2412.

Casalnuovo, C., Vasilescu, B., Devanbu, P., Filkov, V., 2015. Developer onboarding in github: the role of prior social links and language experience, in: Proceedings of the 2015 10th joint meeting on foundations of software engineering, ACM. pp. 817–828.

Cherry, J., Arrieta, M., Brown, E., Ramaswamy, S., 2004. An interactive visualization tool for understanding complex programs. Software Engineering Research and Practice , 49–56.

Cohen, J., 1988. Statistical power analysis for the behavioral sciences. 2nd ed.

Dagenais, B., Ossher, H., Bellamy, R., Robillard, M., de Vries, J., 2010. Moving into a new software project landscape, in: Proceedings of the 32nd International Conference on Software Engineering, pp. 275–284.

DeMarco, T., Lister, T., 1987. Peopleware: Productive Projects and Teams. Dorset House Publishing.

Dit, B., Revelle, M., Gethers, M., Poshyvanyk, D., 2013. Feature location in source code: a taxonomy and survey. Journal of software: Evolution and Process 25, 53–95.

Ducheneaut, N., 2005. Socialization in an open source software community: A socio-technical analysis. Computer Supported Cooperative Work (CSCW) 14, 323–368.

Enes, P., 2005. Acquiring and Sharing Expert Knowledge. Master's thesis. Norwegian University of Science and Technology.

Fagerholm, F., Guinea, A., Münch, J., Borenstein, J., 2014a. The role of mentoring and project characteristics for onboarding in open source software projects, in: Proceedings of the 8th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM), pp. 55:1–55:10.

Fagerholm, F., Guinea, A.S., Borenstein, J., Münch, J., 2014b. Onboarding in open source projects. IEEE Software 31, 54–61.

Fan, J., Buckley, M.R., Litchfield, R.C., 2012. Orientation programs that may facilitate newcomer adjustment: A literature review and future research agenda, in: Research in personnel and human resources management. Emerald Group Publishing Limited, pp. 87–143.

Fang, Y., Neufeld, D., 2009. Understanding sustained participation in open source software projects. Journal of Management Information Systems 25, 9–50.

Faul, F., Erdfelder, E., Buchner, A., Lang, A.G., 2009. Statistical power analyses using g*power 3.1: Tests for correlation and regression analyses. Behavior Research Methods 41, 1149–1160.

Feldman, D.C., 1989. Socialization, resocialization, and training: Reframing the research agenda, in: Goldstein, I.L. (Ed.), Training and Development in Organizations. Jossey-Bass, pp. 376–416.

Fisher, C.D., 1985. Social support and adjustment to work: A longitudinal study. Journal of Management 11, 39–53.

Forrest, C., 2018. Software had the highest job turnover rate of any industry in 2017. Https://www.techrepublic.com/article/software-had-the-highest-job-turnover-rate-of-any-industry-in-2017/.

Gharehyazie, M., Posnett, D., Vasilescu, B., Filkov, V., 2015. Developer initiation and social interactions in oss: A case study of the apache software foundation. Empirical Software Engineering 20, 1318–1353.

Graziotin, D., Fagerholm, F., Wang, X., Abrahamsson, P., 2018. What happens when software developers are (un)happy. The Journal of Systems and Software 140, 32–47.

Gundry, L.K., 1993. Fitting into technical organizations: The socialization of newcomer engineers. IEEE Transactions on engineering management 40, 335–345.

Gupta, P.D., Bhattacharya, S., Sheorey, P., Coelho, P., 2018. Relationship between onboarding experience and turnover intention: intervening role of locus of control and self-efficacy. Industrial and Commercial Training 50, 61–80.

Hair, J., Risher, J., Sarstedt, M., Ringle, C., 2019. When to use and how to report the results of PLS-SEM. European Business Review 31, 2–24.

Hair, J.F., Hult, G.T.M., Ringle, C., Sarstedt, M., 2016. A primer on partial least squares structural equation modeling (PLS-SEM). Sage.

Hair, J.F., Ringle, C.M., Sarstedt, M., 2011. PLS-SEM: Indeed a silver bullet. Journal of Marketing theory and Practice 19, 139–152.

Hall, T., Beecham, S., Verner, J., Wilson, D., 2008. The impact of staff turnover on software projects: The importance of understanding what makes software

practitioners tick, in: Proceedings of ACM SIGMIS CPR, ACM. pp. 30–39.
Henseler, J., Ringle, C.M., Sarstedt, M., 2015. A new criterion for assessing discriminant validity in variance-based structural equation modeling. J. Acad. Mark. Sci. 43, 115–135.
Henseler, J., Ringle, C.M., Sinkovics, R.R., 2009. The use of partial least squares path modeling in international marketing, in: New challenges to international marketing. Emerald Group Publishing Limited, pp. 277–319.
Hertzum, M., Pejtersen, A.M., 2000. The information-seeking practices of engineers: searching for documents as well as for people. Information Processing & Management 36, 761–778.
Hilton, M., Begel, A., 2018. A study of organizational dynamics of software teams, in: Proceedings of the ACM/IEEE 40th International Conference on Software Engineering: Software Engineering in Practice, ACM. pp. 191–200.
Holtom, B.C., Mitchell, T.R., Lee, T.W., Eberly, M.B., 2008. Turnover and retention research: A glance at the past, a closer review of the present, and a venture into the future. The Academy of Management Annals 2, 231–274.
Hom, P.W., Lee, T.W., Shaw, J.D., Hausknecht, J.P., 2017. One hundred years of employee turnover theory and research. Journal of Applied Psychology 102, 530–545.
Johnson, M., Senges, M., 2010. Learning to be a programmer in a complex organization: A case study on practice-based learning during the onboarding process at google. Journal of Workplace Learning 22, 180–194.
Klein, H.J., Fan, J., Preacher, K.J., 2006. The effects of early socialization experiences on content mastery and outcomes: A mediational approach. Journal of Vocational Behavior 68, 96–115.
Klein, H.J., Weaver, N.A., 2000. The effectiveness of an organizational-level orientation training program in the socialization of new hires. Personnel Psychology 53, 4766.
Korte, R., Lin, S., 2013. Getting on board: Organizational socialization and the contribution of social capital. Human Relations 66, 407–428.
LaToza, T.D., Venolia, G., DeLine, R., 2006. Maintaining mental models: a study of developer work habits, in: 28th International Conference on Software Engineering, pp. 492–501.
Lave, J., Wenger, E., 1991. Situated learning: Legitimate Peripheral Participation. Cambridge University Press.
Lavigna, B., 2009. Getting onboard: Integrating and engaging new employees. Government Finance Review 25, 65–70.
Lee, S., Fang, X., 2008. Perception gaps about skills requirement for entry-level IS professionals between recruiters and students: An exploratory study. Inf. Resour. Manage. J. 21, 39–63.
Lee, T.W., Mitchell, T.R., 1994. An alternative approach: The unfolding model of voluntary employee turnover. Academy of Management Review 19, 51–89.
Legier, J., Woodward, B., Martin, N.L., 2013. Reassessing the skills required of graduates of an information systems program: An updated analysis. Information Systems Education Journal 11.
Lenberg, P., Tengberg, L.G.W., Feldt, R., 2017. An initial analysis of software engineers' attitudes towards organizational change. Empirical Software Engineering 22, 2179–2205.
Lethbridge, T.C., 1998. A survey of the relevance of computer science and software engineering education, in: Proceedings of the 11th Conference on Software Engineering Education and Training (CSEET '98), IEEE Computer Society. pp. 56–66.
Lethbridge, T.C., 2000. Priorities for the education and training of software engineers. The Journal of Systems and Software 53, 53–71.
Lethbridge, T.C., Singer, J., Forward, A., 2003. How software engineers use documentation: The state of the practice. IEEE software 6, 35–39.
Littman, D.C., Pinto, J., Letovsky, S., Soloway, E., 1987. Mental models and software maintenance. Journal of Systems and Software 7, 341–355.
Louis, M.R., Posner, B.Z., Powell, G.N., 1983. The availability and helpfulness of socialization practices. Personnel Psychology 36, 857–866.
Marcoulides, G.A., Saunders, C., 2006. Editor's comments: PLS: A silver bullet? MIS Quarterly 30, iii–ix.
McGuire, E.G., Randall, K.A., 1998. Process improvement competencies for IS professionals: a survey of perceived needs, in: (Ed.), R.A. (Ed.), Proceedings of the 1998 ACM SIGCPR conference on Computer personnel research (SIGCPR '98), ACM. pp. 1–8.
McMurtrey, M.E., Downey, J.P., Zeltmann, S.M., Friedman, W.H., 2008. Critical skill sets of entry-level IT professionals: An empirical examination of perceptions from field personnel. Journal of Information Technology Education: Research 7, 101–120.
Metcalf, A.Y., Stoller, J., Habermann, M., Fry, T., 2015. Respiratory therapist job perceptions: the impact of protocol use. Respiratory care 60.
Miller, J., 2018. Why do software engineers change jobs so frequently? Https://www.forbes.com/sites/quora/2018/02/06/why-do-software-engineers-change-jobs-so-frequently/.
Nelson, D.L., Quick, J.C., 1991. Social support and new- comer adjustment in organizations: Attachment theory at work? Journal of organizational behavior 12, 543–554.
Nitzl, C., Roldán, J., Cepeda, C., 2016. Mediation analysis in partial least squares path modeling: Helping researchers discuss more sophisticated models. Industrial Management and Data Systems 119, 1849–1864.
O'Brien, M., 2007. Evolving a Model of the Information-Seeking Behaviour of Industrial Programmers. Ph.D. thesis. University of Limerick.
Panichella, S., 2015. Supporting newcomers in software development projects, in: IEEE International Conference on Software Maintenance and Evolution (ICSME), pp. 586–589.
Papanikolaou, A., Karakoidas, V., Vlachos, V., Venieris, A., Ilioudis, C., Zouganelis, G., 2011. A hacker's perspective on educating future security experts, in: 5th Panhellenic Conference on Informatics with international participation (PCI 2011), pp. 68–72.
Parolia, N., Chen, J.V., Jiang, J.J., Klein, G., 2015. Conflict resolution effectiveness on the implementation efficiency and achievement of business objectives in IT programs: A study of IT vendors. Information and Software Technology 66, 30–39.
Parolia, N., Jiang, J.J., Klein, G., 2013. The presence and development of competency in IT programs. The Journal of Systems and Software 86, 3140–3150.
Pham, R., Kiesling, S., Singler, L., Schneider, K., 2017. Onboarding inexperienced developers: struggles and perceptions regarding automated testing. Software Quality Journal 25, 1239–1268.
Pham, R., Stoliar, Y., Schneider, K., 2015. Automatically recommending test code examples to inexperienced developers, in: 10th Joint Meeting on Foundations of Software Engineering, pp. 890–893.
Qureshi, I., Fang, Y., 2011. Socialization in open source software projects: A growth mixture modeling approach. Organizational Research Methods 14, 208–238.
Radermacher, A., Walia, G., 2013. Gaps between industry expectations and the abilities of graduates, in: Proceeding of the 44th ACM technical symposium on Computer science education (SIGCSE '13), ACM. pp. 525–530.
Rastogi, A., Thummalapenta, S., Zimmermann, T., Nagappan, N., Czerwonka, J., 2015. Ramp-up journey of new hires tug of war of aids and impediments, in: Proceedings of the ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM), pp. 96–105.
Rawlins, W., 1992. Friendship matters: Communication, dialectics, and the life course. Aldine de Gruyter.
Russo, D., Stol, K.J., 2019. Soft theory: a pragmatic alternative to conduct quantitative empirical studies, in: Proceedings of the Joint 7th International Workshop on Conducting Empirical Studies in Industry and 6th International Workshop on Software Engineering Research and Industrial Practice, IEEE Press. pp. 30–33.
Saks, A.M., 1995. Longitudinal field investigation of the moderating and mediating effects of self-efficacy on the relationship between training and newcomer adjustment. J. Appl. Psychol. 80.
Sarma, A., Gerosa, M., Steinmacher, I., Leano, R., 2016. Training the future workforce through task curation in an oss ecosystem, in: Proceedings of the 2016 24th ACM SIGSOFT International Symposium on Foundations of Software Engineering, pp. 932–935.
Sias, P., Cahill, D., 2013. From coworkers to friends: The development of peer friendships in the workplace. Western Journal of Communication 62, 273–299.
Sias, P.M., 2005. Workplace relationship quality and employee information experiences. Communication Studies 56, 375–395.
Sillito, J., Murphy, G.C., Volder, K.D., 2008. Asking and answering questions during a programming change task. IEEE Transactions on Software Engineering 34, 434–451.
Sim, S.E., Holt, R., 1998. The ramp-up problem in software projects: A case study of how software immigrants naturalize, in: Proceedings of the Twentieth International Conference on Software Engineering, pp. 361–370.
Simmons, C.B., Simmons, L.L., 2010. Gaps in the computer science curriculum: an exploratory study of industry professionals. Journal of Computing Sciences in Colleges 25, 60–65.
Simon, D., Jackson, K., 2013. A closer look at information systems graduate

preparation and job needs: Implications for higher education curriculum enhancements. World Journal of Education 3, 52–62.
Snell, A., 2006. Researching onboarding best practice: using research to connect onboarding processes with employee satisfaction. Strategic HR Review 5, 32–35.
Spector, P.E., 1985. Measurement of human service staff satisfaction: Development of the job satisfaction survey. American journal of community psychology 13, 693–713.
Steinmacher, I., Silva, M.A.G., Gerosa, M.A., 2014. Barriers faced by newcomers to open source projects: a systematic review, in: IFIP International Conference on Open Source Systems, p. 153163.
Stol, K.J., Fitzgerald, B., 2018. The ABC of software engineering research. ACM Trans Softw Engineer Methodol 27.
Surakka, S., 2007. What subjects and skills are important for software developers? Communications of the ACM 50, 73–78.
Tang, H.L., Lee, S., Koh, S., 2001. Educational gaps as perceived by IS educators: A survey of knowledge and skill requirements. Journal of Computer Information Systems 41, 76–84.
Tesch, D., Braun, G., Crable, E., 2008. An examination of employers' perceptions and expectations of IS entry-level personal and interpersonal skills. Information Systems Education Journal 6, 3–16.
Tett, R.P., Meyer, J.P., 1993. Job satisfaction, organizational commitment, turnover intention, and turnover: path analyses based on meta-analytic findings. Personnel psychology 46, 259–293.
Trent, R.H., 1988. Perspectives on the academic preparation of MIS professionals, in: Proceedings of the ACM SIGCPR conference on Management of information systems personnel (SIGCPR '88), ACM. pp. 119–119.
Van Maanen, J., 1978. People processing: Major strategies of organizational socialization and their consequences, in: Paap, J. (Ed.), New directions in human resource management. Prentice Hall, pp. 19–36.
Van Maanen, J.E., Schein, E.H., 1979. Toward a theory of organizational socialization, in: Staw, B. (Ed.), Research in organizational behavior. JAI Press. volume 1, pp. 209–264.
Viana, D., Conte, T., de Souza, C.R.B., 2014. Knowledge transfer between senior and novice software engineers: A qualitative analysis, in: Proceedings of the 26th International Conference on Software Engineering and Knowledge Engineering, pp. 235–240.
Vijayasarathy, L., Turk, D., 2012. Drivers of agile software development use: Dialectic interplay between benefits and hindrances. Information and Software Technology 54, 137–148.
Von Mayrhauser, A., Vans, A.M., Howe, A.E., 1997. Program understanding behaviour during enhancement of large-scale software. Journal of Software Maintenance: Research and Practice 9, 299–327.
Wanous, J.P., 1992. Organizational entry: Recruitment, selection, orientation, and socialization of newcomers. Prentice Hall.
Wanous, J.P., Reichers, A.E., 2000. New employee orientation programs. Human resource management review 10, 435–451.
Watkins, M., 2013. The first 90 days, updated and expanded: proven strategies for getting up to speed faster and smarter. Harvard Business Review Press .
Wesson, M.J., Gogus, C.I., 2005. Shaking hands with a computer: an examination of two methods of organizational newcomer orientation. Journal of Applied Psychology 90.
Yates, R., Power, N., Buckley, J., 2019. Characterizing the transfer of program comprehension in onboarding: an information-push perspective. Empirical Software Engineering in press.
Zhao, X., Lynch Jr., J.G., Chen, Q., 2010. Reconsidering Baron and Kenny: Myths and truths about mediation analysis. Journal of Consumer Research 37, 197–206.

Gaurav G. Sharma is a DevOps Engineer in Security Intelligence with IBM in Ireland. He was the co-founder of an e-commerce start-up called Pocket My Need in his final year of engineering. He has a keen interest in project management, automation, and DevOps methodologies. He received a M.Sc. in Computer Science from University College Cork, and a B.Eng. in Information Technology from the University of Mumbai. Contact him at gaurav.gyan.sharma@ibm.com

Dr Klaas-Jan Stol is a lecturer with the School of Computer Science and Information Technology at University College Cork. He is a Science Foundation Ireland (SFI) Principal Investigator, and an SFI Funded Investigator with Lero, the Irish Software Research Centre. His research focuses on contemporary software engineering strategies and research methodology. Stol received a Ph.D. from the University of Limerick and a M.Sc. from the University of Groningen. Contact him at klaas-jan.stol@lero.ie.