Towards a Theory of Software Developer Job Satisfaction and Perceived Productivity

Margaret-Anne Storey, *Member, IEEE*, Thomas Zimmermann, *Member, IEEE*,
Christian Bird, *Member, IEEE*, Jacek Czerwonka, *Member, IEEE*, Brendan Murphy, *Member, IEEE*,
Eirini Kalliamvakou, *Member, IEEE*

✦

Abstract—Developer satisfaction and work productivity are important considerations for software companies. Enhanced developer satisfaction may improve the attraction, retention and health of employees, while higher productivity should reduce costs and increase customer satisfaction through faster software improvements. Many researchers and companies assume that perceived productivity and job satisfaction are related and may be used as proxies for one another, but these claims are a current topic of debate. There are also many social and technical factors that may impact satisfaction and productivity, but which factors have the most impact is not clear, especially for specific development contexts. Through our research, we developed a theory articulating a bidirectional relationship between software developer job satisfaction and perceived productivity, and identified what additional social and technical factors, challenges and work context variables influence this relationship. The constructs and relationships in our theory were derived in part from related literature in software engineering and knowledge work, and we validated and extended these concepts through a rigorously designed survey instrument. We instantiate our theory with a large software company, which suggests a number of propositions about the relative impact of various factors and challenges on developer satisfaction and perceived productivity. Our survey instrument and analysis approach can be applied to other development settings, while our findings lead to concrete recommendations for practitioners and researchers.

Index Terms—software engineering management; empirical studies; software companies; theory

1 INTRODUCTION

Improving developer job satisfaction and productivity have been recognized as critical goals by many software companies [1], [2] and are a point of interest in recent company-led surveys (e.g., the Stack Overflow survey[1], or the GitLab survey[2]). More satisfied developers would allow companies to attract and retain talent, while more productive developers could help reduce costs, increase profits, and improve product quality. Retaining talent is especially important because high turnover introduces challenges with software quality when important knowledge is lost [3], [4]. Productivity is also impacted as new developers have to learn the project landscape and existing developers have to spend time training them [5].

Recent research reports on what makes developers feel more productive [6], what motivates them [7], and what affects their happiness [8]. Research from other disciplines has showed us how to distinguish between human-related constructs such as satisfaction, happiness, and motivation [9], [10]. Combined, this work enhances our understanding by indicating the things that could contribute to developers feeling satisfied or feeling productive. However, we lack an understanding of the *relationship* between these elements and how it could improve software engineering outcomes.

The research we present in this paper aims to understand and explain the relationship between job satisfaction and perceived productivity for software engineers. Our investigation was informed by seminal work in organizational psychology by Judge *et al.*, where job satisfaction is widely accepted to be positively correlated with work performance [10]. We build on this work to investigate which factors influence the relationship between developer satisfaction and perceived productivity in software engineering.

First, we look to *identify which factors* of their jobs make software engineers feel more satisfied, and how their overall job satisfaction and satisfaction with individual factors impact their perceived productivity. On the flip side, we also look to *identify which challenges* developers face that impact their job satisfaction. Finally, we aim to account for the role of context in development work; specifically we *investigate how experience level and time spent on various development activities impact developers' job satisfaction and perceived productivity*.

To develop our theory, we conducted a study in a large software company. The first author spent several months at the company observing and learning how different organizations within this company aimed to understand and measure developers' productivity and their job satisfaction. The existing efforts in this company revealed that job and engineering tool satisfaction were often used as proxies for perceived productivity. However, not enough was known on how much satisfaction and productivity are related and which other work factors may influence job satisfaction and perceived productivity. Following a literature review on these concepts and other factors that emerged over several months of observations on site, we developed and deployed multiple versions of a survey to derive, refine, confirm, and investigate factors that impact developer job satisfaction and perceived productivity. We iteratively developed our pool of candidate factors by taking into account related work in software engineering and other disciplines, as well as previous internal surveys conducted in the company under study. Through our study, our research provided answers to

---

1. https://insights.stackoverflow.com/survey/2017
2. https://page.gitlab.com/rs/194-VVC-221/images/ gitlab-enterprise-survey-2016-report.pdf

the following research questions:

RQ1: Which **social and technical factors** are important to developers (RQ1.1), what is their perceived relative importance (RQ1.2), and how satisfied are developers with these factors (RQ1.3)?

RQ2: Which **challenges** do developers experience (RQ2.1), how impactful are each of these challenges perceived by developers (RQ2.2), and how do these challenges impact satisfaction with the social and technical factors (RQ2.3)?

RQ3: How do the social and technical factors impact the relationship between overall job satisfaction and perceived productivity (RQ3.1), and what is the impact of **work context variables** on the relationship between job satisfaction and perceived productivity (RQ3.2)?

By answering these questions, we propose a generic *theory* to posit that (1) a variety of social and technical factors, and challenges, contribute to a bi-directional relationship between job satisfaction and perceived productivity, and (2) that which factors matter depends on a variety of work context variables. We then show how this theory can be instantiated (or scoped) to specific development contexts.

In the next section (Section 2), we provide some background on related work. Then we describe the survey methodology and its design, and our analysis approach in Section 3. We present the results to our research questions in Sections 4, 5, and 6. We present and discuss our proposed theory of satisfaction and productivity in Section 7 and discuss the implications of our findings and theory for practitioners and researchers in Section 8. We detail the limitations and threats to validity of our research in Section 9 and conclude the paper in Section 10.

2 BACKGROUND

Understanding productivity in software engineering has become an important topic of interest since improving it depends on many things [11]. Equally important is to understand its relationship to other factors that may affect productivity. In this paper, we look at the relationship between job satisfaction and perceived productivity, and we discuss research related to these concepts below. We have used this research combined with insights gained during a company on site visit, to generate a pool of factors that potentially impact job satisfaction and perceived productivity, and to inform our survey design. We provide details about the origin of all factors we used in Section 3.

2.1 Developer Satisfaction

Developer satisfaction has been discussed in conjunction with other human aspects of software engineering, such as developer happiness and developer motivation. We have reviewed work in these areas to include factors that may play a part in developer satisfaction in our study.

Happiness in software development has recently been studied in depth [12]. The focus has been on understanding the factors that cause happiness (or unhappiness) when software engineers are developing software [8], as well as the corresponding consequences on the outcome of development [12]. We have, therefore, included factors in our study that are informed by the work on developer happiness.

For example, Graziotin et al. [8] report that lacking skill by coworkers can be a source of unhappiness for developers, and we included *"skilled coworkers"* as one of the factors potentially influencing work satisfaction in our study.

Although it is reasonable to assume that happiness and satisfaction are related, they are distinct constructs. As Wright and Cropanzano [9] point out, researchers frequently use the term happiness to refer to psychological well-being, which refers to one's life as a whole (among other characteristics). In our research, we focus on satisfaction with aspects of one's job, differentiating our construct from happiness. Therefore, we align with Wright and Cropanzano's [9] definition of job satisfaction (citing Brief [13]) as "an internal state that is expressed by affectively and/or cognitively evaluating an experienced job with some degree of favor or disfavor", and have positioned our survey questions accordingly.

The other human aspect that has been discussed alongside satisfaction is developer motivation. Beecham et al. [7] systematically reviewed literature in software engineering and identified several factors that contribute to software engineers' motivation, as well as external signs of motivation or demotivation. Subsequent work by Sharp et al. [14] reviewed several models found in the literature and ended up proposing a model of motivation in software engineering that includes motivators, outcomes, characteristics, and context. More recently, empirical studies by França and colleagues [15], [16], [17] have identified a variety of factors that affect motivation, such as career progression, or autonomy—at the same time, the authors point out that motivation and job satisfaction are not the same thing [18].

Inspired and informed by the work on developer motivation, we have included relevant factors that may be important to developers in our investigation of job satisfaction and perceived productivity. We have been selective, however, as we agree with work that regards motivation and job satisfaction as related but not identical constructs. To demonstrate, Beecham et al. [7] mention that managers play a role in motivating (or demotivating software engineers), and we included the factor *"manager"* in our survey. In contrast, França et al. [18] mention punctuality as one of the behavioral descriptors of motivation—we have not included a relevant factor as punctuality is a result of motivation rather than a factor that impacts it.

Finally, job satisfaction has also become a subject of inquiry in non-academic developer surveys. The yearly Stack Overflow surveys[3], the International Game Developers Association developer satisfaction surveys[4], and the GitLab annual global developer survey[5] all look at how satisfied developers are with various aspects of their jobs. This signals the importance that industry places on understanding and capturing developers' job satisfaction.

2.2 Developer Productivity

Recently, much attention has been placed on understanding both how software developers work and what makes them productive. Through a systematic literature review, Wagner

---

3. https://insights.stackoverflow.com/survey/2019
4. http://c.ymcdn.com/sites/www.igda.org/resource/resmgr/2017_DSS_/!IGDA_DSS_2017_SummaryReport.pdf
5. https://about.gitlab.com/developer-survey/2018/

and Ruhe [19], [20] identified 51 factors that influence productivity. In addition to the identified technical factors that seem to dominate productivity studies in software engineering, Wagner and Ruhe [19], [20] also distilled a number of soft factors that focus on aspects such as organizational culture and working environment. Wagner and Ruhe's review of the literature included studies that measure perceived productivity as well as use performance measures such as *lines of code* or *function points* as proxies to productivity.

Using a different lens, Meyer et al. [6] looked at how developers perceive and think about their own productivity. Through a survey and subsequent observations and interviews, the study brought to the surface that the developers' sense of how productive they are may be distorted by how many interruptions and context switches they experience. Expanding on their earlier work, Meyer et al. [21] also identified six developer profiles based on the activities that developers feel make them productive—in our study we conceptualize developer profiles based on the time spent on different activities.

Murphy-Hill et al. [22] asked 622 developers in a survey across 3 companies about productivity factors and self-rated productivity. They found that non-technical factors—such as job enthusiasm, peer support for new ideas, and useful feedback about job performance—correlated most strongly with self-rated productivity. In our study, we use similar factors to investigate the relation between perceived productivity and *job satisfaction*. We also studied challenges and their perceived impact on satisfaction. We synthesize the findings into a theory that is then instantiated to different work contexts.

Measuring performance through perceived productivity introduces potential threats to construct validity, but measuring performance of development activity metrics is also potentially problematic. Much attention has been expended on the use of metrics from mined software project data to measure productivity (in terms of velocity and quality). However, such data has been shown to be misleading [23] and may hide activities that may drive down the productivity of an individual developer, but may drive up the productivity or satisfaction of a team overall (e.g., by mentoring newcomers), or may hide activities (e.g., learning a new skill) that influence quality over a longer period of time. In short, the use of a reliable metric of performance in software development remains elusive. These findings highlight that productivity is not only multi-faceted (i.e., various factors influence it) but also highly perceptual—capturing developers' views of their own productivity can be a way to measure performance.

2.3 How Job Satisfaction and Productivity are Related

In organizational psychology, Judge et al. [10] presented a unified theory of the relationship between job satisfaction and performance in 2001, considered as a seminal work in the field. Before this, some researchers assumed no relationship between job satisfaction and performance [24], while others assumed there was a unidirectional relationship between satisfaction and performance [25], [26] and found different variables to account for the relationships discovered [27], [28], [29]. To understand these inconsistent results, Judge et al., conducted a rigorous meta-analysis of over 250 studies and identified 17 unique factors (e.g., autonomy, self-efficacy) that may account for and influence a bi-directional relationship between satisfaction and performance. Of particular interest to our work, Judge et al. found that work complexity impacts the reported correlation between satisfaction and performance, with a higher correlation for jobs of higher complexity over jobs with lower or medium complexity [10]. Job complexity is also explored by Shaw and Gupta [30] in the field of psychology, and is associated with varying levels of job performance depending on how complex participants perceive their work to be. Since software development involves complex work, we may expect that perceived work complexity may impact the relationship between job satisfaction and perceived productivity for developers.

In our research, we aligned with work that views performance and productivity as being related, but considered them beyond the measures of inputs and outputs, and thus focus on capturing perceived productivity. In particular, we were inspired by Judge et al.'s [10] research and findings and built on it as follows:

- we adopted their initial theory that claims a bi-directional *relationship* between job satisfaction and performance, and we applied their theory to investigate job satisfaction and perceived productivity in software engineering;
- we considered candidate factors and relationships they identified as relevant in our own investigation (see the supplemental material [31] for the specific factors we included in our work); and
- we studied a more nuanced view of work complexity in software development as different types of development activity may impact how job satisfaction and perceived productivity are related.

In our study, we also considered challenges software developers experience in their work and how these challenges impact how satisfied developers are with the various factors of their jobs. This is a further extension on the work by Judge et al. [10] as they do not explicitly discuss challenges in their study. We discuss the methodology of our study in more detail next.

3 RESEARCH GOALS AND METHODOLOGY

The main **research goals** driving our study consisted of understanding which social and technical factors and challenges may impact developer job satisfaction, and to develop a theory that captures how these factors influence the complex interplay between job satisfaction and perceived productivity.

This goal grew from a three month site visit by the first author which aimed to understand and align different efforts at a large company to understand and measure developer productivity. During this site visit, the first author interviewed developers and team leads, attended internal meetings concerning productivity, and examined existing survey results. These initial interviews and team meeting observations are outside the scope of this paper, but we mention this to provide motivation for our survey. Furthermore, the internal survey results and meeting notes cannot be shared outside the company. The factors that emerged from this visit and the disagreement between the relationship between job satisfaction and perceived productivity, are also found in other research literature. The research contribution we report in this paper is focused on the survey.
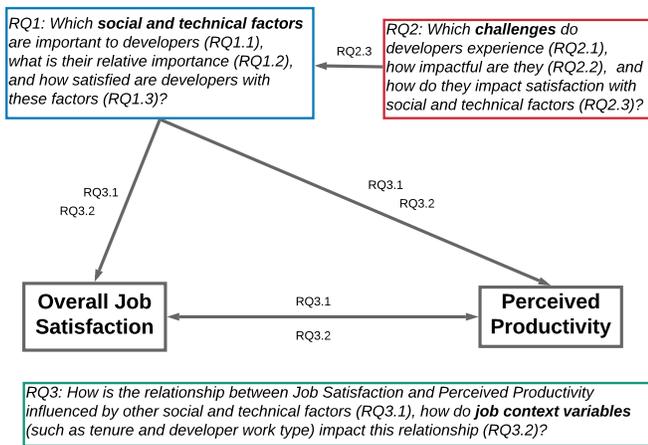
Fig. 1. A summary of the main research questions we explored to inform our theory of developer satisfaction and productivity.

Figure 1 shows the **main research questions** we aimed to answer through our survey and shows how the answers to those questions helped us form an initial theory about developer job satisfaction and perceived productivity (presented in Section 7). Our survey was developed in an iterative manner to derive, confirm, and investigate the social and technical factors and challenges that impact developer job satisfaction and perceived productivity. In the following, we describe characteristics of the case company we studied, and discuss how we designed and refined the survey, and how we analyzed our data.

3.1 The Case Company

Our case company, Microsoft, is a large software company with tens of thousands of developers distributed in offices around the world. The company has significant variety in terms of the products developed, the size and composition of the development teams, as well as the software development tools and processes: some organizations at the case company use traditional waterfall processes, while others use a variety of agile practices.

Microsoft's leadership team cares deeply about developer satisfaction and productivity, and several teams expend considerable effort to understand the factors that may impede satisfaction and productivity within their sub-organizations. During an initial three month research visit at the company, the first author attended meetings and informally interviewed members from four different organizations at the company, to understand how they conceptualized and aimed to measure productivity and developer satisfaction. Internal surveys were also examined, and pointed to various factors that may impact developer satisfaction and productivity at this company. We found that there was already an emphasis on improving engineering tools and processes, but that other factors played a role, such as challenges with technical dependencies and documentation resources. However, it was not clear how much impact engineering tools or these other factors had on developer satisfaction and productivity, motivating our research.

3.2 Developer Satisfaction and Productivity Survey

The main goal of the survey was to determine which social and technical factors are important to developers, how satisfied they are with these factors, and which challenges developers experience. The survey was developed in two main iterations (both iterations were also piloted numerous times). From our literature review (see Section 2), as well as insights from our onsite observations and examination of internal survey verbatim comments, we initially identified 30 factors that may impact satisfaction and/or productivity, as well as 15 challenges.

In our survey, using Likert-type scale questions, we asked developers how important the social and technical factors are to them, how satisfied they are currently with these factors in their current job, which challenges impact their work and how much of an impact they have (from very little to a great extent). For perceived productivity we asked how satisfied developers were with their productivity. We phrased the question in this way to be consistent with the phrasing of other questions that probed on satisfaction of factors. We also used open-ended questions to probe for additional factors and challenges in case any new ones might appear.

Following several small pilot studies, we sent a first version of our survey to 4000 software engineers in March 2017. To incentivize participation, survey respondents could enter a raffle of four $50 Amazon.com gift certificates. No reminder emails were sent. We received 591 responses, a response rate of 14% (comparable to the response rates of many other software engineering surveys [32]). Despite running several pilots, we were surprised to find an additional 10 factors and 9 additional challenges when two of the authors coded the answers to the open-ended questions. This prompted us to update the survey to include these additional factors and challenges, and redeploy it in October 2017.

Our initial survey deployment also revealed that "type of work" was a very important factor in terms of overall developer satisfaction, and in turn their productivity. Thus, in the final version of the survey, we added questions to probe how developers spend their time: we asked them to approximate how many hours they spend writing code, testing, reviewing code, writing documentation, working with requirements, attending meetings, answering emails, learning, doing admin tasks, networking, and helping others.

Our final survey was distributed to another 5000 employees in software engineering, program management, and data science, sampled uniformly at random across all product groups and geographic locations, but not including engineers solicited in the earlier pilot surveys. To incentivize participation, survey respondents could enter a raffle of four $50 Amazon.com gift certificates. No reminder emails were sent. We received 640 responses in total, a response rate of 13% (comparable to the response rates of many other software engineering surveys [32]), of which 465 indicated that they were software developers. In this paper, we consider only responses from the 465 developers that answered our survey. The (sanitized) final survey instrument we used in our study is provided in the supplemental material [31], an extract of the survey with the actual responses is shown in Figure 2. We used the answers to the questions "Overall, how satisfied are you with your current job?" (*Overall Satisfaction*) and "I am satisfied with my productivity at work." (*Perceived*

Overall, how satisfied are you with your current job?
(Very Dissatisfied / Dissatisfied / Neither satisfied nor dissatisfied / Satisfied / Very Satisfied)

I am satisfied with my productivity at work.
(Strongly disagree / Disagree / Neither disagree nor agree / Agree / Strongly agree / Not applicable)

Please rate your agreement with each of the following statements: *(Strongly disagree / Disagree / Neither disagree nor agree / Agree / Strongly agree / Not applicable)*
- I am satisfied with my manager
- I am satisfied with the feedback I receive on my work
- ...

Please let us know how important the following factors are to you. *(Not important / Slightly important / Moderately important / Important / Very important / Not applicable)*
- Having a good manager
- Receiving feedback on my work
- ...

How much do each of the following challenges impact you? *(Not at all / Very little / Somewhat / To a great extent / Not applicable)*
- Poorly defined goals
- Poor team culture
- ...

We want to get a sense of your typical work week. In the table below, please enter roughly how many hours per week you spend on each of the activities.
- Writing code
- Debugging or fixing bugs
- ...

Fig. 2. An extract of the final survey, which asked about job satisfaction, perceived productivity, and importance and satisfaction with a variety of social and technical factors. The survey also asked about challenges and how developers spend their time. The complete survey instrument is provided in the supplemental material [31]. The items in the survey were presented in random order to the survey participants.

Productivity) as the dependent variables for the analysis in this paper.

The 44 factors we included in our survey to investigate job satisfaction and perceived productivity came from a variety of sources. 25 factors (57%) came from reviewed literature. 5 factors (11%) surfaced as relevant during the on-site visit, through discussions with developers and other stakeholders in the case company. 10 factors (23%) came from responses to open-ended questions included in the first main iteration of our survey deployed at the case company in the Spring of 2017. More information on this earlier survey is provided in the description of our methodology. Finally, 4 factors (9%) were added from Shaw and Gupta's paper on work complexity [30], as the earlier survey indicated that the type of work may have an influence on developer satisfaction and productivity. The complete mapping of factors to where they originated (earlier survey, internal discussions, literature) is provided in table form in the supplemental material [31]. In the survey, the items were presented in random order to the survey participants in order to reduce ordering bias.

3.3 Analysis Approach

The data we analyzed in this paper comes exclusively from the responses to the final survey deployed in the Fall of 2017.

We used R to quantitatively analyze the survey data and produce visualizations.

Our initial analyses for both RQ1 and RQ2 examined the distribution of Likert-type scale responses for importance of and satisfaction with each of the social and technical factors, as well as the impact and the frequency of each challenge. Ranking and visualization of these distributions helped identify the factors developers felt were most or least important to them and what challenges were encountered, along with their impact.

One aspect of RQ2 was to explore how the challenges developers experience impact satisfaction with each of the social and technical factors. In this case, we looked for the relationships between 24 challenges and 44 factors. For each possible challenge/factor pair, we performed a correlation analysis and report pairs where there was a statistically significant correlation. Since our analysis was on Likert scores, which may not be normally distributed, we use a Spearman (non-parametric) correlation. Note that since we performed over 1000 correlations, it is likely that spurious statistically significant correlations ($p < 0.05$) may have occurred by chance. We address this by using Bonferroni p-value correction [33] and only report relationships that are statistically significant at the 0.05 level after such correction. All statistically significant correlations had a positive correlation above 0.75, indicating a strong relationship.

For RQ3.1, we aimed to understand the relationship between individual factors and perceived productivity and satisfaction. To accomplish this, we used statistical analysis to model these relationships, using the factors as independent variables and perceived productivity and overall satisfaction as dependent variables. That is, because each respondent indicated their satisfaction level with each factor and also indicated their overall satisfaction, we were able to model the relationship of each factor with overall satisfaction (and similarly, with perceived productivity).

A common approach for measuring the impact of a number of factors on satisfaction or productivity is to use linear regression [34], [22]. For the analysis of RQ3, we used stepwise linear regression with *Overall Satisfaction* and *Perceived Productivity* as the dependent variables. Regression relies on an independence assumption (that the impact of one factor is not affected by the impact of another) and that the impact of a factor is global and consistent across the range of outcome values (if factor X increases satisfaction by 0.5, then it will increase a base satisfaction of 1.0 to 1.5 *and* from 4.0 to 4.5). To check for the independence of factors we computed correlations between the satisfaction scores for each pair of factors. We then applied hierarchical clustering on the resulting correlation matrix. As distance function between two factors X_a and X_b we used $1 - abs(cor(X_a, X_b))$. This distance function puts highly *cor*related factors into the same clusters. Since correlations can be negative (no factors were), we use the *abs*olute value of the correlation. The result of hierarchical clustering is a *dendrogram*, i.e., a tree representation that illustrates how the clusters are merged for different distance thresholds. For the analysis of RQ3, we combined factors that were in "highly correlated" clusters, which we defined as a correlation of 0.5 or more (and corresponds to a distance 0.5 or less). As an example, the satisfaction for the factors *Engineering processes*, *Engineering*

tools, and *Collaboration tools* was highly correlated and we combined the factors into the composite factor *Engineering system*. When combining the factors, we averaged the scores. For example, if the satisfaction with the three base factors was 4, 5, and 4, the score for the composite factor *Engineering system* was $(4 + 5 + 4)/3 = 4.33$.

To further check for collinearity among the explanatory variables in the regression models, we checked for Variable Inflation Factors (VIF). A common practice is to remove any variables in the final model that have a VIF score higher than 5 as suggested by Fox [35]. None of the factors in our models had a VIF score higher than 5, most scores were lower than 2.5.

To facilitate ranking and comparison of coefficients for the different models [36], we centered and scaled the input data for the regression models. Specifically, we transformed each factor X with $(X - mean(X))/\text{sd}(X)$. Subtracting the *mean* centered all factors around 0. Dividing by the standard deviation (*sd*) scaled all factors to the same unit.

In this paper, if not stated otherwise, the findings from the regression analysis are statistically significant with a p-value of 0.05 or less. We completed all analysis in the R statistical software and in PowerBI.

In RQ3.2, part of our exploration of work context involved dividing developers into groups based on their "type of work". To obtain these groupings, we used the *Partitioning Around Medoids* (PAM) clustering algorithm [37] on the self-reported time spent on different activities from the survey. Most clustering algorithms require the user to specify the number of clusters. One of the benefits of the PAM algorithm is that for each possible k (number of clusters), the algorithm computes a clustering and then returns the clustering with the optimum average silhouette width [38]. We then used the same regression analysis as above on these clusters of developers.

3.4 Survey Demographics

As mentioned earlier, we received 640 responses to our survey, of which 465 indicated they were developers and we focus solely on these responses for the remainder of this paper. Here we report a summary of general demographics of the responses. Not all of the survey questions were required, so not all of the categories we report sum to 465. In terms of gender, 381 (86%) identified as male and 61 (14%) as female. 97 (21%) respondents indicated they are in a management position, while 363 (79%) have no people reporting to them. The experience of the respondents (measured via multiple choice questions) ranged from one year (101 respondents, 22%) to more than 20 years (11, 2%) with a median of between two and five years (107, 23%). Responses came from North America (322, 71%), Europe (52, 11%), and Asia (82, 18%), with North America representing the majority. In terms of working environment, 161 (35%) work in their own office, while 303 (65%) share their office or open workspace with others (247 of those share their workspace with at least five other people, 53%). From a collaborative perspective, 413 developers (89%) responded that they coordinate their work with at least three people, and 184 (40%) indicated they coordinate with six or more people, evidence that development at the company is a highly collaborative activity.

4 WHICH SOCIAL AND TECHNICAL FACTORS MATTER TO DEVELOPERS? (RQ1)

RQ1.1: Which social and technical factors impact job satisfaction and perceived productivity?
From the three month site visit (which consisted of informal interviews with internal organizations and analysis of previous organization specific internal surveys), literature review, and analysis of the open-ended questions from the first deployment of our survey in March 2017, we arrived at the 44 factors shown in the left-hand column of Table 1.[6]

The *work complexity* factors (shown near the bottom of the table) deserve special mention. These were added to our final survey when the *type of work* factor emerged as an important factor in the Spring 2017 survey. Shaw et al. propose [30] that certain job complexity factors impact job performance and well-being. In our survey, we did not ask about importance nor satisfaction for these factors, but rather we asked the respondents for their agreement (along a 5-point Likert-type scale) with statements for these factors (e.g., "I can complete my tasks" and "My job takes a long time to learn").

We reviewed the open-ended questions for potential additional factors. No additional factors emerged from our final survey deployment with the exception of privacy and contributing to social good; we have not included these in our analysis as they were mentioned only one time each.

RQ1.2: What is the perceived importance of these factors?

For each factor, respondents were asked to rate its importance using Likert-type choices of "not important", "slightly important", "moderately important", "important", and "very important".

The 2nd column of Table 1 shows the relative perceived **importance** of the factors as rated by the developer survey respondents. Close to half of the respondents felt the factors were all somewhat important (as we included factors that showed importance in the literature or previous surveys). The percentage next to the distribution indicates the percentage of responses that rated the factor as being "important" (4) or "very important" (5). The rank represents the order of factors by percentage. Having a good *manager* (rank #1), feeling *productive* (rank #2), being *fairly rewarded* and having a positive *team culture* are ranked as important or very important by over 94% of the respondents. While *using one's skills effectively* and having *impact at work* are ranked as moderately important or important by over 93%.

Training (for tools, soft skills, and technologies) have lower perceived importance than other factors (but still important to over 43% of the respondents). A surprising result was that having *a private office* is important or very important to only 44% of the respondents. This factor featured prominently in many of the internal surveys we reviewed during the site visit and we expected to see this emerge as important.

RQ1.3: How satisfied are developers with these factors?
We asked developers how **satisfied** they are with each of the factors we identified. The third column of Table 1 shows the levels of developer satisfaction with the different factors,

---

6. The origin of the factors (in terms of literature, on-site visit, or coded from the first survey) is shown in the supplemental material [31].

TABLE 1
Social and technical factors that matter to developers(RQ1.1). Their importance (RQ 1.2) was rated on a scale from 1 (not important) to 5 (very important). The factors are sorted by their relative importance. For job satisfaction, developers were asked to indicate agreement that they were satisfied with each factor on a scale from 1 (strongly disagree) to 5 (strongly agree). The percentages are the respondents that answered 4 or 5 for Importance (*Impt*) and Satisfaction (*Sat*). *Rank* is based on ordering of these percentages.

| Factor (RQ1.1) | Importance (RQ1.2) | | | Satisfaction (RQ1.3) | | |
	Distribution 1 2 3 4 5	Impt.	Rank	Distribution 1 2 3 4 5	Sat.	Rank
Manager		97.8%	#1		80.9%	#3
Perceived productivity		97.0%	#2		61.1%	#29
Rewards		96.3%	#3		63.2%	#26
Team culture		94.8%	#4		73.1%	#10
Skills are well used		93.5%	#5		63.8%	#24
Impactful work		93.3%	#6		71.4%	#16
Skilled co-workers		92.4%	#7		85.1%	#1
Technical competency		92.0%	#8		72.5%	#12
Ability to achieve goals		91.7%	#9		74.0%	#8
Work life balance		91.3%	#10		71.9%	#13
Organization culture		90.7%	#11		68.3%	#18
Salary		90.5%	#12		57.1%	#32
Time to complete tasks		90.1%	#13		73.8%	#9
Benefits		89.2%	#14		77.7%	#6
Type of work		88.6%	#15		75.0%	#7
Work feedback		87.9%	#16		68.2%	#19
Engineering processes		87.8%	#17		57.5%	#33
Important member on one's team		87.6%	#18		71.8%	#14
Hardware		87.4%	#19		71.5%	#15
Appreciation shown for work		86.8%	#20		66.2%	#20
Engineering tools		85.7%	#21		59.7%	#30
Collaborative team		84.8%	#22		78.0%	#5
Promotions		83.8%	#23		47.5%	#38
Vision		82.0%	#24		63.2%	#25
Well defined goals		81.3%	#25		61.9%	#28
Clear priorities		81.3%	#26		65.0%	#22
Job security		79.8%	#27		79.3%	#4
Stress		78.8%	#28		56.7%	#33
Learning skills useful for future		78.3%	#29		63.1%	#27
Physical working environment		77.5%	#30		72.9%	#11
Autonomy		72.1%	#31		64.7%	#23
Collaboration tools		71.3%	#32		66.2%	#21
Working privately (interruption free)		70.3%	#33		56.4%	#34
Availability of documentation		67.0%	#34		46.4%	#39
Proximity to team		62.1%	#35		83.5%	#2
Lateral move opportunities		56.1%	#36		50.7%	#36
Training for engineering technologies		55.1%	#37		41.8%	#40
Training for soft skills		54.7%	#38		53.4%	#35
Private office / Nr. people in workspace		44.3%	#39		68.4%	#17
Training for engineering tools		43.5%	#40		48.1%	#37

WORK COMPLEXITY
We did not ask about importance or satisfaction of factors related to work complexities.
Instead we asked about agreement *with statements related to work complexity.*

I am able to complete tasks					85.6%	
Job requires a lot of skill					79.6%	
Takes long time to learn job					62.8%	
Job is complex					58.8%	

OVERALL SATISFACTION
Overall satisfaction					71.2%	

including agreement with statements concerning the complexity of their work. The percentage next to the distribution corresponds to respondents that agree or strongly agree they are satisfied with the particular factor. The rank is how the factor compares to the other factors when ordered by percentage of agreement. For most of the factors, over 50% of the respondents agree or strongly agree that they are satisfied with these factors. 85% agree they can complete their tasks at work, but only 61% said they are satisfied with their productivity, while 97% agree productivity is an important factor to them. There is high satisfaction with team factors (*team culture*, *skilled co-workers*, *collaborative team*), while the factors about *training for engineering technology* and *tools*, *availability of documentation*, and opportunity for *promotions* have less than 50% agreement of satisfaction.

> **Takeaways from RQ1:** Having a good manager, being productive, fair rewards, collaborative team culture, using one's skills effectively and having impact at work were perceived as most important at our case company. Developer satisfaction with their manager, team culture and skills was high, but satisfaction with personal productivity, engineering tools and processes, training and documentation was relatively low.

5 WHICH CHALLENGES DO DEVELOPERS EXPERIENCE? (RQ2)

Related literature discusses not just enablers of developer satisfaction and productivity, but also the challenges or barriers that impede satisfaction and productivity (see Section 2). Although we expected that poor satisfaction with social factors and technical factors can be considered as challenges, we also anticipated additional challenges may emerge.

RQ2.1: Which challenges impact developers?
In the final survey, we queried about the impact of a list of 24 challenges[7] and again invited the respondents to tell us of any other challenges they experience. This time only one new challenge emerged, which was "poor language skills" (mentioned only once). Table 2 shows the full list of challenges. Not all challenges are the dual of factors, notably challenges such as *too many communication channels*, *too many meetings* and *too many emails*, and challenges about software quality such as *poor architecture*, *legacy code*, and *too many external dependencies* emerged.

RQ2.2: What impact do these challenges have?
In Table 2, we show the relative distribution of those that responded that a given challenge is either a slight challenge (1), a moderate challenge (2), or a big challenge (3). The table is ranked by the average impact of a challenge. Challenges with *software architecture*, *legacy code*, and *documentation* were perceived as having the most negative impact. Note that the impact of a challenge, does not correspond to frequency of a challenge, e.g., a challenge with high impact can be infrequent. The most frequently reported challenges were related to *finding relevant information*, *legacy code*, *software architecture*. The least frequently reported challenges were related to *team culture*, *pooly qualified co-workers*, and *managers*.

7. The origin of the challenges, in terms of the site visit, literature, or coded from the first survey, is shown in the supplemental material [31].

TABLE 2
Challenges reported by developers (RQ2.1) and how impactful they reported these challenges to be (RQ2.2). The rank is based on the average of the impact of the challenge. Impact levels are small (1), moderate (2), and large (3).

Challenge(s)	Distribution 1 2 3	Impact	Rank
Poor software architecture		2.12	#1
Legacy code		2.09	#2
Finding relevant information		2.07	#3
Too many external dependencies		2.05	#4
Poor engineering tools		2.02	#5
Too many interruptions		2.01	#6
Manager		1.98	#7
Team culture		1.97	#8
Poor software development processes		1.96	#9
Time to do required work		1.95	#10
Lack of quiet space for work		1.94	#11
Unclear requirements		1.93	#12
Poorly defined goals		1.92	#13
Poor hardware resources		1.92	#14
Lack of vision		1.90	#15
Too many communication channels		1.89	#16
Too many meetings		1.82	#17
Too many emails		1.79	#18
Development stack/architecture		1.75	#19
Poorly qualified co-workers		1.72	#20
Insufficient training		1.72	#21
Technical competency		1.71	#22
Poor collaboration tools		1.69	#23
Interacting with people		1.68	#24

RQ2.3: How do challenges impact satisfaction of factors?

As many of the challenges were the dual of the social and technical factors we identified, we suspected that a challenge (if reported by many) would show a correlation with the satisfaction scores of its corresponding factor. In Figure 3, we show how some challenges have relationships not just with their corresponding factor (e.g., one would expect that if *My manager* is a challenge, it would lead to lower satisfaction with the *My manager* factor), but also have a statistically significant relationship with many other social and technical factors. In all cases, the relationships showed a negative correlation; those who indicated they faced the challenge also indicated a lower level of satisfaction for the factor.

From Fig. 3, we see that when developers report *manager* as a challenge, they not only report lower satisfaction with their *manager*, but they also report lower satisfaction with 15 other factors. Some of these relationships are not surprising, as managers impact whether developers may be shown *appreciation* for their work, how much *autonomy* they feel, and may influence the *team culture*. But some correlated factors such as *type of work* were more surprising to us. Also the correlation of *too many external dependencies* challenges with the *work-life balance* factor was not something we expected to find. Conversely, we see that some factors show a relationship with more than one challenge. For example, the factor *collaboration tools* correlates not just with the challenge of *poor collaboration tools*, but also with the challenges of *poor engineering tools*, *poorly defined goals*, and *poor software architecture*.
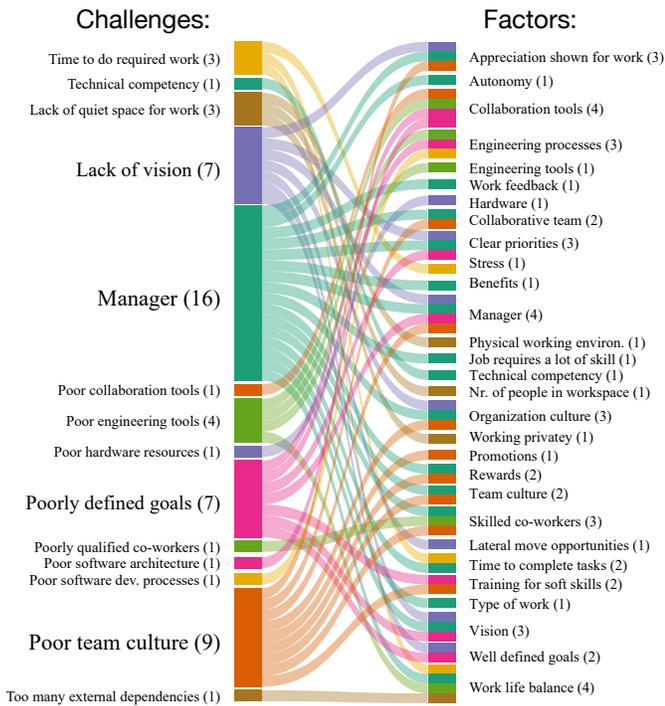
Fig. 3. What challenges correlate with satisfaction scores of social and technical factors. Numbers in ()s indicate how many relations affect the challenge or factor. (RQ2.3)

Fig. 4. The result of clustering technical/social factors into groups based on their correlations with each other. The correlation cut-off of 0.50 (vertical dashed red line) is used to identify 20 groups CF1..CF20, which are then refined into composite factors by averaging their scores.

Takeaways from RQ2: At the case company, poor *software architecture*, *legacy code*, poor *documentation*, poor *engineering tools*, and too many *work interruptions* are **challenges** that developers feel have a big impact. The challenge of a poor *manager* is correlated with lower satisfaction across 16 different social and technical factors.

---

6 HOW IS OVERALL JOB SATISFACTION AND PERCEIVED PRODUCTIVITY RELATED? (RQ3)

We consider this relationship for all developers and how other factors may mediate this relationship (RQ3.1). We then consider how various work context variables (specifically tenure and time coding) impact these relationships (RQ3.2).

For this analysis, we grouped the 44 individual factors into 20 groups as follows. We first identified groups of factors with highly correlated satisfaction scores; each group corresponds to a **composite factor**, for which we compute a new satisfaction score by averaging the scores of the constituent factors. To identify highly correlated groups, we applied hierarchical clustering to the correlation matrix. Figure 4 shows the dendrogram with the results. The tree structure to the right of the list of factors indicates the order in which factors are merged into groups. For example, *type of work* is first merged with *impact of work*, followed by *vision for my work* and *achieve the goals*. To identify groups of highly correlated factors, we used a cut-off of 0.5 (indicated by the vertical dashed red line). In the example, we combined the four factors in group CF1 in Fig. 4 into the composite factor *impactful work*.

We made adjustments to the result of the hierarchical clustering as follows:

- We split the cluster CF4 into three composite factors (1) *appreciation and rewards*, (2) *autonomy*, and (3) *work culture* because they capture different aspects of software developers' work.
- We combined the clusters CF7 (*training for technologies; training for engineering tools; documentation*) and CF8 (*training for soft skills*) into the composite factor *training and documentation* because the two clusters were conceptually related.
- We combined the clusters CF19 (*my job requires a lot of skill; my job is very complex*) and CF20 (*long time to learn job*) into the composite factor *job characteristics* because the two clusters were conceptually related (as discussed by Shaw and Gupta [30]).

The mapping of the resulting composite factors to constituent factors is shown in the first and second column of Table 6.

RQ3.1: How do social and technical factors impact relationships between job satisfaction and perceived productivity?

To answer this research question, we computed linear regression models for *overall job satisfaction* and *perceived productivity* to model how other factors explain the variation in the responses to these two questions.

We see from Table 3 that the composite factors *appreciation and rewards* (0.117), *impactful work* (0.248), *important contributor* (0.163), *work culture* (0.198), *work life balance* (0.119) and *perceived productivity* (0.097) explain the **job satisfaction** of developers. The highest coefficient is for the factor *impactful work* (0.248): thus an increase of one standard deviation in this factor leads to an increase of 0.248 standard deviations of overall job satisfaction.

TABLE 3
The linear regression model with standardized coefficients showing how different factors influence overall **job satisfaction** for developers. The level of statistical significance is indicated with asterisks: (*) for $p < .05$, (**) for $p < .01$, and (***) for $p < .001$. Adjusted R-squared: 0.581. (RQ3.1)

Variable	Coeff.	
Appreciation and rewards	0.117	*
Impactful work	0.248	***
Important contributor	0.163	***
Work culture	0.198	***
Work life balance	0.119	**
Perceived Productivity	0.097	*

TABLE 4
The linear regression model with standardized coefficients showing how different factors influence **perceived productivity** for developers. The level of statistical significance is indicated with asterisks: (*) for $p < .05$, (**) for $p < .01$, and (***) for $p < .001$. Adjusted R-squared: 0.5585. (RQ3.1)

Variable	Coeff.	
Autonomy	0.201	***
Can complete tasks	0.195	***
Can switch teams	−0.082	*
Compensation	−0.114	**
Engineering system	0.250	***
Important contributor	0.109	*
Impactful work	0.208	**
Job characteristics	−0.103	**
Personal technical skills	0.089	*
Work environment	0.078	*
Job Satisfaction	0.122	*

TABLE 5
Three developer clusters based on time spent on work activities, showing % of total time and number of hours spent per week on each activity. (RQ3.1)

Activity	Cluster C1 (size 169)		Cluster C2 (size 191)		Cluster C3 (size 48)	
Coding, debugging, bug fixing	60.4%	28.0	36.3%	18.9	9.3%	4.9
Working on specs/requirements	2.1%	1.1	4.2%	2.4	4.4%	2.6
Running tests on code	8.5%	4.0	8.5%	4.7	7.3%	2.5
Reviewing code	5.3%	2.5	8.7%	4.4	5.8%	2.9
Maintaining relationships	1.4%	0.7	2.5%	1.4	3.3%	2.0
Meetings	8.4%	3.8	12.2%	6.3	27.8%	12.9
Continuous learning	2.3%	1.1	4.1%	2.3	4.4%	2.3
Helping others, mentoring	3.2%	1.6	7.4%	3.9	11.5%	5.0
Documentation	1.3%	0.7	2.7%	1.6	2.7%	1.4
Email	6.0%	2.7	10.7%	5.7	19.2%	10.4
Administrative tasks	1.1%	0.5	2.7%	1.5	4.3%	2.3
Overall satisfaction	3.8		3.7		3.6	
Productivity	3.7		3.4		3.4	
Satisfaction with work type	4.0		3.9		3.8	

In Table 4, we see that many composite factors explain the variation with **perceived productivity**. The factors with the highest coefficients are *engineering system* (0.250), *impactful work* (0.208), *autonomy* (0.201), and *can complete tasks* (0.195). Some factors have negative coefficients: *can switch teams* (−0.082), *compensation* (−0.114), and *job characteristics* (−0.103). This means that an increase in these factors is linked with a decrease in how productive developers feel. For example, a developer who is more satisfied with their compensation may work on more challenging tasks that make them feel less productive. Other factors that explain perceived productivity with positive coefficients are *important contributor* (0.109), *personal technical skills* (0.089), *work environment* (0.078), and *job satisfaction* (0.122).

RQ3.2: How does work context impact the relationship between job satisfaction and perceived productivity?

For this research question, we investigated if and how work context variables may change which factors influence developers' job satisfaction and perceived productivity. Since *type of work* emerged as a key factor during our analysis, we consider time spent on different developer activities (see Section 3.2), as well as seniority of developers in terms of *technical experience*. We note that this is a departure from previous work where work context has included variables such as managers and pay [39]. Such variables are still included in our list of factors impacting satisfaction and productivity (gathered from literature as described in Section 2). We focused on tenure and type of work as work context since these factors emerged as relevant in our analysis

and the case company. For example, through discussions with stakeholders with our case company, we learned that the amount of technical experience was a factor that may influence both work and productivity satisfaction. Challenges reported by respondents to the Stack Overflow developer survey also differed by level of experience.

For **"developer experience"**, we split developers into two groups: junior developers (Jr) that have less than or equal to 5 years of experience (144 in this group), and senior developers (Sr) that have more than 5 years of experience (313 in this group). We use a threshold of 5 years, because in previous internal studies at our case company, we found a marked difference in the responses of those who have been at the company less than and more than 5 years. We computed stepwise linear regressions for these two groups (for satisfaction and perceived productivity) and found that different factors emerge in the models as statistically significant. The regression models are summarized in the RQ3.2 columns of Table 6.

For junior developers, the only statistically significant variable in the job satisfaction regression model is *perceived productivity*. This is not to suggest that other factors do not impact their overall satisfaction, but rather for these junior developers, productivity explains most of the variation in their **job satisfaction** responses. Meanwhile, for senior developers, *impactful work*, *work culture*, and *work life balance*, *important contributor*, and *appreciation and rewards* explain most of the variation in their **job satisfaction** scores. Similarly, there are different regression models for perceived productivity. *Impactful work*, *autonomy*, *can complete tasks*, *job security*, and *job satisfaction* explain the variation in the junior developers' **perceived productivity**, while for senior developers, the variables *compensation*, *work environment*, *can complete tasks*, *engineering system*, *can switch teams*, *autonomy*, and *impactful work* explains the variation in their **perceived productivity**.

For the **"type of work"** demographic, defining different groups was more involved. We used a clustering algorithm to characterize developers according to how they perceive they spend their time on various activities (see Table 5). We found three main clusters. Cluster 1 (C1) includes 169 developers who spend most of their time coding/debugging and testing code, but relatively little time on emails, meetings,

and helping others. Cluster 2 (C2) includes 191 developers who spend about half their time writing/debugging/testing code and more time than Cluster 1 developers on code review, meetings, emails, and helping others. Cluster 3 (C3) includes 48 developers who spend only about 5 hours per week writing code, but much more time in meetings (about 13 hours) and more time on email (about 10 hours).

For C1 and C2 developers, we computed regression models to identify important variables. (C3 contains too few developers to compute a regression model.) For C1 developers, those that spend most of their time coding, *job characteristics*, *work culture*, and how their *appreciation and rewards* are the important variables in the **job satisfaction** model, while for C2 developers, the variables *work culture*, *important contributor*, *impactful work*, and *perceived productivity* explain the variation in their **job satisfaction** scores. In terms of explaining variation in their **perceived productivity**, for C1 (heavy coders), the variables *job characteristics*, *can complete tasks*, *engineering system*, *clear work goals*, *autonomy*, *appreciation and rewards* and *impactful work* are important. For C2 developers (those that code less than C1), the variables *compensation*, *personal technical skills*, *can complete tasks*, *engineering system*, *autonomy*, *important contributor*, and *job satsfaction* explain the variation in their **perceived productivity**.

We draw several observations from Table 6

- For **job satisfaction**, the factors *work culture* (4 out of 5 models), *appreciation and reward* (3 out of 5), *impactful work* (3 out of 5), and *important contributor* (3 out of 5) are statistically significant in most of the models for different work contexts. The factors *work culture* and *impactful work* have the highest coefficients among those factors.
- For **perceived productivity**, the factors *autonomy* and *can complete tasks* are statistically significant in all models (5 out of 5). The factors *engineering systems* (4 out of 5), *impactful work* (4 out of 5), and *compensation* (3 out of 5) are also statistically significant in many models for different work contexts.
- *Job satisfaction* is statistically significant in 3 out of 5 models for **perceived productivity** for different work contexts. Similarly, *perceived productivity* is statistically significant in 3 of the 5 models for **job satisfaction** for different work contexts. The bi-directional relationship between these two constructs is expected based on the work by Judge *et al.* [10], where more productive developers may be more satisfied, and more satisfied developers may be more productive. The factor *impactful work* was important for both job satisfaction and perceived productivity.
- Murphy-Hill et al. [22] found that the three statements that correlated the most with self-rated productivity at Google were "I use the best tools and practices to develop my software", "I am enthusiastic about my job", and "My job allows me to use my personal judgment in carrying out my work". Similarly, in our analysis *autonomy* and *engineering system* have a strong influence. While we did not ask about the job enthusiasm, we also find that non-technical factors such as *impactful work* have a strong influence. (Murphy-Hill et al. did not relate their factors to job satisfaction.)
- Fishbein and Ajzen [40] showed that when people

are asked to rate products, the variables that best predict preferences in a regression model do not always match the same participants' ratings of the importance of factors. This could be interpreted as *"people don't know what's important to them"*. In our analysis, most of the composite factors with high perceived importance ($> 90\%$) are relevant in the regression models: *perceived productivity*, *work culture*, *personal technical skills*, and *important contributor*. However there is also the factor *team skills* with high perceived importance that is not present in any of the regression models. The factor *work environment* was only considered as important by 64.0% but is present in the regression models. We'd like to point out that the absence of a statistically significant correlation does not mean that a factor is unimportant.

We next discuss the implications of these findings.

Takeaways from RQ3: Developers who report they do *impactful work* and are an *important contrbutor* within a positive *work culture*, feel the **most satisfied** with their work. Developers who report they have *autonomy*, do *impactful work*, *can complete tasks*, and are satisfied with the *engineering system*, feel the **most satisfied with their productivity**. At our case company, *work context variables* (notably Work Type and Experience) impact which factors explain **overall job satisfaction** and **perceived productivity**, and in turn can influence the relationships between them.

---

7 TOWARDS A THEORY OF DEVELOPER SATISFACTION AND PERCEIVED PRODUCTIVITY

There is a general lack of appreciation and understanding of the impact of theories in software engineering, with many papers focusing on describing what is observed without any interpretation or conceptualization of the scientific contributions behind the findings in a way that helps explain them or helps in making predictions [41]. One way to add value to the findings from a study is to develop or build on an existing theory so that any findings can be used to communicate and further develop ideas with others in the research community. Likewise in industry, a theory may help with decision making in terms of changes to technology or the processes used [42]. There are different types of theory [42]: some theories describe or conceptualize "what is", while other theories help to explain and/or predict what may happen if changes are made. Theories may also be used to guide design and action.

Several researchers agree that the main elements of a theory [42], [41] are its **constructs**—the entities that the theory strives to describe (which in turn may be **operationalized** using particular measures or metrics), **relationships** between those constructs (that is how constructs are related and why they are related, e.g., through causality or correlations), **boundaries** to scope the applicability of the theory, **propositions**, that in the case of predictive theories, represent predictions based on the theory's constructs and relationships, and finally, **hypotheses** which may instantiate propositions by replacing constructs with appropriate measures.

Figure 1 shows the main research questions we posed in our study. This figure also suggests an initial theory for developer satisfaction and perceived productivity. The two

TABLE 6
The impact of **work context** on job satisfaction and perceived productivity. The first and second column show the mapping from composite factor to the individual technical/social factors. For each composite factor, the first column shows the average perceived importance (*Impt.*) of its constituent factors. In the second column, the table shows the rank of the perceived importance of the constituent factors in parenthesis (#). The remaining columns show standardized coefficients of linear regression models for different work contexts: the entire population of developers (All), junior developers (Jr), senior developers (Sr), developers who spent most of their time coding (C1), and developers who spend less time coding and more time on code review, meetings, emails, and helping others (C2). The third developer cluster (C3) contained too few developers to compute a regression model and hence is not included in the table. Blank cells indicate that a factor was removed during the stepwise regression or not statistically significant in the final model. All coefficients are statistically significant at .05 or lower.

		Satisfaction (RQ3.2)					Productivity (RQ3.2)				
Adjusted R^2:		0.58	0.55	0.59	0.71	0.57	0.56	0.69	0.52	0.52	0.60
Composite Factor	**Constituent Factor(s)**	All	Jr	Sr	C1	C2	All	Jr	Sr	C1	C2
Appreciation and rewards (Importance 88.7%)	Rewards for my work (#3); Promotion opportunities (#23); Feedback on my work (#16); Appreciation shown for my work (#20)	0.12		0.14	0.28					-0.26	
Autonomy (Importance 76.7%)	How my work priorities are defined (#26); Autonomy (#31)						0.20	0.20	0.18	0.25	0.15
Can complete tasks	Can complete my tasks						0.20	0.21	0.20	0.20	0.24
Can switch teams (Importance 56.1%)	That I can switch teams (#36)						-0.08		-0.12		
Clear work goals (Impt. 81.3%)	Work goals are well defined (#25)									-0.17	
Compensation (Impt. 89.8%)	My salary (#12); My benefits (#14)						-0.11		-0.16		-0.17
Engineering system (Impt. 81.6%)	Engineering tools (#21); Engineering processes (#17); Collaboration tools (#32)						0.25		0.30	0.26	0.31
Hardware (Impt. 87.4%)	Hardware (#19)										
Impactful work (Impt. 86.8%)	Type of work (#15); Impact of work (#6); Vision for my work (#24); Can achieve the goals (#9); Learning skills useful in future (#29)	0.25		0.25		0.31	0.21	0.36	0.18	0.28	
Important contributor (Impt. 90.6%)	My skills are well used (#5); Important member on my team (#18)	0.16		0.16		0.18	0.11				0.15
Job characteristics	My job requires a lot of skill; My job is very complex; Long time to learn job					0.16	-0.10			-0.24	
Job security (Impt. 79.8%)	Job security (#27)							-0.15			
Personal technical skills (Impt. 92.0%)	My technical skills (#8)						0.09				0.11
Proximity to team (Impt. 62.1%)	Proximity to my team (#35)										
Team skills (Impt. 92.4%)	Team skills and abilities (#7)										
Training and documentation (Impt. 55.1%)	Training for technologies (#37); Training for engineering tools (#40); Documentation (#34); Training for soft skills (#38)										
Work culture (Impt. 92.0%)	Team Culture (#4); Organization Culture (#11); How collaborative my team is (#22); My manager (#1)	0.20		0.22	0.27	0.20					
Work environment (Impt. 64.0%)	Number of people in my workspace (#39); My physical work environment (#30); Private work with few interruptions (#33)						0.08		0.10		
Work life balance (Impt. 86.8%)	Work life balance (#10); Time to complete my work (#13); I feel not overly stressed at work (#28)	0.12		0.15							
Perceived Productivity (Impt. 97.0%)	Perceived Productivity (#2)	0.10	0.22			0.25					
Overall Job Satisfaction	Overall Job Satisfaction						0.12	0.17			0.23

main **constructs** in our theory are developer satisfaction and perceived productivity. We anticipated bi-directional **relationship** between these two constructs (building on the work by Judge *et al.*), where more productive developers may be more satisfied, and more satisfied developers may be more productive. Table 6 shows that this bi-directional relationship does exist (although in some cases it is indirect and mediated by other social and technical factors), moreover, work context variables also play a role.

In Fig. 5 we present a generic theory that captures how constructs—social and technical factors (and indirectly challenges)—relate to the job satisfaction and perceived productivity constructs. Which factors and how they impact these relationships may vary according to specific contextual work factors (in our case, we found variations in these factors for type of work and work experience). However, we emphasize that the factors, challenges, and work context variables may be quite different for other companies or development contexts.

We used the meta-study by Judge et al. [10] as a summary of factors that earlier work found contributing to job satisfaction, and built on them in our investigation and theory building. Through our case study and survey, we instantiated the theory for the company we studied, and as such, it is **bounded** to this context. The developer satisfaction and productivity constructs are **operationalized** by reported satisfaction and perceived productivity, respectively.

Developing, refining, and deploying the survey at our case company either confirmed or revealed 44 social and technical factors (in answer to RQ1.1) and 24 challenges
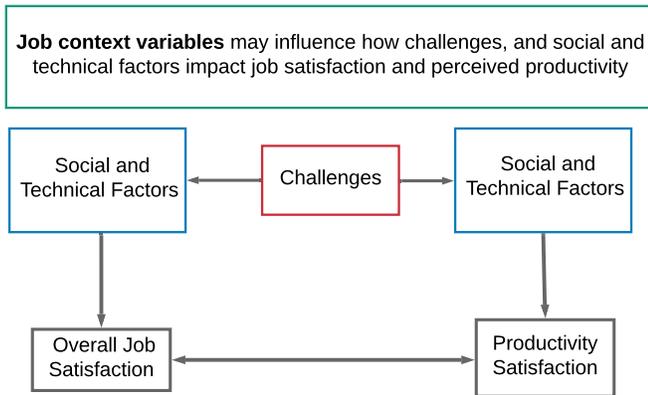
Fig. 5. Towards a generic theory of developer job satisfaction and perceived productivity. Table 1 lists social and technical factors we found in our study, and Table 2 for the list of challenges. The context variables that emerged from our study are tenure and type of work but other variables such as gender or culture may play a role in this or other cases.

Fig. 6. Instantiating the theory of developer satisfaction and perceived productivity for our case company (for all developers that answered our survey). We find their job satisfaction and perceived productivity are related and also find which other factors (and indirectly the main challenges) are related to job satisfaction and perceived productivity.
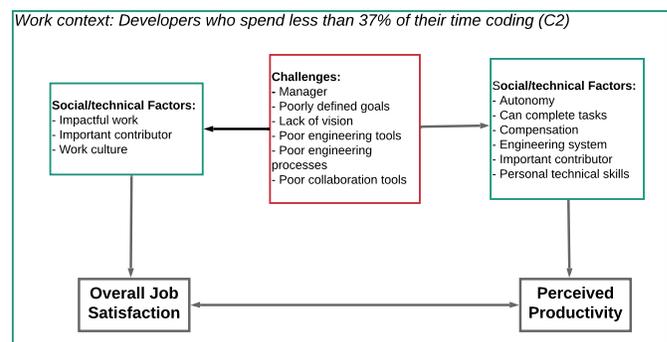

Fig. 7. Instantiating and scoping the theory to developers that spend less than 37% their time coding (C2). We find their job satisfaction and perceived productivity are related, and which factors (and indirectly which challenges) are related to job satisfaction and perceived productivity.

that impact these factors (in answer to RQ2). To answer RQ3, we identified 20 composite factors that are social and technical factors but represent groups of highly correlated factors. These composite factors were used as the variables in our regression models. The models showed which of these composite factors help explain the variation in overall satisfaction and perceived productivity at our case company.

Figure 6 shows the *instantiation* of the generic theory for our case company and includes the specific composite factors that appeared in our models, as well as the challenges that we showed (as part of RQ2) that impact these particular factors. A further instantiation and scoping of this theory to developers who spend less than 37% of their time coding/testing/debugging (cluster C2) is shown in Figure 7. In this case, the figure shows which composite social/technical factors and indirectly which challenges relate to the main constructs of job satisfaction and perceived productivity specifically for the developers in the C2 cluster.

We discuss how the factors in our theory and our theory instantiations link to earlier work, as well as the implications from these findings in the next section. Note that our case company as well as different work context variables set a boundary (i.e., scope) for the theory instantiations we present. Consideration of other work context variables (see Table 6) are indicative of other instantiations of the generic theory.

Our instantiations of the theory at our case company now give us a basis to form **propositions** that could lead to actionable insights for our case company and may suggest actions that could be applied in similar development contexts.

Examples of propositions are:

- A manager that defines clear work priorities and establishes a good work culture should improve developer satisfaction and lead to higher perceived productivity.
- Showing developers how their work has impact and how they are an important contributor may make them feel more satisfied and productive.
- More effective engineering tools and processes should help support higher perceived productivity and job satisfaction, however, other non-technical factors need to be considered.

These propositions may be framed as **hypotheses** to be tested empirically. Testing such hypotheses would require identifying suitable measures for each of the constructs in the propositions (e.g., more effective engineering tools may be operationalized by asking developers how satisfied they are with their engineering tools). The implications of the propositions and theory for practitioners and researchers are discussed further in the next section of this paper.

8 IMPLICATIONS FOR PRACTICE AND RESEARCH

Many organizations—our case company included—strive to improve engineering tools and processes with the aim of better productivity and satisfaction. Our research shows that changing one may impact another, and that there are many non-technical factors that may also have an impact. Our survey results lead to some propositions that we discussed above, that may also apply to similar large companies. We discuss the implications of these propositions and mention related work. Lastly, we discuss how our theory and survey instrument may inform future research.

8.1 Impact of managers and work culture on developer job satisfaction

The survey respondents at our case company indicated that they believe that *manager* is the most important factor to them

(ranked #1 in terms of importance in Table 1). While the vast majority are satisfied with their manager, we also see from Fig. 3 that when a manager is a challenge, this correlates with lower satisfaction across 16 different factors (we also found manager correlates with lower overall job satisfaction). Our findings (see Table 6) also suggest that consideration should be given to how managers show *appreciation*, give *feedback* on work, and how they promote a positive *team and work culture*.

A recent study at our case company [43] to identify essential engineering manager attributes found that *growing talent* (which includes providing feedback) and *building team culture* are key to engineers being satisfied with their manager. Our findings also agree with insights from outside software engineering. In their Job Characteristics Theory (JCT), Hackman & Oldham [39], [44] identified *core job characteristics* (including feedback [45]) that influence employee's *psychological states* and affect *work-related outcomes* such as motivation and job satisfaction. The characteristic-state-outcome relationship is moderated by *Context Satisfaction*, which includes satisfaction with one's manager.

8.2 Impactful work and developer job satisfaction

Having *impact at work* and being an *important contributor* were factors that explain variation in overall job satisfaction and perceived productivity scores in the models we presented as part of RQ3. This is in line with JCT's core characteristic *Task Significance* [39], [44] by which employees find a job that has substantial impact for others to be more meaningful to them. Organizations may wish to consider how the impact that developers' work has on end users or products could be made more visible to them.

We also found that *type of work* and how that matches to developer skills also relates to work impact, as well as to job satisfaction and perceived productivity. Similar findings are identified in [43] where great managers are described as *recognizing individuality* by tailoring the work and tasks to the skills and preferences of developers. The Job Characteristics Theory includes similar discoveries; *Knowledge and Skill* contribute to the motivation employees experience for a job and how satisfied they are with it [39], [44].

We further see that the composite factor, *Job characteristics*, is an important variable in the regression model for the C1 developers, those developers that spend more of their time coding and thus may be required to do more complex work. This resonates with the findings by Shaw and Gupta [30] (as mentioned in Section 2) that varying levels of job performance may depend on how complex participants perceive their work to be.

8.3 Impact of engineering systems on perceived productivity

Engineering processes and tools were believed to be moderately or very important by over 85% of the survey respondents, but satisfaction with them is quite low (see Table 1). Furthermore, satisfaction with engineering systems (processes and tools) helps to explain some of the variation in perceived productivity (see Table 6).

Collaboration tools (often an integral part of an engineering system) are likewise important but we saw mixed results in terms of developer satisfaction with collaboration tools (see Fig. 1). This is noteworthy, as demographics at our case company showed that most developers that answered our survey (89%) coordinate their work with three or more people, while 40% coordinate with six or more. How collaboration tools influence productivity, especially at a team level needs further study.

8.4 Impact of non-technical factors on perceived productivity

As we noted above, satisfaction with engineering systems (processes and tools) may help explain some of the variation in perceived productivity, but our models further show that developer satisfaction with other non-technical factors are also important (see Table 6). The related factors that clustered under the work environment variable and the ability to work interruption-free, was a composite factor that explained some of the perceived productivity in our models (see RQ3).

However, we expected that the developer's work environment, in particular having a private office space, would be noted in our survey as being very important, and that we may see poor satisfaction in this regard as many developers at Microsoft work in shared spaces and we had heard them complain in other surveys and in person. Oddly, this factor was not rated that high on importance by many developers, while satisfaction with this factor was mixed (see Table 6). In the regression models, the work environment factor was a significant factor for the entire population and also for senior developers.

8.5 Our theory and survey may guide future research

We anticipate that our theory—in particular, the nuanced bi-directional relationships between satisfaction and perceived productivity, and the set of factors and challenges we identified—can be useful for other researchers that study human aspects of software development. As such, our theory is a descriptive theory, but it also aims to start to explain [42] the relationships it proposes. The propositions we list point to future work to explore further.

The key takeaway from our theory is that the use of satisfaction as a construct for productivity is often assumed, but our findings show that although one impacts the other, they cannot be used in place of the other as other factors play a significant role. In particular, non-technical factors may need to be considered when engineering tool or process improvements are suggested.

Finally, the survey instrument and our analysis can be reused but open ended questions should probe on satisfaction factors and challenges specific to other settings. In our case, we considered one large company, and two levels of work experience and developer activity. A consideration across more nuanced levels of work experience (e.g., after 1, 2 years etc) could lead to quite different factors and challenges that influence the results. Furthermore, other work context variables, such as gender, age, and location could be considered to assess which factors and challenges may influence the interplay between job satisfaction and perceived productivity.

9 LIMITATIONS

We describe the threats to validity of our study and limitations with our proposed theory.

External validity: Single-case empirical studies are historically shown to contribute to scientific discovery [46], and intense observation has delivered insights in the social sciences [47, pp. 95]. The company we studied employs tens of thousands of software engineers that work on diverse products across many domains (operating systems, databases, cloud software, software tools, to productivity software) and they use many tools and processes. Our survey participants varied in role, geography, and age, which means that we captured a range of experiences which may improve generalizability. That said, our results likely generalize more to large software organizations than to small companies or open source settings, or to companies where software is not the primary focus. We do not claim that our results are representative of the views of software engineers in general.

As with any survey, there may have been non-response bias, i.e., the results might not be representative of the population because the participants disproportionately possess certain traits which affect the outcome. In addition, our survey was advertised as a "Software Development Satisfaction and Productivity Survey" and therefore could have been subject to self-selection bias, e.g., developers might have been more likely to participate in the survey if they were unsatisfied (to vent) or highly productive (to gloat). To avoid non-response and self-selection bias, we kept the survey as short as possible, were transparent about the survey length (single-page survey), provided an incentive to participate (raffle), and kept the survey anonymous.

Construct validity: We consider self-reported perceived productivity, as objective productivity metrics in software development remain elusive due to the creative and collaborative nature of development work. In our survey we asked about productivity satisfaction to probe about perceived productivity ("I am satisfied with my productivity at work"). We framed our question to be consistent with other questions about satisfaction/important factors in our survey. We recognize that productivity satisfaction may differ to perceived productivity for some developers. For example, some people might know they're not very productive and are fine with that. We believe that this is the exception because in the survey 97.0% of developers indicated that being productivity is important to them while only 61.1% felt productive. Another threat is that satisfaction may be overly influenced by cognitive state. We inquired about mood and excitement in our survey, and found that although these cognitive states are positively correlated with satisfaction (0.61 and 0.65 respectively), they refer to different cognitive states.

We also asked about the perceived importance of factors and the perceived impact of challenges, as well as perceptions of time spent doing various development tasks. These perceptions may not accurately represent reality nor may they be indicative of future actions that developers may undertake. In particular, we do not claim that the estimates of time spent on certain tasks are accurate, but rather use these estimates to identify different clusters of developers that believe they spend their time differently, as this may influence their perceived productivity and job satisfaction. We asked developers about the time spent because not all activities can be gathered through automated telemetry on a computer, let alone the privacy issues to consider in such an approach. Self-reported time data is commonly in large-scale time-use surveys, such as the American Time Use Survey[48].

Instead of using existing validated measurements, we designed our own measurements for satisfaction, productivity, and other complex psychological constructs. This introduces possible threats to validity, most notably related to construct validity. For instance, response items may not capture the intended meaning of the concepts or constructs or participants might misunderstand the response item due to insufficient conceptualization. To avoid misunderstandings and ambiguous interpretations, we piloted the survey with a small group of employees. Another threat to construct validity is that each factor or construct was measured by a single response item only; therefore no evaluation of reliability of the measures is possible. We chose single response items for satisfaction, productivity, and the 44 factors to keep the survey length reasonable because shorter questionnaires have been found to receive higher response rates [49]. While the use of multi-item scales such as Macdonald and MacIntyre's generic job satisfaction scale [50] is preferable, the use of single items for job satisfaction is acceptable, when the situation precludes a multi-item scale, for example due to survey length. For job satisfaction, it was found that single-item measures (as used in this study) are highly correlated with scale measures of satisfaction [51], [52]. Self-rated, single-item response items for productivity have also been found to correlate with objective productivity measures for software engineers [22]. To further alleviate this threat we combined factors with highly correlated satisfaction scores into composite factors for the regression analysis for RQ3.

Finally, many of the factors, challenges, and work context variables we included are referenced in other literature (either in software engineering or beyond), while the new factors and challenges emerged directly from our open-ended questions in the survey.

Internal validity: We recognize that surveys can introduce biases, may contain ambiguous questions, and may be incomplete. We attempted to mitigate these issues by piloting the survey and analyzing the results. However, even after so many pilots, we found new factors and challenges in the Spring 2017 deployment of the survey. Hence we refined the survey and redeployed the final version in the Fall of 2017. In this final iteration (also piloted), no new factors or challenges emerged from the open-ended responses we solicited. We randomized the order of the items for the factors, challenges, and work activities to offset a bias from the item order, but there may be a response bias due to the people that opted to answer our survey. Self reporting may also be inaccurate as we acknowledge in the paper (especially in terms of the number of hours worked).

We recognize that reliance on a single research method is also limited. While the three month on-site visit was to provide context (informal interviews, meetings, internal documents) to the first author before designing the survey, the research presented in this paper is scoped only to the survey.

Transferability and credibility of our theory: We are confident that the main constructs (overall satisfaction and perceived productivity) and relationships in the general theory we propose are likely transferable to other setting as we build on and have overlapping findings with other rigorous research

outside of software development. However, the specific factors and challenges we found at our case company may not transfer. In terms of our findings at this company, we presented the survey results to numerous organizations at our case company to check on the resonance and credibility of the specific factors, challenges and work context variables we found relevant for the specific instantiations of the theory. These interactive presentations were in lieu of member checking (as our survey was anonymous) but we recognize the feedback from these is not as rigorous as member checking could have been. That is, it was relatively easy for other developers to say that the results we found resonated with them, while the developers we actually surveyed may not have agreed if we were able to talk directly to them.

In terms of the theoretical relationships we put forward, we do not claim that our methods prove causal relationships between overall job satisfaction and perceived productivity. At best our theory is descriptive and partially explanatory, but it is not developed sufficiently to be used (currently) in a predictive manner. But following on the work of Judge et al., (and his theory which is highly regarded in organizational psychology), we do feel that there is likely a bi-directional causal relationship between overall job satisfaction and perceived productivity. This insight is valuable because it may mean that more productive developers are likely to be more satisfied, and that more satisfied developers are likely to be productive, but future work is needed to validate these claims. Conversely, others may consider the direction of the relationship as common sense, but the relationship between employee satisfaction and performance has been debated in organizational psychology literature for decades. Moreover, by showing that other factors (some of which we identify as unique factors for developers, e.g., engineering tools and technical competency and training) correlate with overall job satisfaction, this finding indicates that improving productivity requires attention across many factors, and that a number of additional challenges may need to be addressed to improve satisfaction with those factors, which in turn can improve developer overall job satisfaction and perceived productivity as well.

10 CONCLUSIONS

How to conceptualize productivity and reliably measure satisfaction have been challenges faced by the software engineering community for some time. Attempts to derive metrics that are based solely on software quality attributes or development activities fail to capture the rich relationship that exists between satisfaction and productivity. Through our research, we present a theory (see Fig. 5) that describes how overall job satisfaction and perceived productivity are related (in a bi-directional manner) and articulates how factors and challenges may impact these constructs for particular developers and specific work contexts. Our theory suggests that improving one without paying attention to the other may be detrimental, and that many different social and technical factors may need to be addressed.

In the company we studied, we found that how managers manage had an impact on developer satisfaction with numerous factors, and whether developers can effectively use their skills and believe their work has impact likely impacted their overall job satisfaction, and in turn their perceived productivity. Addressing the challenges our participants reported to us (e.g., improving how managers manage, and work culture) suggests a path for action that we anticipate may improve developer overall job satisfaction and perceived productivity in this company.

Although some of the specific findings from our study resonate with the results from other developer surveys (notably impactful work and manager quality), we propose that the survey instrument we developed and our survey analysis approach can be used to reveal which specific factors may more greatly influence overall job satisfaction and perceived productivity outcomes in other settings (e.g., factors such as stress may play a bigger role in some work contexts). Our hope is that the results from this survey may lead to actionable insights that will improve developer satisfaction and perceived productivity in these other settings.

REFERENCES

[1] B. W. Boehm, "Improving software productivity," in *Computer*. Citeseer, 1987.

[2] T. Acton and W. Golden, "Training the knowledge worker: a descriptive study of training practices in irish software companies," *Journal of European Industrial Training*, vol. 27, no. 2/3/4, pp. 137–146, 2003.

[3] M. Foucault, M. Palyart, X. Blanc, G. C. Murphy, and J.-R. Falleri, "Impact of developer turnover on quality in open-source software," in *Proceedings of the 2015 10th Joint Meeting on Foundations of Software Engineering*. ACM, 2015, pp. 829–841.

[4] P. C. Rigby, Y. C. Zhu, S. M. Donadelli, and A. Mockus, "Quantifying and mitigating turnover-induced knowledge loss: case studies of chrome and a project at avaya," in *Proceedings of the 38th International Conference on Software Engineering*. ACM, 2016, pp. 1006–1016.

[5] B. Dagenais, H. Ossher, R. K. Bellamy, M. P. Robillard, and J. P. De Vries, "Moving into a new software project landscape," in *Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering-Volume 1*. ACM, 2010, pp. 275–284.

[6] A. N. Meyer, T. Fritz, G. C. Murphy, and T. Zimmermann, "Software developers' perceptions of productivity," in *Proceedings of the 22th International Symposium on Foundations of Software Engineering*, November 2014.

[7] S. Beecham, N. Baddoo, T. Hall, H. Robinson, and H. Sharp, "Motivation in software engineering: A systematic literature review," *Information and software technology*, vol. 50, no. 9-10, pp. 860–878, 2008.

[8] D. Graziotin, F. Fagerholm, X. Wang, and P. Abrahamsson, "On the unhappiness of software developers," *arXiv preprint arXiv:1703.04993*, 2017.

[9] T. A. Wright and R. Cropanzano, "Psychological well-being and job satisfaction as predictors of job performance." *Journal of occupational health psychology*, vol. 5, no. 1, p. 84, 2000.

[10] T. A. Judge, C. J. Thoresen, J. E. Bono, and G. K. Patton, "The job satisfaction–job performance relationship: A qualitative and quantitative review." *Psychological bulletin*, vol. 127, no. 3, p. 376, 2001.

[11] C. Sadowski and T. Zimmermann, *Rethinking Productivity in Software Engineering*. Springer, 2019.

[12] D. Graziotin, F. Fagerholm, X. Wang, and P. Abrahamsson, "What happens when software developers are (un) happy," *arXiv preprint arXiv:1707.00432*, 2017.

[13] A. P. Brief and L. Roberson, "Job attitude organization: An exploratory study," *Journal of Applied Social Psychology*, vol. 19, no. 9, pp. 717–727, 1989.

[14] H. Sharp, N. Baddoo, S. Beecham, T. Hall, and H. Robinson, "Models of motivation in software engineering," *Information and software technology*, vol. 51, no. 1, pp. 219–233, 2009.

[15] A. C. C. Franca, D. E. Carneiro, and F. Q. da Silva, "Towards an explanatory theory of motivation in software engineering: A qualitative case study of a small software company," in *Software Engineering (SBES), 2012 26th Brazilian Symposium on*. IEEE, 2012, pp. 61–70.

[16] A. C. C. França, A. C. de Araújo, and F. Q. da Silva, "Motivation of software engineers: A qualitative case study of a research and development organisation," in *Cooperative and Human Aspects of Software Engineering (CHASE), 2013 6th International Workshop on*. IEEE, 2013, pp. 9–16.

[17] A. C. C. França, F. Q. Da Silva, A. de LC Felix, and D. E. Carneiro, "Motivation in software engineering industrial practice: A cross-case analysis of two software organisations," *Information and Software Technology*, vol. 56, no. 1, pp. 79–101, 2014.

[18] C. França, H. Sharp, and F. Q. Da Silva, "Motivated software engineers are engaged and focused, while satisfied ones are happy," in *Proceedings of the 8th ACM/IEEE International Symposium on Empirical software Engineering and Measurement*. ACM, 2014, p. 32.

[19] S. Wagner and M. Ruhe, "A systematic review of productivity factors in software development," *arXiv preprint arXiv:1801.06475*, 2018.

[20] ——, "A structured review of productivity factors in software development," TUM Institut fur informatik, Tech. Rep. TUM-INFO-09-I0832-0/1.-FI, 2008.

[21] A. N. Meyer, T. Zimmermann, and T. Fritz, "Characterizing software developers by perceptions of productivity," in *Empirical Software Engineering and Measurement (ESEM), 2017 ACM/IEEE International Symposium on*. IEEE, 2017, pp. 105–110.

[22] E. Murphy-Hill, C. Jaspan, C. Sadowski, D. Shepherd, M. Phillips, C. Winter, A. Knight, E. Smith, and M. Jorde, "What predicts software developers' productivity?" *IEEE Transactions on Software Engineering*, pp. 1–1, 2019.

[23] J. Aranda and G. Venolia, "The secret life of bugs: Going past the errors and omissions in software repositories," in *Proceedings of the 31st International Conference on Software Engineering*. IEEE Computer Society, 2009, pp. 298–308.

[24] D. B. Greenberger, S. Strasser, L. L. Cummings, and R. B. Dunham, "The impact of personal control on performance and satisfaction," *Organizational behavior and human decision processes*, vol. 43, no. 1, pp. 29–51, 1989.

[25] M. Fishbein, "The prediction of behaviors from attitudinal variables," *Advances in communication research*, pp. 3–31, 1973.

[26] J. M. Olson and M. P. Zanna, "Attitudes and attitude change," *Annual review of psychology*, vol. 44, no. 1, pp. 117–154, 1993.

[27] J. L. Pierce, D. G. Gardner, L. L. Cummings, and R. B. Dunham, "Organization-based self-esteem: Construct definition, measurement, and validation," *Academy of Management journal*, vol. 32, no. 3, pp. 622–648, 1989.

[28] S. P. Brown and R. A. Peterson, "Antecedents and consequences of salesperson job satisfaction: Meta-analysis and assessment of causal effects," *Journal of marketing research*, vol. 30, no. 1, p. 63, 1993.

[29] C. Orpen, "The relationship between satisfaction and performance under contingent and noncontingent reward schedules." *Psychological Studies*, 1981.

[30] J. D. Shaw and N. Gupta, "Job complexity, performance, and wellbeing: When does supplies-values fit matter?" *Personnel Psychology*, vol. 57, no. 4, pp. 847–879, 2004.

[31] M.-A. Storey, T. Zimmermann, C. Bird, J. Czerwonka, B. Murphy, and E. Kalliamvakou, "Supplemental material for towards a theory of software developerjob satisfaction and perceived productivity," http://bit.ly/2oXIrE4, 7 2019.

[32] T. Punter, M. Ciolkowski, B. Freimut, and I. John, "Conducting on-line surveys in software engineering," in *2003 International Symposium on Empirical Software Engineering, 2003. ISESE 2003. Proceedings*. IEEE, 2003, pp. 80–88.

[33] S. Dowdy, S. Wearden, and D. Chilko, *Statistics for research*. John Wiley & Sons, 2011, vol. 512.

[34] B. Johnson, T. Zimmermann, and C. Bird, "The effect of work environments on productivity and satisfaction of software engineers," *IEEE Transactions on Software Engineering*, pp. 1–1, 2019.

[35] J. Fox, *Applied Regression Analysis and Generalized Linear Models (2nd edition)*. Sage Publications, 2008.

[36] J. A. Durlak, "How to select, calculate, and interpret effect sizes," *Journal of pediatric psychology*, vol. 34, no. 9, pp. 917–928, 2009.

[37] L. Kaufman and P. J. Rousseeuw, *Finding groups in data: an introduction to cluster analysis*. John Wiley & Sons, 2009, vol. 344.

[38] P. J. Rousseeuw, "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis," *Journal of computational and applied mathematics*, vol. 20, pp. 53–65, 1987.

[39] J. R. Hackman and G. R. Oldham, "Development of the job diagnostic survey." *Journal of Applied psychology*, vol. 60, no. 2, p. 159, 1975.

[40] M. Fishbein and I. Ajzen, *Belief, attitude, intention, and behavior: An introduction to theory and research*. Addison-Wesley, 1977.

[41] K.-J. Stol and B. Fitzgerald, "Theory-oriented software engineering," *Science of Computer Programming*, vol. 101, pp. 79 – 98, 2015, towards general theories of software engineering. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167642314005425

[42] D. Sjøberg, D. T., B. Anda, and J. Hannay, *Building Theories in Software Engineering*. Springer-Verlag, London, 2008.

[43] E. Kalliamvakou, C. Bird, T. Zimmermann, A. Begel, R. DeLine, and D. M. German, "What makes a great manager of software engineers?" *IEEE Transactions on Software Engineering*, vol. 45, no. 1, pp. 87–106, 2019.

[44] J. R. Hackman and G. R. Oldham, *Work redesign*. Addison-Wesley, 1980.

[45] R. Katz, "Motivating technical professionals today," *Research-Technology Management*, vol. 48, no. 6, pp. 19–27, 2005.

[46] B. Flyvbjerg, "Five misunderstandings about case-study research," *Qualitative inquiry*, vol. 12, no. 2, pp. 219–245, 2006.

[47] A. Kuper, *The social science encyclopedia*. Routledge, 2004.

[48] L. L. Stinson, "Measuring how people spend their time: a time-use survey design," *Monthly Lab. Rev.*, vol. 122, p. 12, 1999.

[49] E. Deutskens, K. De Ruyter, M. Wetzels, and P. Oosterveld, "Response rate and response quality of internet-based surveys: An experimental study," *Marketing letters*, vol. 15, no. 1, pp. 21–36, 2004.

[50] S. Macdonald and P. MacIntyre, "The generic job satisfaction scale: Scale development and its correlates," *Employee Assistance Quarterly*, vol. 13, no. 2, pp. 1–16, 1997.

[51] J. P. Wanous, A. E. Reichers, and M. J. Hudy, "Overall job satisfaction: how good are single-item measures?" *Journal of applied Psychology*, vol. 82, no. 2, p. 247, 1997.

[52] C. L. Dolbier, J. A. Webster, K. T. McCalister, M. W. Mallon, and M. A. Steinhardt, "Reliability and validity of a single-item measure of job satisfaction," *American Journal of Health Promotion*, vol. 19, no. 3, pp. 194–198, 2005.

Margaret-Anne Storey is a Professor of Computer Science at the University of Victoria, Canada.

Thomas Zimmermann is a Principal Researcher at Microsoft. He received his Ph.D. degree from the Saarland University in Germany. His research interests include empirical software engineering, data science, and mining software repositories. His work focusses on software productivity, software analytics, and recommender systems for software development. He is a member of the IEEE Computer Society. His homepage is http://thomas-zimmermann.com.

Christian Bird is a Principal Researcher in the Empirical Software Engineering group at Microsoft Research. He focuses on using qualitative and quantitative methods to both understand and help software teams. Christian received his Bachelor's degree from Brigham Young University and his Ph.D. from the University of California, Davis. He lives in Redmond, Washington with his wife and three (very active) children.

Jacek Czerwonka is a Principal Engineering Manager at Microsoft.

Brendan Murphy is a Sr. Principal Researcher at Microsoft.

Eirini Kalliamvakou is a Postdoctoral Fellow at the University of Victoria, Canada.